THE UNIVERSITY OF HULL

Advanced Document Data Extraction Techniques to Improve Supply Chain Performance

being a Thesis submitted for the Degree of

(PhD in Systems Science)

in the University of Hull

by

Vikash Sharma
MSc. Advanced Computational Method, University of Leicester, UK
MSc. Computer Application, Symbiosis International University, India
BCA, WBUT, India

(July 2021)

ABSTRACT

In this thesis, a novel machine learning technique to extract text-based information from scanned images has been developed. This information extraction is performed in the context of scanned invoices and bills used in financial transactions. These financial transactions contain a considerable amount of data that must be extracted, refined, and stored digitally before it can be used for analysis. Converting this data into a digital format is often a time-consuming process. Automation and data optimisation show promise as methods for reducing the time required and the cost of Supply Chain Management (SCM) processes, especially Supplier Invoice Management (SIM), Financial Supply Chain Management (FSCM) and Supply Chain procurement processes. This thesis uses a cross-disciplinary approach involving Computer Science and Operational Management to explore the benefit of automated invoice data extraction in business and its impact on SCM. The study adopts a multimethod approach based on empirical research, surveys, and interviews performed on selected companies.

The expert system developed in this thesis focuses on two distinct areas of research: Text/Object Detection and Text Extraction. For Text/Object Detection, the Faster R-CNN model was analysed. While this model yields outstanding results in terms of object detection, it is limited by poor performance when image quality is low. The Generative Adversarial Network (GAN) model is proposed in response to this limitation. The GAN model is a generator network that is implemented with the help of the Faster R-CNN model and a discriminator that relies on PatchGAN. The output of the GAN model is text data with bonding boxes. For text extraction from the bounding box, a novel data extraction framework consisting of various processes including XML processing in case of existing OCR engine, bounding box pre-processing, text clean up, OCR error correction, spell check, type check, pattern-based matching, and finally, a learning mechanism for automatizing future data extraction was designed. Whichever fields the system can extract successfully are provided in key-value format.

The efficiency of the proposed system was validated using existing datasets such as SROIE and VATI. Real-time data was validated using invoices that were collected by two companies that provide invoice automation services in various countries. Currently, these scanned invoices are sent to an OCR system such as OmniPage, Tesseract, or

ABBYY FRE to extract text blocks and later, a rule-based engine is used to extract relevant data. While the system’s methodology is robust, the companies surveyed were not satisfied with its accuracy. Thus, they sought out new, optimized solutions. To confirm the results, the engines were used to return XML-based files with text and metadata identified. The output XML data was then fed into this new system for information extraction. This system uses the existing OCR engine and a novel, self-adaptive, learning-based OCR engine. This new engine is based on the GAN model for better text identification. Experiments were conducted on various invoice formats to further test and refine its extraction capabilities. For cost optimisation and the analysis of spend classification, additional data were provided by another company in London that holds expertise in reducing their clients' procurement costs. This data was fed into our system to get a deeper level of spend classification and categorisation. This helped the company to reduce its reliance on human effort and allowed for greater efficiency in comparison with the process of performing similar tasks manually using excel sheets and Business Intelligence (BI) tools.

The intention behind the development of this novel methodology was twofold. First, to test and develop a novel solution that does not depend on any specific OCR technology. Second, to increase the information extraction accuracy factor over that of existing methodologies. Finally, it evaluates the real-world need for the system and the impact it would have on SCM. This newly developed method is generic and can extract text from any given invoice, making it a valuable tool for optimizing SCM. In addition, the system uses a template-matching approach to ensure the quality of the extracted information.

Keywords

Information Extraction, Invoice Data Extraction, Receipt Data Extraction, Automated Data Extraction, Machine Learning, Spend Classification, Generative Adversarial Network, CNN, RCNN, Bi-LSTM, GAN.

Acknowledgements

What does knowledge mean to you?

"Knowledge... is everything

Knowledge is spirit, wisdom, courage, light, sound.

Knowledge is my Bible, God...

Knowledge is my teacher."

- Movie: Black (2005)

First and foremost, I am very thankful to my PhD supervisor, Prof. Nishikant Mishra, for sharing his valuable and kind guidance and motivate me throughout the term of my PhD research. His guidance and constructive criticism have always pushed me to think beyond. His knowledge and experience in supply chain and computer science have helped me understand the nature of problems in business and technology. His dynamism, vision, and honesty have left a lasting impression on me. He has taught me how to do research in the simplest way possible. I would also like to thank him for his friendship, empathy, and great sense of humour. I am extending my heartfelt thanks to his family for their acceptance and patience during the discussions. Working and researching under his direction was a great honour and privilege.

I wish to show my gratitude towards the staff and students' welfare of the University of Hull, who supported me immensely during the research. The team at doctoral college was very quick to respond, especially Andrea C Bell. She is incredibly supportive and helpful. I thank the faculty members, especially my co-supervisor, Professor Yasmin Merali, for her valuable feedback during the review meeting. I also thank Dr Ashish Dwivedi for his guidance during the internal PhD review and feedback. I would also like to sincerely thanks Dr Rameshwar Dubey and Dr Pravin S Metkewar. Without their persistent help, the goal of this research would not have been realised and completed with success. They have been extremely helpful in guiding me in my PhD career.

Finally, I would like to thank my family and friends for their love, understanding and continuous support. The research journey has been very amazing, and it would not have been possible without constant encouragement from near and dear ones.

Table of Contents

List of Figures

List of Tables

List of Abbreviation

Table 1: List of Abbreviation

"Devised by the author".

OCR	Optical Character Reader
NLP	Natural Language Processing
CNN	Convolution Neural Network
ANN	Artificial Neural Network

BK	Burkhard Keller
RNN	Recurrent Neural Network
SVM	Support Vector Machine
R-CNN	Regional Convolution Neural Network
FR-CNN	Fast Regional Convolution Neural Network
GAN	Generative Adversarial Network
RoI	Region of Interest
GRU	Gated Recurrent Units
LSTM	Long short-term memory
SSD	Single Shot detector
HCR	Handwriting Character Recognition
GCNN	Graph Convolutional Neural Network
DNN	Deep Neural Network
HMM	Hidden Markov Model
KNN	K-Nearest Neighbours
FPN	Feature Pyramid Network
CTPN	Connectionist Text Proposal Network
EATEN	Entity-aware Attention Text Extraction Network
OpenCV	Open-Source Computer Vision Library
YOLOv3	You Only Look Once-version 3
SROIE	Scanned receipts OCR and information extraction
VATI	Value Added Tax Invoices
CSV	Comma Separated Values
IoU	Intersection-over-Union
cGAN	conditional Generative Adversarial Networks
XML	Extensible Markup Language
FRE	Fine Reader Engine
FSNS	French Street Name Signs
Bi-LSTM	Bidirectional Long Short-Term Memory
LM	Language Model
NMS	Non-Maximum Suppression
CRNN	Convolutional Recurrent Neural Network
ESM	Exact String Matching

ASM	Approximate String Matching
RPN	Regional Proposal Network
CBOW	continuous bag-of-words
VRDs	Visually Rich Documents
FTFDNet	Financial Ticket Faster Detection network
RLSA	Run Length Smearing Algorithm
NLT	Natural Language Text
SWRL	Semantic Web rule language
NER	Named Entity Recognition
MSER	Maximally Stable Extremal Region
MRF	Markov Random Field
GNNs	Graph Neural Networks
AIESI	Abstractive Information Extraction from Scanned Invoices
KIPE	Key Invoice Parameter Extraction
OASM	Online Approximate String Matching
HTA	Hough Transform Accumulator
CTC	Connectionist Temporal Classification
RMRS	Regular Matching and Recursive Segmentation
EESRGAN	Edge-Enhanced Super Resolution Generative Adversarial network
MTGAN	Multitask Generative Adversarial Network
RRDB	Residual in Residual Dense Blocks
GAN-KD	Generative Adversarial Networks - Knowledge Distillation
LD	Levenshtein Distance
TOL	Tolerance Limit
IR	Information Retrieval
SCMIL	Sequence-to-sequence text Correction Model for Indic Languages
DEA	Data Envelopment Analysis
SRM	Supplier Relationship Management
QDC	Quadratic Discriminate Classifier
KYC	Know Your Customer
BI	Business Intelligence
ICT	Information and Communication Technology
FFT	Fast Fourier Transform

SEK	Stroke Ending Keypoint
MSERs	Maximally Stable Extremal Regions
SVT	Street View Text
AON	Arbitrary Orientation Network
AF-RPN	Anchor-Free Region Proposal Network
RCR-CNN	Rotated Cascade Region-based Convolutional Neural Network
PSENet	Progressive Scale Expansion Network
SWT	Stroke Width Transform
WER	Word Error Rate
LWER	Lattice Word Error Rate
CRF	Conditional Random Fields
FSM	Finite State Machine
RNN	Recurrent Neural Network
SGD	Stochastic Gradient Descent
CRAFT	Character Region Awareness For Text
EAST	Efficient and Accurate Scene Text
AEM	Adversarial-neural Event Model
FCA	Formal Concept Analysis

1. Introduction

In this thesis, a novel, automated information extraction system for scanned invoices and bills is proposed. The intention of this research is to reduce the time and effort required to convert scanned financial documents into digital formats and to improve the accuracy of this information extraction process. The proposed system uses a machine learning, template-based, pattern matching approach and a state-of-the-art, rule-based system that enhances data extraction accuracy with minimal human intervention. As the system is exposed to increasing amounts of real-time data over time, the need for human intervention is further reduced due to its ability to learn from past data interactions. As such, it will generate more accurate output in less time than current systems. The study also highlights the importance of automated invoice data extraction systems in financial supply chain management. This chapter describes the study's background research, motivation, and research objectives in detail. It also further explains the analytical framework utilized in the production of this thesis, data collection methodologies, and finally, it presents the thesis's overall structure.

1.1. Background

The field of information or data extraction deals with the transference of data from unstructured sources (e.g., printed documents, letters, and bills) to structured forms such as spreadsheets, text-based documents, or data entry programs. In the context of business management or SCM, data extraction systems (whether automated or manual) are responsible for extracting data from scanned financial documents or transaction records (i.e., invoices and receipts) and storing it in a meaningful digital format that can be used for analysis. The process of data extraction is an important aspect of SCM due to the prevalence of transaction records and the importance of the information they contain. The details frequently contained in these transaction records include items bought or sold, buyer and seller details, transaction dates, the amount paid or received, calculated tax, discounts applied, and several others, which will be discussed in greater detail in Chapter 2.

To generalise, invoice and receipt data extraction serves two main purposes. The first purpose of data extraction is to help ensure accurate accounting and regulatory compliance. According to most national and international financial norms, companies must store transaction records for use when filing taxes or participating in audits. To this end, the information contained in transaction records must be extracted and stored in an accurate and efficient way. The second purpose of invoice and receipt data extraction is to build datasets that can be used in market analysis, customer relations analysis, fraud detection, banking, and finance.

Most transaction records are stored as either digital images or PDFs. Extracting data from these scanned hard copy formats is a complicated process for several reasons. For one, data is not uniformly structured or represented across different records. Both structure and representation vary by industry, country, company, and sometimes even within a single organisation. The process of extracting data from images is complicated further by variance in font size, alignment, and text orientation (Taghva et al., 1994b), as seen in figure 1.

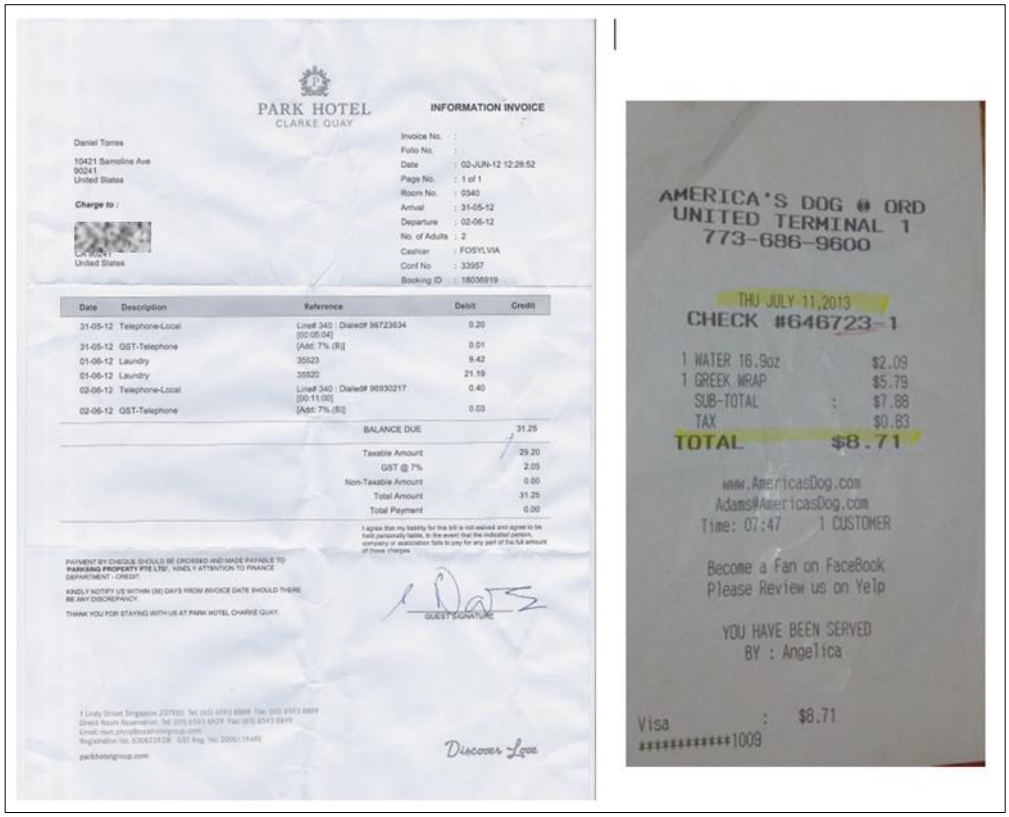


Figure 1: Sample invoice and bills.

"Devised by the author".

Paper quality, text position, format, text highlighting, handwritten elements, and overwriting are all traits that pose a challenge for automated data extraction systems, especially when the data in question is contained in an image format. This is problematic for the businesses, executives, and analysts trying to gather meaningful insights from data available to them. With more (and more accurate) data available for analysis, companies can make smarter decisions and more accurate predictions. Often, the limiting factor in terms of data quantity and quality is due to the system that is used for data extraction. Thus, systems that can overcome the challenges of unstructured data representation and extract data accurately and efficiently can provide a great competitive edge. Developing these systems is of great interest to the field of 'Big Data'.

Big Data is a field of inquiry that seeks to develop methods and systems for analysing and utilising complex datasets through the identification of factors such as volume, velocity, veracity, variability, and viscosity. Though Big Data is difficult to define, size is present in most interpretations of the concept (Gandomi & Haider, 2015). Big Data has been hailed as a potentially revolutionary management tool that exceeds the analytic power of the simpler methodologies of the past (Wamba et al., 2015). Although many dimensions of Big Data have been studied and researched in SCM, there is still a gap in the understanding of unstructured data within the field. Despite the significance of the information hidden within unstructured financial data, it remains difficult to extract relevant information. The continual generation of new, unstructured transaction records combined with the difficulty of transforming this unstructured data into a structured form has led to an enormous backlog of underutilized data. In other words, "The volume and variety of data have far outstripped the capacity of manual analysis, and in some cases have exceeded the capacity of conventional databases" (Provost & Fawcett, 2013). This phenomenon has become known as the 'Big Data problem'. Improving our understanding of unstructured data and our ability to manipulate, transform, and store it will be of great benefit to businesses in the future.

The importance of unstructured data in the digital age as a result of social media, online shopping, and news is difficult to overstate. In 2013, (Insurance et al., 2013) estimated that around 80-85% of data are in an unstructured form. Later, in 2015, (Gandomi & Haider, 2015) estimated that this figure was closer to 95%. Based on estimates like these, it can safely be stated that the need to understand these unstructured data forms is great. Due to data being maintained in digital formats, "these digital shadows are the subjects

of Big Data research, which optimists see as an outstandingly large sample of real behaviour that is revolutionizing social science" (Wu & Huberman, 2007; Onnela & Reed-Tsochas, 2010; Golder & Macy, 2011; Aral & Walker, 2012; Ormerod, 2012; Bentley et al., 2014).

As such, one primary goal of Big Data research is finding new ways to automatically process these huge quantities of unstructured data with the ultimate aim of improving business operation. With the vast amount of data now available, companies in almost every industry are mining data for competitive advantages. Business leaders are seeking out larger datasets and the tools to analyse them in the hopes of gaining informational advantages over competitors. These informational advantages allow them to make smarter decisions and more accurate predictions which, in turn, translate into economic advantages. Firms that recognize the importance of data have distinct advantages over firms that rely on instinct and intuition for decisions. This drive to gain advantages has pushed Big Data towards developing an enhanced understanding of unstructured data forms as well as improved methods for data extraction.

There are two primary methods for data extraction. The first method is by human intervention, whereby a human reads the transaction record and documents the relevant details using some data entry software. The second method is by some form of automated technology, whereby a program is developed to scan transaction records and record the data without the need for human intervention. The latter method relies on a software technology called an Optical Character Reader (OCR). An OCR is a type of software that automatically converts printed text into digital text. It serves as a translator, partially bridging the language gaps that differentiate humans and computer understanding. As humans, we are able to extract meaning from an immense variety of text, regardless of size, alignment, or spacing. Standard computers require uniform rules to understand a text. When we need computers to understand information that is represented in forms that are typically only understood by humans, we must develop OCR software to act as a translator, and this software requires more processing and complex functionalities than the average computer.

Today, OCR is used in a variety of contexts and for a variety of reasons. One example of this variety is its use in libraries as a tool for digitizing and preserving books, diaries, letters, and manuscripts. Another example of OCR's utility can be seen in its application as a tool for digitizing mail, bills, and credit card statements.

OCR software essentially allows computers to read as we do by giving them the tools required to convert the image-based text and writing into machine-encoded text formats such as .txt or .doc. While OCR software is able to convert image-based text to machine-encoded text, it remains a challenging task as images include various and unencoded typefaces which alter the orientation and characteristics of the text. Handwritten images are equally difficult to convert because every person has a unique manner and style of writing text. OCR overcomes both challenges by utilizing pattern recognition, feature detection, or a combination of both mechanisms.

Pattern recognition works by individually comparing the characters which make up the text in question to a stored database of fonts and typefaces with the goal of identifying matches. When a series of matches are located between the external characters and the and internal characters contained within the database, the system can recognise and process the text. However, this method relies on completely uniform characters to be implemented. As OCR was first being theorized, this rule of pattern recognition necessitated the creation of fonts such as OCR-A with uniform features that could be understood by both computers and humans. Now, more refined feature recognition techniques exist which can identify characters by how well they adhere to predetermined rules. In place of recognizing complete complex patterns of characters, OCR now recognizes individual component features like stroked lines, curves, and cells. For example, a capital ‘A’ might be defined as a character made up of three strokes, no curves, and a single cell.

Recent advancements in the printed document digitisation technology have led to the production of large-scale document images. “For large scale processing and retrieval, the document images are converted to corresponding text files using OCR” (Mohapatra et al., 2013). The quality of the document input into an OCR system has a direct impact on the quality of the OCR system’s output. If an image is of low quality or is distorted in some way, the accuracy of the OCR output is adversely affected. These errors also occur during various stages of OCR, like in part-of-speech tagging. It even might occur using boundary detection in sentences and tokenisation or words. These errors are commonly caused when OCR fails to recognize a character causing errors in the output text. OCR extracts data, which further gets validated by human intervention before saving it in the system. Currently, many businesses are doing automated data entry with software like OmniPage, ABBYY FRE or Textract. One of the many applications of OCR is the automation of

business invoice processing by extracting data from them. Many businesses still use printed invoices that need to be mined for data. Even if the invoices are being sent to the customers in a digital format, the information contained within them still needs to be processed, either manually or by using an OCR tool.

According to (Miloudi et al., 2016), the concept of the supply chain refers to the flow of materials, information, payments, and services from raw materials suppliers through manufacturers, warehouses, and distributers, to the end consumer. It includes the organisations and processes that create value and deliver information, products, and services to customers. The financial supply chain is a key part of the supply chain system and involves the flow and use of cash throughout the physical network. The management of these financial flows involves money transfers, payments, scheduled payments, and countless other complex phenomena. The inflows and outflows of cash are constant and continue throughout an organisation's lifespan. These cash flows go mostly unnoticed because they depend on the movement of goods and are thus reliant on market demand. The structural non-synchronisation of cash flows with the transfer of products and services impacts the ability of a company to operate smoothly. Nonetheless, maintaining and improving financial management performance is challenging for almost all companies. The authors have addressed the financial issues of SCM by considering a specific problem: the scheduling of invoice payments and cash collection to improve working capital performances. Therefore, if an aspect of invoice management (e.g., automation of printed invoices) can be optimised, businesses will see operational improvements. In other words, there is a cost to maintaining these data sets and reducing this cost will help the business make decisions and take actions more quickly. The details of the cost optimisation and spend analysis have been explained in the performance evaluation section of Chapters 3 and 4. Further, in appendix III, the detailed assessment of spend analysis have been presented. It basically includes spend classification, cost optimization and later, supplier selection and comparison.

Dealing with supplier invoices is a crucial aspect of the operation of most businesses. Occasionally, a delay when entering data from an invoice into a data entry system can result in a late payment to a supplier. These delays can occur whether the conversion of printed invoices was performed manually or automatically. In some cases, these late payments lead to penalties (Petr, 2019). To optimize the manual entry or verification of data, high-quality data extraction is essential. Although there are many well-known

systems for data extraction, it is still a challenge to apply custom business requirements to them in order to further increase data the quality of extracted data.

Invoice data extraction has a large body of academic literature behind it as well as considerable application in real-world contexts. Since this research is only focused on the invoice and receipt text extraction, I did the research based on the existing methods for those tasks and developed research objectives in line with this research and studied those. I have undertaken the work to review all the studies in OCR and text extraction. And later, it emphasised the focus on study related to invoice data extraction. The figure below highlights the background work in invoice data extraction related work. A detailed explanation is presented in the literature review section of chapter two.

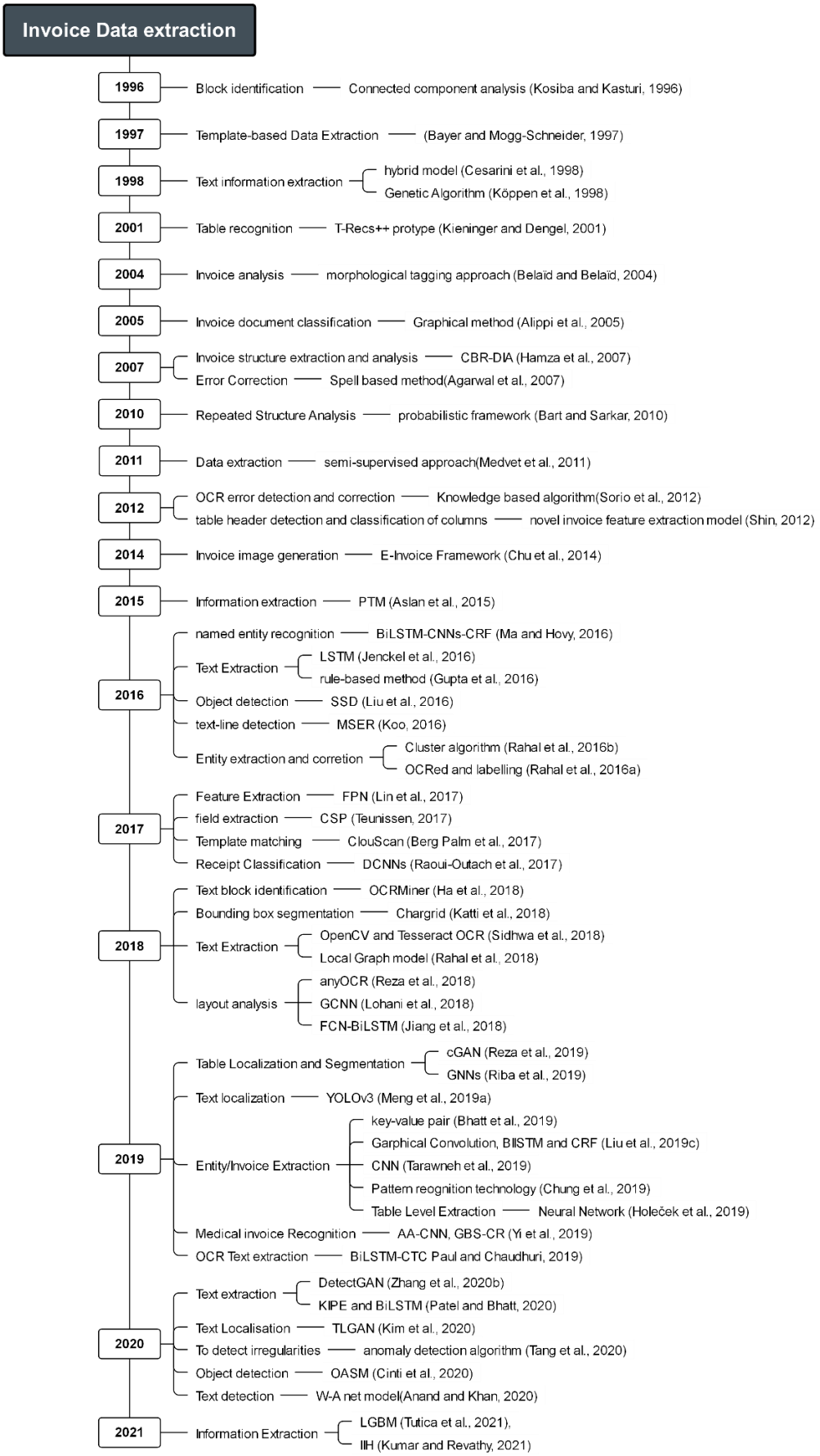


Figure 2: The research hierarchy of data extraction for invoice/ receipts datasets.

"Devised by the author".

According to figure 2, in the 1990s, researchers were using basic image processing algorithms to overcome the challenges of invoice data extraction. In the 1996s (Kosiba & Kasturi, 1996) did research on invoice block identification-based analysis of requirements and drawbacks within the subject. Furthermore, in the 2000s, researchers focused on table detection (Kieninger & Dengel, 2001), classification (Alippi et al., 2005) and extraction (Hamza et al., 2007) of the text using basic machine learning models. These included both supervised and unsupervised learning models like CNN, RNN, ANN, and others. However, they still faced problems related to large dataset management, image quality, and performance.

Between 2010 to 2020, deep learning-based algorithms began to gain attraction in the fields of text extraction (Bart & Sarkar, 2010), invoice data extraction, and spell-checking problems. Deep learning models such as GAN (Reza et al., 2019) and LSTM (Jenckel et al., 2016) were then applied. The addition of deep learning to the toolset of invoice data extraction research has led the field to its best performing models. Today, according to (Kumar & Revathy, 2021), further improvements are continually being made by various companies and researchers to overcome the limitations of existing data extraction systems and make invoice data extraction systems fully automated. This thesis both responds and adds to this body of research.

1.2. Motivation

The research for this thesis on automated data extraction methods was partially motivated by the numerous benefits to FSCM that automatic invoice data extraction can offer, including fewer delays and cost reductions. The supply chain performance contribution is explained in detail in appendix section III. This reports how better spend analysis and supplier selection helps in the supply chain process. Invoice management is a time-consuming process, especially when documents exist in a scanned or image-based format. To reduce the time required to process invoices, a faster automation system is required. While many industries are already using automated data extraction systems, there is still considerable room for improvement in terms of the speed of the extraction process, the quantity of data that is extracted, and the overall accuracy of the extraction. Additionally, many of these systems still require considerable human intervention, which could be avoided by improving the automated systems. The amount of information contained

within transaction records like invoices and receipts, along with the time it takes to transfer their information from one form to another, have established automated data extraction as a key element of SCM decision-making. This thesis focuses on Financial Supply Chain optimization for faster Supplier Invoice Management. Examples of its possible impact areas include cash flow maintenance, currency exchange considerations, product pricing, customer-vendor relationship, and deeper expenses classification. Every factor gets impacted due to a delay in printed information being added to the system. It also gets impacted if not all the relevant data is extracted. Every aspect is covered here and then narrowed down to define the scope based on the respective impact factor.

Furthermore, most of the existing software and systems rely on a minimum number of fields required by respective business regulations and purposes. The idea of this system is to extract almost all the information for use in subsequent analytics. Currently, no single research is being carried out to develop a model that can extract all the invoicing information. Researchers typically target between five and eight data points or fields and use these to determine the accuracy of their model. With the help of this study, I would share all the work that has already been done in this area and, at the same time, propose a new method based on machine learning techniques by further adding a novel rule-based solution. This framework will further add valuable input to ongoing research in this field.

Furthermore, as was stated in the introduction, there has been an enormous increase in the number of transactions and invoices generated by businesses in recent years. This represents a sizable problem for Big Data, too. The analysis of this so-called ‘Big Data’ has received considerable attention in the past 4 to 5 years. Furthermore, the evolution of digital technologies and the Internet of Things has led to a huge, Big Data backlog at most factories, organisations, or even at the individual level. Web-based tools and applications also lead to the creation of significant amounts of Big Data, especially through social media usage, internet documents, and other ways of sharing data on the internet (Acharjya & Ahmed, 2016). These data sets are so large and complex that they are difficult to analyse and manage using traditional database management tools (Acharjya & Ahmed, 2016), as these data sets sometimes reach sizes in the petabytes size and come in both structured and unstructured forms.

Big Data analytics must be involved to process high volumes of complex data using traditional and machine learning computational techniques (Kakhani et al., 2013; Acharjya & Ahmed, 2016). There are many methods of data extraction (Gandomi &

Haider, 2015). Unlike the hardware computing problem as stated by Moore's law, handling of data at massive scales still exist. It has been pointed out that this is the reason that Big Data is called 'Big Data'. It is data which cannot be handled by using the most current analytic and management methods because the data volume has increased to the extent that makes it impossible to store in a single machine. Therefore, standard data analysis is impossible (Fisher et al., 2012; Tsai et al., 2015). From a text analytics perspective, (Agarwal et al., 2007) state that the importance of text mining applications is growing proportionally to the exponential growth of electronic text. They accept that the internet is not the only source. One of the contributors mentions handwritten and printed documents, which have been analysed by the OCR process as a large pool of electronic text documents. While research is being carried out on unstructured data online, data that are maintained in printed forms like invoices and bills have not been the focus of much research.

(Bartoli et al., 2014) add that despite huge advances and widespread diffusion of Information and Communication Technology (ICT), manual data entry is still an essential ingredient of many inter-organisational workflows. In many practical cases, human operators who extract the desired information from printed documents and insert that information into another document or application are the glue between different organisations. As every firm generates invoices in its own firm-specific template, it becomes challenging for the receiver to find the desired items on each invoice. For example, invoice number, date, total, VAT amount all differ in their locations, sizes, and characteristics from one invoice to another. Automating workflows of this kind would involve template-specific extraction rules. According to the CEO of American Medical Depot, Sukrit Agrawal (Gupta & Dutta, 2011), a distributor of medical and surgical supplies, his company has close to 500 invoices to be paid at any given moment. In addition, they continually have the same number of accounts receivables. An optimised model to schedule payments would be a huge benefit to this company and many others. As such, the application of text analysis and data extraction can also be applied in the manufacturing and service sector for improving operations.

Today, Big Data is beginning to show up in several sectors, including medicine, retail, manufacturing, and research. Through the internet, digital devices, internet-based applications, and search indexing, Big Data is always nearby. This creation of Big Data through internet search indexing and social computing creates many unforeseen

challenges for traditional analytics (Acharjya & Ahmed, 2016). These kinds of challenges occur because the existing algorithms for analysis may not give a response in a satisfactory amount of time as the system is dealing with immense amounts of dimensional data. Technically, the area of concern is the clustering part which is an important area when dealing with data analysis (Huang, 1997). Therefore, the major challenge is in designing the storage systems and developing an analytical tool that generates outputs perfectly, regardless of where the data originated and in what format (structured or unstructured). This thesis aims to provide a foundation for further academic progress in this regard in addition to concrete or practical progress in invoice data extraction.

Before starting any data analysis, the input data must be well structured. Big Data represents a massive body of unstructured or semi-structured data, and this makes analysis, representation, and access challenging. To overcome these challenges, it is necessary for the data quality to be improved through pre-processing. Big Data is available from many heterogeneous resources, and, just as in the real world, the data sets are inconsistent, noisy, and incomplete. Therefore, the techniques for cleaning the data, integrating it, transforming it, and reducing noise should be applied (Khan et al., 2014). However, the problems do not end there. Different challenges occur in the sub-processes, too, concerning the data-driven application. The confidentiality challenge is one of them. Other challenges involved in Big Data analytics are data security, inconsistency, instability and incompleteness (Labrinidis & Jagadish, 2012; Khan et al., 2014). Without sufficient security, the smart operation is not practical because smart operations can face major security issues when compared to traditional internet-based applications (Qiu et al., 2012; Wang et al., 2016). This thesis is partially motivated by the need to develop systems for improving image and data quality for the benefit of business operations.

The combined effect of supply chain leadership and governance mechanism affects both supply chain structure and supply chain learning, and MNCs change their supply chain structure to facilitate supply chain (Jia et al., 2019). Therefore, one of the next analyses in the supply chain after invoices and expenses that are well documented in the system is to perform supplier identification and classification. With this, the business wants to understand the spend category and take better decisions in supplier selection and cost optimisation. There is a huge demand in the industry to optimize the cost in the supplier procurement system. Every business wants to reduce the cost by further analysing the

expenses in the lowest specific categories. This will help them decide in everyday business related to supplier selection or spend optimization on a real-time basis. In the following section, the research objectives which arise from the motivations for this thesis are presented in detail.

1.3. Research Objectives

The thesis aims to address the following research objectives:

1. Study existing methods:
    - 1 (a). To determine the scope and depth of relevant source material. Since invoice management, SCM, and machine learning are interdisciplinary topics, articles which warrant discussion are published in a wide variety of journals.
    - 1 (b). To investigate the complexity of data extraction from scanned bills.
    - 1 (c). To identify the problems and challenges that currently exist in the industry.
    - 1 (d). To explore existing methods for automated data extraction.
    - 1 (e). To identify the non-machine learning approaches taken along with the issues and challenges that face them.
    - 1 (f). To perform an industry survey to identify the importance and impact of automated data extraction.
2. Design a novel data extraction method:
    - 2 (a). To apply various machine learning and pattern recognition techniques to increase extraction accuracy.
    - 2 (b). To develop a novel method that performs well on generic invoices.
    - 2 (c). To identify the impact of automated data extraction on SCM optimisation.
    - 2 (d). To validate the model's accuracy, cost, and speed.
3. Enhance error correction:
    - 3 (a). To study and investigate the role of text correction and enhancement in the decision-making process of the system.
    - 3 (b). To identify and apply the best methodology for invoice data extraction.
    - 3 (c). To validate the model's accuracy, cost, and speed.
4. Application significance:
    - 4 (a). To evaluate the significance of supplier selection and spend classification.
    - 4 (b). To evaluate additional application areas.

4 (c). To develop a self-adaptive system for any kind of invoice across various industries.

The overall objective will be to increase the efficiency of the system, reduce human intervention and minimize business costs. This research is carried out on the inductive approach to invoice text extraction. The research strategies include the solution methods for the data extraction and spell check enhancement. Furthermore, the survey feedback presents the challenges faced by stakeholders at the time of using existing text extraction tools/ software. For the analysis of the experimental results, three distinct types of datasets have been studied in this thesis: two academic datasets of invoices/receipts and a real-time company dataset. The proposed method uses both qualitative and quantitative methodology and the research method framework considered is empirical in nature.

1.4. Research Philosophy

Effectively synthesising new information with the theoretical foundations of the field requires an understanding of research paradigms and methodologies. In other words, to select the appropriate research framework for a given area of study or research objective, we must first understand the various research philosophies. Therefore, we will begin with a brief overview of the various research philosophies.

Research philosophy is the study of the theoretical basis for research and the acquisition of knowledge. Stated in more general terms, research philosophy seeks to understand the source, nature, and development of knowledge (Bajpai, 2011). According to (Kuhn, 1962), a ‘paradigm’ is a set of shared structures, agreements, and beliefs among researchers used to understand problems and research within the scientific community. Paradigms help establish a standard for research that allows for more consistent and high-quality effective research across a given field. Scientific research philosophy provides a foundation for research and involves choices regarding things like research approach, information gathering and handling methods, the method for exploration, and the formulating of a research question. The exemplar of the scientific study, sequentially, comprises of epistemology methodology, ontology, and methods. (Symon & Cassell, 2012) state that ontology, epistemology, and methodology are the three main philosophical schools that form the foundation of any research. Furthermore, they argue

that a research method can be determined by selecting the correct research paradigm in order to present the required information, problems, challenges and solutions.

The methodological choice must be connected to the philosophical position of the investigator and the scrutinized social science spectacle. Several philosophical schools may be relevant to the field of research simultaneously; however, certain extreme methodologies might be mutually exclusive. Intermediary philosophical approaches permit the researcher to merge their approach, philosophy, and research topic in a symbiotic relationship. Research philosophy can thus be described as the development of research theory through its relationships with other aspects of the scientific process and its characteristics, such as strengths and weaknesses. The theory or assumption is discerned as an initial account of reasoning, although it is based on the philosophizing individuals understanding and perception that are born because of intellectual activity. Since research stems from assumptions, different research methodologies may have different definitions of key concepts such as the nature of truth, understanding, and knowledge.

Furthermore, (Saunders et al., 2019) states that every researcher tends to adopt views and assumptions in line with the way that they perceive the world. This, in turn, is entirely based on their discipline and experience. Therefore, a brief overview of what research philosophy is and how it can be applied in an appropriate manner is required to produce an effective and valid thesis. According to (Crotty, 1998), “any ontological situation requires epistemological position. An ontology consists of two contrasting standpoints which are constructivism/ subjectivism and objectivism”. “The concept of constructivism has influenced a number of disciplines, including psychology, sociology, education and the history of science” (Eddy, 2004). It helps the researcher “construct beliefs and interpret data based on that belief, within the framework of cultural, historical and social context” (Bryman, 2015). On the other hand, an “objectivist create single realities through the understanding of sequential procedural investigation that produces knowledge as a repertoire of actions in response to specific environmental stimuli” (Kundi & Nawaz, 2010). “An objectivist educator believes that there is one true and correct reality, which we can come to know following the objective methods of science” (Vrasidas, 2000).

Philosophy serves as a guide for research. Its key components are epistemology and ontology. Ontology involves the philosophy of reality or deals with nature, whereas epistemology is the philosophy of knowledge, or how people identify truth. The

philosophical paradigm of research refers to practices and principles that govern review within the discipline by outlining and determining the processes, methods, and lenses through which the study is conducted. In many ways, research philosophy leads the researcher to their research question, helps them plan to investigate the problem, chooses a research design, and identifies the methods used in collecting, analysing, and interpreting data. Therefore, the researcher must be transparent regarding the paradigm that influenced the study because it will allow them to assemble a review and chose their research tactic. The researcher does not always state the philosophical foundation on which the study is based; however, the foundation can be identified by locating the research questions, cautiously analysing the literature review, examining the researcher's methods, and considering the determination of the study. The research approaches are nominated to systematically assist the study plan, assemble data, and explore information.

In addition, there are various levels of understanding regarding how this research contributes to current research practices. The main objective of the research is to further reduce the time required to convert scanned financial documents into digital formats for storage and analysis by building a new system on the foundations of current management theories. The study also suggested how theoretical concepts can be practically implemented to advance document data extraction techniques, improve supply chain management, and validate the theory. As such, the theoretical contribution includes doing a literature review on the existing study, testing existing methods applied for invoice data extraction and further extending the theoretical framework for better data extraction. The methodological contribution includes an in-depth interview taken to identify problems and challenges and the gap in the existing literature that was missing. Method related to object detection, table level extraction and spell check for invoice data extraction were extended in line with few other methods that are discussed in detail in the following chapters. Further, from an empirical contribution point of view, the newly developed system was tested with the existing system and proved to present an increase in efficiency and accuracy of invoice data extraction. Finally, this thesis provides a vast managerial contribution in the form of a better tool, cost optimization, reduction in human resources and much more refined spend analysis.

This section will also describe the epistemology and ontological point of this thesis by describing the following points defined by ‘layers’ in the ‘research onion’ diagram from (Saunders et al., 2019): Philosophy, approach to theory development, methodological

choice, strategy(ies), time horizon, techniques and procedures. It is important to understand each 'layer' so that it can be determined which characteristics and methodologies this thesis exhibits. To begin, the outer layer deals with the **research philosophy**. The five schools are positivism, critical realism, interpretivism, postmodernism and pragmatism.

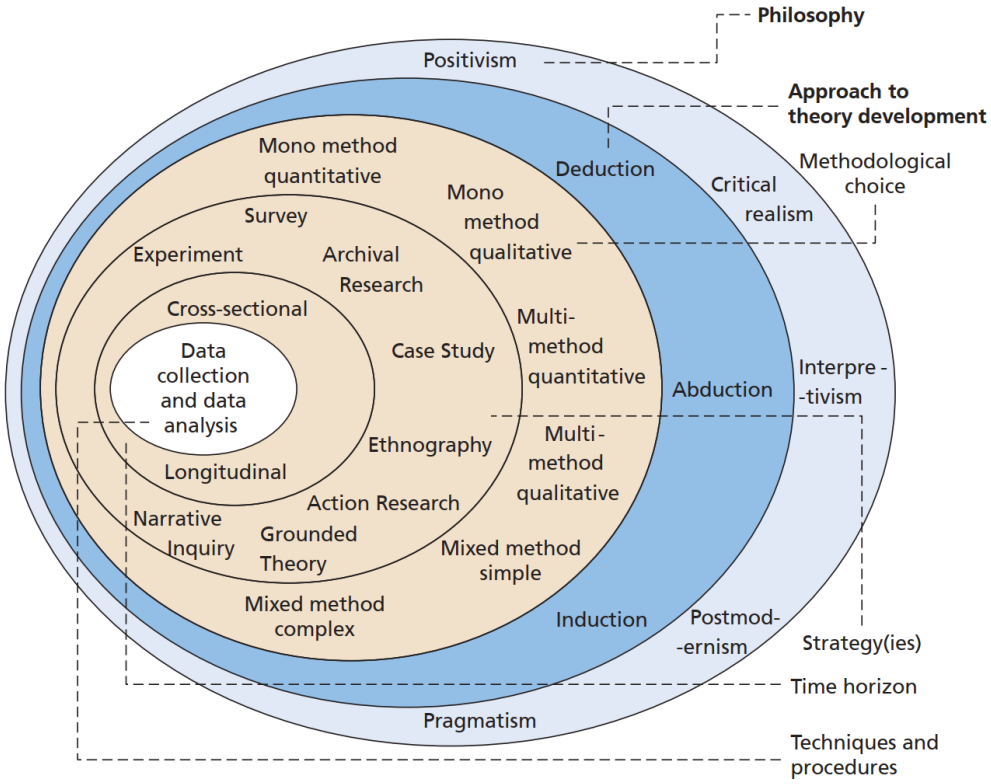


Figure 3: The 'research onion'. Source (Saunders et al., 2019)

A. Research Philosophy

1. Positivism

Positivism deals with a study that is conducted with knowledge based on facts that are gained through observation and measurement (Saunders et al., 2019). In the positivist school of thought, data is the benchmark of truth. It fills knowledge gaps and produces analysis and results through statistical analysis. It acknowledges that the study of social reality only exists objectively, independent of the research (Bryman, 2015). Positivist research philosophy asserts that social work can be recognized in an objective manner. The researcher is an objective analyst in this research philosophy, separating himself from personal values and working autonomously. Positivist research philosophy supports the

idea that only knowledge acquired through direct observation is reliable. In positivist research philosophy, the role of the scientist is restricted to information gathering and data interpretation in an objective way. In positivist, research findings are typically observable and quantifiable. The quantifiable results upon which positivism depends can then be used in statistical analysis and interpretation. Positivism exists within the empiricist framework, which holds that all knowledge must emanate from human experiences of integration with various environments. Positivist research philosophy has an atomistic, ontological perception of the world, which consists of detached, visible features and proceedings that intermingle in a recognizable, indomitable, and regular manner. The researcher should be fully independent of the study, according to the positivist approach, and there should be no personal bias involved in the study's execution. It has been argued that positivist studies mainly adopt the deductive method, whereas the inductive research method is generally related to a phenomenological philosophy. Positivism affirms that researchers must focus on facts, while phenomenology is concerned with implications and human interest.

Now, based on the analysis done from a philosophical research point of view and finding where the research problem fits, it seems that this study is based on positivism, and this research philosophy fits well into the area of a study done in the thesis. This is because data is based on factual, quantifiable information collected over time. (Bryman, 2015) stated that, irrespective of the research philosophy applied, social reality is objectively defined. This is the positivist approach. Positivism is linked to quantitative research and involves the deductive approach (Saunders et al., 2019). Thus, data needs to be analysed statistically to verify or falsify the theory. Data is collected and interpreted through an objective approach in which results are both observable and quantifiable. This approach considers all existing methods in the formulation of a new method with the aim of refining the answer. The methods can also be combined during the advancement of research. Furthermore, using the quantitative methodology and experimental strategy, the data collection and analysis is done.

2. Critical realism

Critical realism deals with the study based on how the world works and what can be proven based on observation. In other words, empirical investigation. This means the development of knowledge-based on the scientific method. On the other hand, critical realism deals with the study that brings us to empirical investigation. It positions itself as

a substitute paradigm to scientific forms of positivism concerned with predictabilities, regression-based variables models and to the strong postmodern turn that denied enlightenment in courtesy of elucidation, with an emphasis on hermeneutics and explanation at the cost of interconnection. Critical realism is depending on the world work and any data that can be confirmed from the observations. Critical realism is difficult to define since it is not a methodology, it is not an empirical program, and it cannot be considered as a true theory since it explains nothing. It can be considered as a meta-theoretical position: it is an automatic philosophical bearing concerned with providing a philosophically informed account of science and social science that can also inform our empirical investigations.

Critical realism is all about ontology or the analysis of the nature of things. Ontological realism stresses that truth exists irrespective of our knowledge of it. Reality is considered to have not sufficiently responded to hermeneutical examination or empirical survey. Generally, social science seeks to ground itself in empirical investigations by paying attention to epistemology at the expense of ontology. This suggests that sociology is interested in how we attained the knowledge which we possess, while questions relating to the substance of our knowledge are largely ignored. The findings have been a focus on approaches and methods of explanation, with inadequate consideration to the question about what type of entities are truly present in the social world and what they resemble.

Ontological realism is devoted to the reasonably independent existence of social realism and our inquiries into the nature of reality, although or understanding about concerning that reality is always socially, culturally, and historically situated. Understanding is articulated from several standpoints conferring to different influences and interests and is transformed by human activity: our knowledge is a concept, setting, and action dependent. Critical realists believe that they cannot be naive in relation to this and must clinch a form of epistemic relativism. Realism entails an obligation to truth; there are no measures of sagacity that happen outside of the ancient period. Due to this, all the representations and specific perspectives have restrictions.

Science is imperfect and scientific understanding is continually expressed in terms of theoretical structures, which cannot be considered as distinctive ways of analysing the empirical world. The reality of things is achievable in different ways, and the depth of intuition normally arises at a cost extent of the scope and the other way round. This suggests that our representations of the world are always fallible, perspectival and

historical and entails the necessity of methodological heterogeneity and does not propose that acquaintance is despairing or the likelihood of realism is an ineffectual quest. For critical realists, the ontology should basically be understood as a comparative degree of independence from epistemology and elucidation.

Critics of critical realism have attacked the strong emphasis on ontological realism. There is something to the critics insofar as sturdy dashes of realism may exceed their limits at the social world concept-dependence expenditure, but the risks are unimportant. Therefore, it is concluded that this research cannot be defined under "critical realism". The ontological belief is not stratified in this research and even the data collected is not historically analysed.

3. Interpretivist

Interpretivist studies attempt to integrate human interest and opinion with research practices. They require the researcher to consider a qualitative approach by interacting with participants through interviews, thus achieving deeper insights into the research objectives (Paul D. Leedy, 2010). "Interpretive researchers assume that access to reality (given or socially constructed) is only through social constructions such as language, consciousness, shared meanings, and instruments" (Myers, 2009). According to (Burrell & Morgan, 2008), interpretivism intends to "explore feelings, emotions and values to understand the subjective experience of individuals". Interpretivism seeks to construct meaning and interpretation based on their social perspective. The data collected for this thesis is more of objective rather than subjective in nature. It does not include any details of feeling, emotions, or values. It is not interpreted differently. Therefore, this study does not fall under "interpretivism" philosophies. From an ontological perspective, too, the data does not depend on social construct through culture and languages.

4. Postmodernism

In broad terms, postmodernism is a sceptical approach to philosophy, art, and criticism that arose as a retreat from the tenets of modernism that precede it. Postmodernism overlaps with research philosophy through two important premises. The first of these is that there is no objective reality, and therefore no objective truth. Postmodernists believe that no information or data can be extracted from our environments without imprinting it with biases, alterations, or assumptions. Our cultures, opinions, methods, beliefs, environments, experiences, and identities all leave imperceptible blemishes on our

observations, robbing them of their objectivity. Postmodernists believe the scientific pursuit of objective truth to be nothing more than ‘naive realism’. The second premise is that that science should be viewed with suspicion because it has, at least historically, been used as a tool of subjugation. Postmodernists point to past scientific pursuits like phrenology and the science of race as examples of science as a tool used for political and oppressive purposes.

Postmodernism is often called idealism, conventionalism, relativism, social constructivism or interpretivism. The primary postmodernist contribution to science and research has been the inclusion of elements of discourse and subjectivity in the scientific process. Postmodernists use the notion of rhetoric, language or discourse as stating to instruments used to construct the world. They typically overlook critical realism and often conflate sociology of science with philosophy of science as if social relations among and within scientific communities determine the etiological, methodological, epistemological, and ontological assumptions of researchers and scholars.

The ontology of postmodernism is hard to ascertain because postmodernists are generally wary of making definitive claims about the nature of the world. However, the majority of postmodernists acknowledge the significance of discourse in its subjective construction for each individual. Some postmodernists take this claim as far as saying that the world is entirely created by the humans’ mind, which is experiencing it. However, a few postmodernists are willing to accept the existence of extra-discursive occurrences.

Postmodernism opposes that the complete truth is both unachievable and meaningless. On the other hand, the knowledge claim truth is relative to the level that goes down to simple convection; accuracy is a matter of collective consensus, negotiation and agreement hence never being complete. Scepticism is likely to rule despite the truth of every knowledge theory being relative to respective adherents or proponents. Therefore, the existing research work in this thesis does not fall into the “postmodernism” research philosophy. The ontological belief of this thesis does not depend on “social construction through power and relations”. It is not interpreted differently. The data in the scanned invoice documents is the reality, and only that needs to be extracted. Further, no qualitative approach is presented for the accuracy extraction and improvement. Data collected is quantifiable and comparable for accuracy enhancement.

5. Pragmatists

Pragmatists believe that the world can be interpreted in many different ways, that no single point of view can ever provide the entire picture, and that there may be multiple realities based equally in truth (Saunders et al., 2019). Unlike positivism and interpretivism, pragmatism can provide synthesis with more than one research approach or strategy within a single study (Wilson, 2010). Again, it is seen the ontological belief of this research does not fit well with the "pragmatists" research philosophical point of view. It is to be noted that although the study might be interpreted as a mixed-method due to the inclusion of survey/interview. The survey was only performed to back the research gaps. And the data collection for this research is purely based on quantitative approach.

B. Research Approach

In the second layer, we find the **research approach** section. There are three possible choices when it comes to approaches: induction, deduction, and abduction. In **inductive** reasoning, we first observe a phenomenon and then make attempts to generalise it into a theory. This is done by first observing an event or a phenomenon, observing a pattern, and then analysing that pattern. Later, a possible hypothesis is created, and then, finally, a theory may be generalised.

In contrast, with **deductive reasoning,** we move from the general to the specific in a top-down approach. First, we formulate a theory about a topic we want to study. We then develop a hypothesis. Later, after data collection, we evaluate our hypothesis for explanatory power and use the evaluation to inform an overarching theory. In contrast to the inductive approach, the researcher starts from a theory that is persuasive and goes ahead to test its insinuations with available information and data. People are associated with scientific investigation in the deductive approach and verify information from the theory, which is general to data which is more specific. The researcher's main objective is to study the theories suggested by others and to use this understanding to developed related, testable hypotheses. The first step is to think about a theory of interest and then get to the hypothesis and observations, which eventually give a confirmation. Specific data is used to test the hypothesis, which either to supports or casts doubt on the original theory. In other words, it is the reverse of the inductive approach.

Finally, **abductive reasoning** is based on logical inference, which seeks to identify the most likely explanation or cause for an event. It is useful for making accurate predictions in the absence of complete information. In general, it is based on incomplete observation,

which seeks to determine the possible prediction. The abductive approach is different from other approaches in that it involves making a conclusion from known information. It gives its best possible explanation using an incomplete set of observations. Unlike in the cogent inductive reasoning that is dependent on complete evidence either being positive or negative, the abductive approach is characterized by partial elucidation, evidence, or some combination of both.

The research approach taken in this thesis is based on deductive reasoning, where the theory was formulated, a hypothesis was created related to the accuracy of data extraction and impact on SCM, and finally, the data was collected to run an experiment and substantiate the theory.

C. Research Strategies

The third layer addresses the **research strategies** which one should consider based on the approach selected in the outer layer. The three main methodologies are **quantitative**, **qualitative**, and **mixed** (both quantitative and qualitative are mixed here). According to (Saunders et al., 2019), "quantitative research is concerned with positivism with importance on deductive approach to test hypothesis, whereas, qualitative research is usually linked with interpretivism to derive theory through the use of the inductive approach (Denzin & Lincoln, 2011; Ritchie & Lewis, 2013). The qualitative research philosophy is naturalistic, humanistic, and interpretive, thus places significant significance on subjectivity. The assumption of ontological is that there is no distinct reality but involves several facts for any phenomenon. Furthermore, every person experience, interpret and perceive a phenomenon or situation of interest since every person has diverse reality experiences. The epistemological assumption is that knowledge is established from subjective observation at the level of in-depth understanding and detailed description.

This thesis follows quantitative research strategies. The quantitative study is developed from the positivist paradigm. Considerable value is placed on control, prediction, objectivity, and rationality. The assumption of ontology is that there is one reality that subsists and can be confirmed through the senses. The epistemological assumption is that knowledge can be explored and defined through a cautious dissection of the phenomenon of interest. There is a belief by researchers that all human behaviour is measurable, purposeful, and objective. It includes the study of hypotheses or research questions that

identify the concept's characteristic and prevalence, evaluate the effect, and cause relationship between test and variable for intervention effectiveness and test the connection. The researcher wants to develop or find the tool or instrument to measure the concerning phenomenon. At the same time, they remain separated from the study to prevent personal biases and values from impacting the study results. Numerical data collection drives the research than it is exposed to statistical analysis.

D. Time Horizon

The next inner layer discusses the time horizon. This defines whether the data was collected at a given point of time or over a period of days, weeks, or even years. The time period of the data has implications regarding study design and approach. The time period is either longitudinal or cross-sectional. In longitudinal, the study is done by comparing the result of two or more time periods. Whereas in cross-sectional, data is collected from the given time period, and analysis is done. For instance, in this study, the data was taken for a given time period and therefore utilized a cross-sectional approach.

E. Data Collection

The final layer discusses the strategies involved with data collection. Strategy selection is linked to the selection of a research approach and methodology. Some data collection, usually done by government agencies on a larger population, is based on surveys. These data are then used for analysis which is manipulated to form different experiments and, as such, recommends empirical research. At the same time, other data collection is based on sampling, observation, interviews, questionnaires, and secondary forms of data.

This research is based on data collected from various companies and stored in the system. Therefore, an objectivist viewpoint will be considered to fulfil the aims and objectives of this thesis. In this study, interviews were only conducted to guide the literature review process by identifying gaps from the point of view of industry stakeholders. Otherwise, the data was collected from companies in the form of secondary data. These specific data were used for empirical framework design.

1.5. Research Methods

This systematic literature review aims to outline the myriad studies, projects, and concepts that form the theoretical foundation of this study and the field in general.

Traditional literature reviews differ substantially from systematic reviews. The primary difference, according to (Rousseau et al., 2008), is their representativeness. While traditional literature reviews tend to "cherry pick" studies, systematic reviews strive to provide an exhaustive overview of research undertaken on a specific area. There are three overarching goals that guide this literature review. The first is to ground the thesis within a relevant and complete body of literature. The second is to identify the gaps within this body of research that the thesis seeks to address. Finally, the third is to identify and explain keywords and concepts. In pursuit of these goals, the literature review will also begin by presenting inclusion and exclusion criteria its methodologies for locating data and identifying keywords. This literature review presents a wide array of studies from several related fields. Studies selected for inclusion in the review come from a wide variety of fields and journals; however, they are unified by their relevance to the practicalities of invoice data extraction and, thus, to the scope of this thesis. The discussion below will initially focus on digitisation and text extraction in a general sense before narrowing its focus to identify gaps and objectives in the current body of research on invoice data extraction methodologies and technologies.

To achieve the objective of this research, a systematic literature review was performed. (Tranfield et al., 2003) suggested a three-stage process of planning, conducting, and reporting. The planning phase was used to determine the scope of the research. Together with the research guide, the topic was slowly developed through discussions with industry representatives on the nature of problems found in the industry. It was determined that research should focus on problems related to extracting text from invoices and the significance of this process in SCM. It was also decided that the gaps needed to be backed by interviews conducted with industry stakeholders. Reviews of the extraction of non-text data or logo related data will be excluded from this review due to lack of relevancy. The second stage defined the formulation of search terms on the basis of the discussions in the first stage of development. This process is called the development of the search strategy and includes all the steps taken to discover literature that is relevant to the research question. A good search strategy is crucial to the review's success and the accuracy of its findings (Bettany-Saltikov, 2010).

The primary aim of this research is to determine the breadth and depth of source material related to invoice management systems and SCM, as well as the future scope of these concepts. Since OCR, invoice management, SCM, and machine learning are

interdisciplinary topics, relevant articles originate in a wide variety of journals, i.e., computer science and management, that wide variety of references were selected from proceedings, journals that included Science Direct, IEEE, Goole Scholar, ACM, Scopus and Springer. Further, in order to generate a more thorough perspective, it is often regarded as important to incorporate grey literature in the study (Tranfield et al., 2003). Therefore, references from an online website, authenticated blogs, references from businesses and organizations in electronic and print formats were also considered, which are not published in academic terms. Additionally, exploring various methodologies already applied in automated data extraction systems and research to identify non-machine learning approaches taken, issues and challenges found and to apply various machine learning and pattern recognition technique to increase accuracy.

Topics relevant to “information extraction”, “invoice data extraction”, “receipt data extraction”, “automated data extraction”, “spend classification”, “invoice supply chain”, “big data text extraction”, “Financial SCM, Invoice”, “OCR” with “CNN / RCNN / Bi-LSTM / GAN” were included in the search query. Any articles that were not related to OCR and specifically did not specifically talk about text extraction were excluded based on research title and abstract summarization. From the papers identified, while reading the papers, backtracking was also performed to gain more insights. Later, the papers were divided into different categories based on the area it was focused. The category was defined as supply chain (including spend, supplier and invoice role in finance), big data, written language and scripts, real scene image, non-invoice document, invoice related document and spell check related reference. The categories were initially divided into "conceptual" and "empirical," and then the "empirical" was further divided into "qualitative," "quantitative," and "design research."

Text extraction related reviews and gap analysis is defined in chapter 2. Chapter 2 also includes the survey interview related study performed to back the research gap specifically related to invoice data extraction. Relevant data were also identified from various ongoing studies, which represent the most recent research in the field and provide a comparative benchmark for analysis. Later, the solution framework was presented in chapter 3. Additionally, a spell check related literature review and solution methodology is presented separately in chapter 4. This was done due to the realisation that further improvements can be made to the solution framework. And also, that a vast subject area

was covered, it was difficult to incorporate and discuss all the research gaps in a single chapter.

1.6. Ethical Issue

As stated plainly, ethics is the study of morality or the attempt to determine right and wrong with the aim of living better lives and creating more just systems (Vadastreanu et al., 2015). Ethical discussions are frequently complicated by individual and societal variations in moral code. Every society defines its own beliefs, rules, and norms regarding the best way to live 'well'. Some moral codes are established on mutual understanding and experience, while others are based on religious beliefs. In society, we have governments and laws along with enforcement and justice systems to help us develop and maintain ethical societies and organizations. Of course, this differs to some extent based on individual interpretations of societal norms and between societies.

In research, ethical issues frequently differ from societal issues. In academics, the field of ethics tries to establish and enforce a standard for research and study that is deemed 'good'. All universities around the world have defined rules and regulations which researchers must follow. These are known as research ethics. This study will strictly adhere to these rules, regulations, and moral practices by complying with the recommendations of the university research committee. It will be submitted for ethical review at every required stage of the project to avoid conflict and misconduct.

There are several ethical theories that might be relevant to the process of conducting research such as this. These include utilitarianism, feminist ethics, social contract, deontological and virtue ethics. Considering the various ethical approaches, this thesis seems to adopt a mixed ethical approach based on utilitarianism and Kantian theory. In the area of research, it follows the Kantian principle of the categorical imperative, which views all subjects as ends in themselves rather than a means to an end. Participants surveyed were not surveyed for the extraction of information; they were surveyed in order to improve the data extraction systems upon which they rely.

1.7. Data Collection

There are multiple aspects of invoice data collection. Some are fully available, some are only partially available, and some are hidden due to privacy concerns. Currently, data has been collected from companies that manage invoices for multiple clients to provide automated invoice data extraction services. This dataset consists of more than forty thousand records from a leading accounting/financial company. The OCRed xml collected from the same sources will be used to make this new system robust. The research for this thesis was conducted within the framework of the Systems Science Department at the University of Hull and followed all ethical issues as defined by the university. However, the author cannot comment on the ethics of the data collected by companies and whether or not they compromised on any specific ethical standard. Applying virtue ethics will ensure that the privacy of data, from a company perspective, is well adhered to.

It must also be acknowledged that some of the data collected might be sensitive. Participants will be assured that their contributions will remain and that they will not be identified in any reports which utilize the information they have offered. As such, details like company names, individual names, and email addresses are not documented in this thesis. Furthermore, the information contained within some document images has been removed or redacted to safeguard the privacy of participants and third parties. Otherwise, in some cases, the publicly available google web search sample images are used.

Although the collected data are quantified, they are also collected based on qualitative factors. Non-numerical descriptive data, as well as nominal data, are examples of the qualitative data which was collected. This collection of data is gathered by performing a survey with three companies.

1. Collecting challenges, issues faced by the business organisation and especially related to invoice management, supplier selection and spend optimisation.
2. Analysis of behavioural patterns and the way they perceive data structure on the invoice. For this, agents involved in data entry and management were interviewed.

Data was collected by companies that manage invoices for numerous international clients by providing automated invoice data extraction services. This dataset consists of more than 40,000 records from a prominent accounting firm. The existing OCRed data collected from the same sources will be used to make the proposed system more robust

and to compare its accuracy to a standardized baseline. Furthermore, existing research related datasets such as the SROIE (Koo, 2016), (Busta et al., 2015), (Zhou et al., 2017a), (Shi et al., 2017), (Li et al., 2018), (Anand & Khan, 2020), (Zhao et al., 2020) dataset with 973 invoices and VATI (Ma & Hovy, 2016),(Katti et al., 2018), (Liu et al., 2019b), (Zhang et al., 2020a) dataset with 5000 invoice samples is used to validate and compare the performance of the result. The reason behind considering the industry dataset was to use the latest real-time data and to show how this new system will perform in the industry, The reasoning behind this is that some research-based datasets neglect crucial aspects of real-world applications. The details of the data collected and used are explained in chapter 2 after a thorough analysis of academic research and survey performed with company representatives.

1.8. Thesis Outline

This thesis is divided into three overarching parts. In the first part, an introduction to the theoretical background of the project is provided, along with a discussion of the concepts involved and an outline of the research objectives. In the second part, an overview of the various problems facing the existing system of data extraction is discussed through a review of the relevant literature and a survey of available data. Finally, in the third part, a novel approach using machine learning, rule-based pattern recognition, and enhanced spell-checking methods is proposed. Below summary table highlights the research gaps extracted from the literature view in chapter 2 and chapter 4. These are the main problem and challenges identified that were missing in the prior research done in this area.

Table 2: Table with a gap of LR what was done and in which chapter it was addressed.
"Devised by the author".

Problem Identified	**Prior Research Papers**	**Contribution In**
Image Quality Issue	(Köppen et al., 1998), (Zhang et al., 2009), (Liu et al., 2016b), (Jacobs et al., 2005) , (Rabbi et al., 2020), (Pegu et al., 2021), (Zhu et al., 2019)	Chapter 3 section 3.4
Data Block Combining Problem	(Jain et al., 2017), (Rahal et al., 2016b), (de Jager & Nel, 2019).	Chapter 3
Line / Table Item Extraction	(Köppen et al., 1998), (Medvet et al., 2011), (Aslan et al., 2015), (Gilani et al., 2017),	Chapter 4 section 4.5

	(Liu et al., 2020), (Reza et al., 2019), (Krieger et al.)	
Printing Position or Short Form of Words	(Medvet et al., 2011), (Jun et al., 2019), (Ho & Nagy, 2000).	Chapter 3 section 3.4
Multiple Similar Keywords	(Dave et al., 2020), (Chiang & Knoblock, 2011), (Grönlund & Johansson, 2019).	Chapter 4 section 4.5
Top-Down versus Left Right Matching of Key-Value	(de Jager & Nel, 2019), (Cinti et al., 2020).	Chapter 4 section 4.5

The outline of the content of each chapter in this thesis is summarized in the following figure and further explained below:

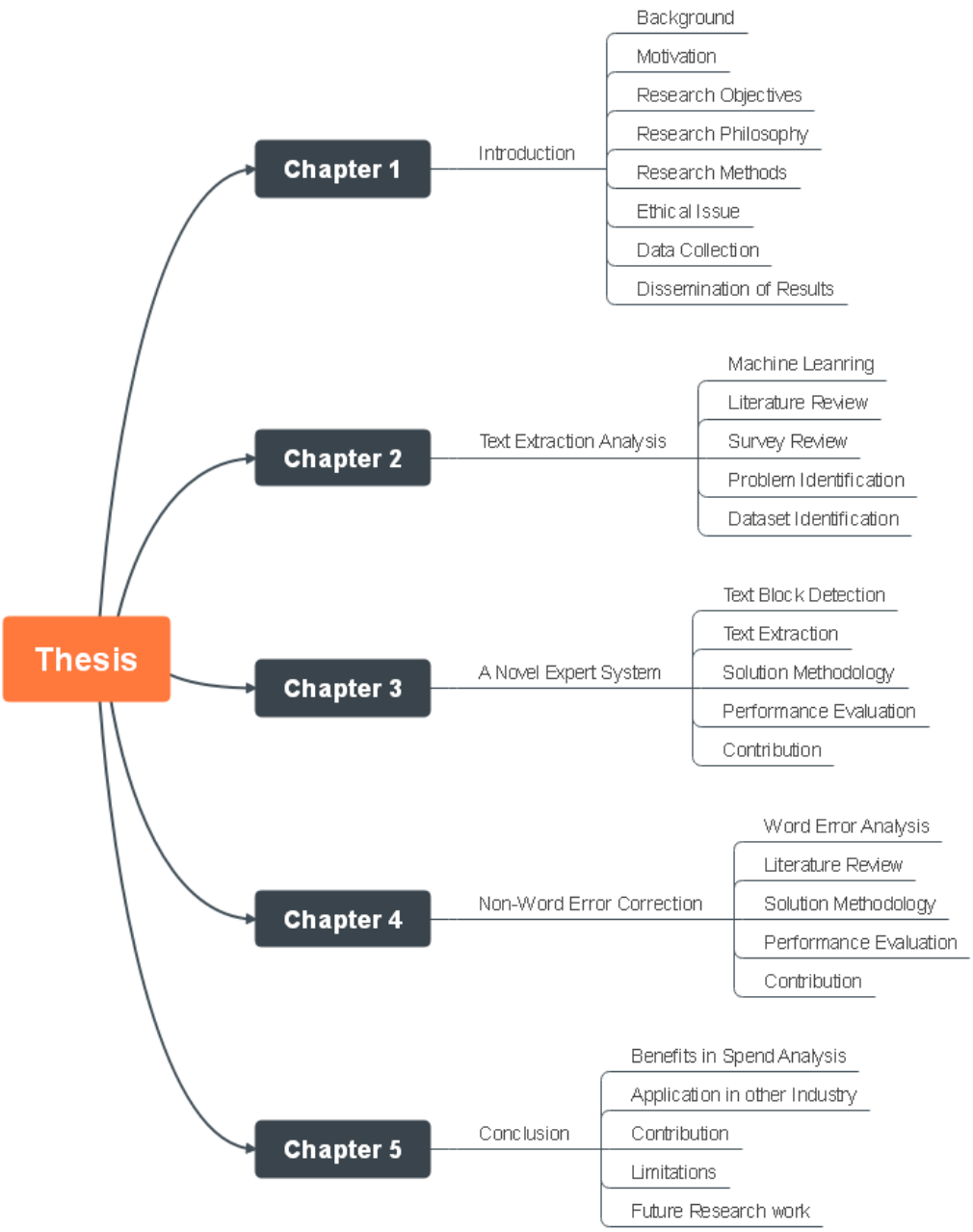


Figure 4: Thesis Layout.

"Devised by the author".

Chapter 1 summarizes the thesis, explains the research objectives, provides background information about the topic and the motivation for the project. It also explains the data collection and research methodology.

Chapter 2 provides information about OCR and machine learning, and later, a review of the literature relating to OCR and text detection and extraction as they relate to invoice data extraction to identify limitations and challenges. A survey review is conducted using a Google Form. Based on the response and the suggestions related to real-time usage, further gaps are identified. In the end, the study of the existing dataset, such as real-time

company data in the form of invoice/receipts, SROIE and VATI, is defined for use in the comparative analysis. The table in the Data Identification section provides all the required information about the dataset. The first set of the research objective that is from 1(a) to 1(f) achieved by doing the necessary literature review of the existing text extraction methods. The survey review performed and identification of gaps and challenges are identified for further research work.

In *chapter 3*, the text detection and extraction methodology are explained. The 'labelImg' software tool is used for training purposes. With the help of the labelImg tool, the fields are tagged with a rectangular box using descriptive monikers like "address", "date", "time", and "total". Each rectangular box has unique coordinates. The labelImg tool then provides an XML output format. To detect the content of tagged boxes, machine learning algorithms based on Faster R-CNN and GAN are tested. Ultimately, the GAN-based model is finalized. Based on the limitations identified in the previous chapter, a solution framework for the unique rule-based engine is explained in the solution methodology. This framework consists of a machine learning model and advanced regular expression, pattern matching, nearest-neighbour matching, spell-checking, and OCR error correction methods. In the end, performance evaluation is performed against the company baseline model as well as the baseline derived from the dataset recommended through the literature review. Based on the evaluation, the results have proven to be better than that of existing methods. Additionally, the need for further improvement has been discussed. The second set of research objectives from 2(a) to 2(d) related to the finalization of the generic model for invoice field detection and the data extraction is accomplished here.

C*hapter 4* is devoted to spell-checking, where the limitations and problems identified in previous chapters are resolved to a reasonable extent. The literature review related to spell-checking is included in this chapter. Based on the literature and the work presented in previous chapters, significant gaps are identified. The analysis between the n-grams, Levenshtein Distance, BK tree, and Bi-LSTM based model is conducted to gain better selections for spelling correction. It is concluded that Bi-LSTM is a better approach for spelling correction and automatic word replacement because it can provide better substrings matches which the other two models fail to do. The third objective set from 3(a) to 3(c) defined in the research objective section above is fulfilled in this chapter, which is a further enhancement of data extraction.

Finally, in *chapter 5*, the conclusion, contribution, limitations, and future work are presented. All the work, limitations, and challenges are summarized, and the effectiveness of this novel approach is analysed. The analysis of the research objective as discussed in chapter 1, is also analysed to determine which objectives were fully achieved and which were not. Its applicability in other domains such as the banking and healthcare industries is also discussed. A separate section is also provided for the discussion of benefits in spend analysis. After that, suggestions for future areas of research are discussed. With this, the last research objective group from 4(a) to 4(c) is also fulfilled at the end of this chapter.

1.9. Dissemination of Results

The research described in this thesis has been disseminated by presenting in the conferences and applying the novel system created in the existing industry to validate the result in real-time. The various list of publications, conferences and applications in existing industry are as follows:

A. Conference Papers

a) SHARMA, V. & MISHRA, D. N. Using Big Data & Prediction Analysis (BDPA) in Effective Pricing Decisions. Norwich Business School Colloquium 2016, 18-Oct-2016 2016 Norwich, UK. (Sharma & Mishra, 2016) :

   This conference presentation discusses the importance of big data and predictive analysis in the effective pricing decision. This was an initial phase in the study of unstructured data and how better pricing decisions can be taken if more financial data can be made available from invoices and receipts.

b) SHARMA, V. & MISHRA, D. N. Impact of Automated Text Extraction from Invoices in Supply Chain Management. Prolog Conference 2018, 29-Jun-2018 2018 Hull, UK. (Sharma & Mishra, 2018) :

   This conference presentation discusses the importance of automatic data extraction from invoices, which is the main area of the thesis. In this conference, the first stage of the novel rule engine results was presented and how the system performed better than the industry baseline accuracy.

c) SHARMA, V. & MISHRA, D. N. Big Data Analytics for Smart Operations in Service or Manufacturing Sector using Invoice Automation System. FBLP PhD Colloquium 2019, 10-Jul-2019 2019 Hull, UK. (Sharma & Mishra, 2019) :

In this paper, the importance of big data and smart operation using automated data extraction specific to the manufacturing industry has been discussed. The paper is yet to be submitted for publication.

B. Survey and Industry Work

Often, the accuracy of OCR tools does not meet industry expectations, even when using the latest text extraction techniques. This even happens after some of the OCR service providers outline the usage of the latest research and techniques and how they compete in text extraction accuracy. Therefore, it was essential to verify the output of the thesis by taking data from companies that are in the business of invoice and receipt automation. In pursuit of this, various company data is collected from two different sources and a further analysis is done on another company in order to understand the details of spending classification and future forecasting on the decision-making process. The dataset is used successfully in the thesis as a comparative study of real-time, generic invoice and receipt automation.

1. **Company A**: The company is based in Ireland and has offices in London, Australia, Europe, and the USA. They provide an integrated service that automatically converts invoice and receipt data for their customers and feeds it into the accounting software system of their choice. The company uses the OmniPage OCR SDK tool for text extraction and then applies a rule-based engine to extract relevant data. The novel system created in this system is used and presented as a demo. A comparison to their existing system was made. The rule-based engine was appreciated and well-received by this company.
2. **Company B**: This company is based in India and provides accounting services to its clients. The company's services include managing the invoices and bills generated by their clients. Following a successful demo, the company has shown interest in integrating this novel solution system with its existing application.
3. **Company C**: This London-based company is interested in cost optimization and spend analysis in the supply chain procurement sector. Following a successful demo,

they have shown interest in adopting automated supplier classification and in-depth spend analysis. They have also provided feedback on optimization of cost due to better information extraction. They have even requested to do the demo on text extraction on legal documents that exist with their client who are into the banking business. The client has shown interest in the demo provided and would like to integrate it with the spend analysis system for a much deeper spend analysis.

2. Text Extraction Analysis

2.1. Introduction

Over the past decade, there has been an explosion of data in almost every sector. As such, the digitisation of data has become a necessity for businesses and organizations hoping to reduce manual effort, minimise errors, and reduce their carbon output. Additionally, the process of digitisation helps improve a company's efficiency when processing invoices, thus optimising its costs and resource allocation. Despite these benefits, converting images of text or data into digital formats has remained a challenging process. The digitisation process was initially performed manually, but the time-consuming, tedious, and detail-oriented nature of the work made the errors common. Later, progress in the field of automation allowed many industries to begin automating the digitisation process. However, these systems still required a good deal of human intervention to validate and verify data. Today, further progress in automation has allowed for near-total automation of the digitisation process for the documents and scanned images which have become established features of SCM. However, while text recognition from the image can be accomplished with several OCR technologies which are based on machine learning algorithms, the amount of relevant information extracted does not meet industry requirements. This is a problem, especially when we examine data extraction in the context of invoices and receipts. An invoice is a document given by a seller to a buyer that lists the amounts, prices, and specifications of the seller's goods and services (Reviso, 2020). Invoice processing is a tedious process when performed manually. A typical invoice will contain details about the buyer and the seller, the date of issue, the item, and occasionally, the seller's bank details. Clearly, invoices contain many kinds of information that allow for deeper data analytics, better forecasting, and smarter decision-making when adequately extracted.

Sectors that deal with invoice processing can track key business factors easily. This helps to provide better customer service, improve the efficiency of employees, and improve profitability. Calculating Accounts Payable (AP) and Accounts Receivables (ARs) manually is not only expensive and time-consuming, but it also has the potential to confuse managers, consumers, and vendors. Companies can avoid these detriments by moving to automatic or semi-automatic digitisation systems. The benefits of doing this include greater transparency, advanced data processing, better working capital, and

simpler monitoring (Babu, 2020). Two sample invoices are provided as reference in the figure below:

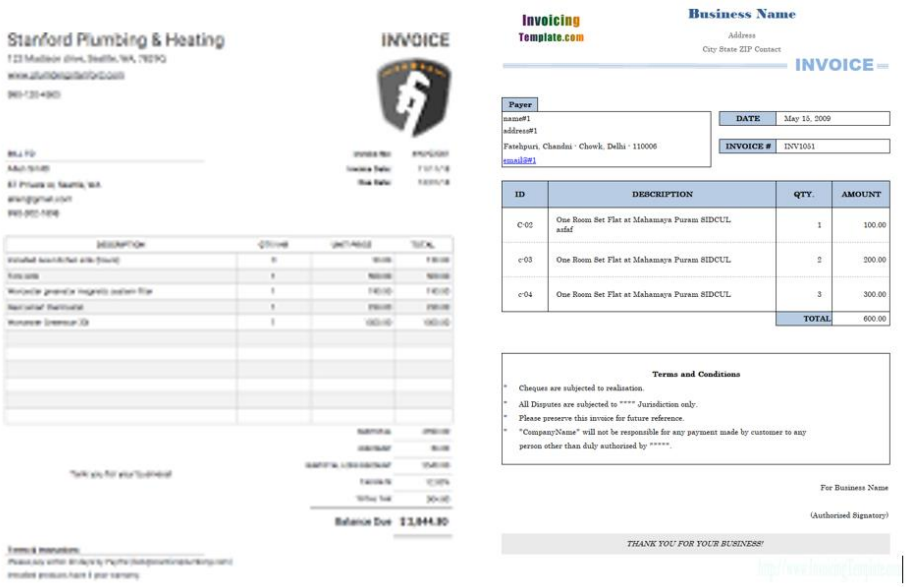


Figure 5: Sample invoice image.
"Devised by the author".

When the data needs to be extracted from an invoice similar to the example above, it passes through several distinct processes. The complete process of data extraction has been presented in the flow chart below. The highlighted box represents the main focus area of this thesis.

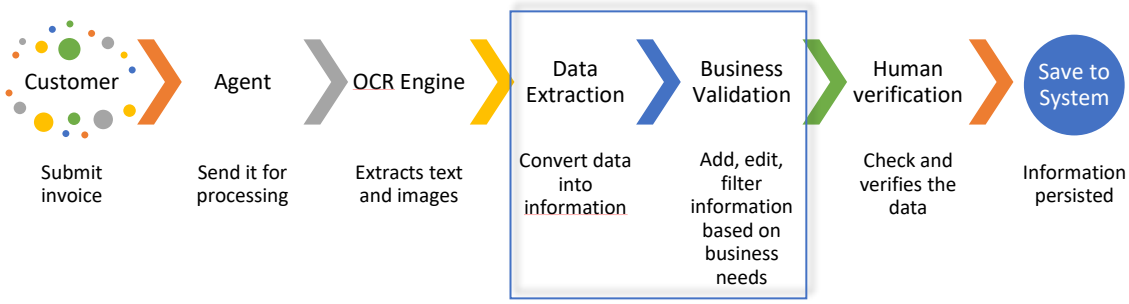


Figure 6: Invoice data extraction business process flow.
"Devised by the author".

First, a scanned or hard copy version of the invoice is submitted by the ***customer*** to the ***agent*** who is responsible for handling invoices and bills. The agent then sends the documents to an automated text extraction system if the company uses one. If they do not, the agent performs all remaining processes by hand. In the case of automated extraction, the ***OCR engine*** accepts the input in the form of images. If the input format is

a PDF, every single page in the PDF is extracted by the system a single image and then image data extraction is performed on each image individually. The scanned images are then sent to the ***OCR engine*** for text extraction. The OCR engine extracts and returns the text and its semantic in the form of XML or JSON files. XML or JSON file formats are internationally recognized file formats for maintaining data in textual form. It is used to transfer information between two or more external systems/services. This structured file with OCR text and its semantic information is then forwarded to the ***data extraction system***. Here, the OCRed textual content is analysed, and specific information is extracted (such as date, amount, vendor, items, and tax). Once extracted, the further structured extracted data is sent for ***business validation*** and then for human verification. The human makes respective changes and updates the system with correct data. Both the extracted as well as corrected data are maintained in the system for accurate comparison and analysis. It is also used for training the machine learning model in case the company uses such software.

Following this overview of the general process of invoice data extraction it is worth mentioning that the extraction process is split into text extraction using the OCR engine and the extraction of relevant information. These are two kinds of technologies, each with unique accuracies, limitations, and applications. The objective of this thesis is not to create a new, enhanced OCR engine but instead to rely on the existing OCR technology, which businesses frequently choose for text extraction tasks based on the performance, accuracy, or cost. Later, it will focus on identifying and correcting errors created by the existing OCR, such as structuring the data appropriately and understanding the relationship between data points in the invoice to extract as much information as possible.

Although this thesis depends on existing OCR software, there is still the need for an enhanced OCR model for data block detection, text extraction, and model training purpose. This will help the system make better decisions and will slowly make our system more robust and efficient. The newly developed, enhanced OCR model uses a machine learning model which replicates the OCR engine and gets trained slowly for better text extraction. This enhanced OCR model step lies before the data extraction step, as highlighted in the following figure.

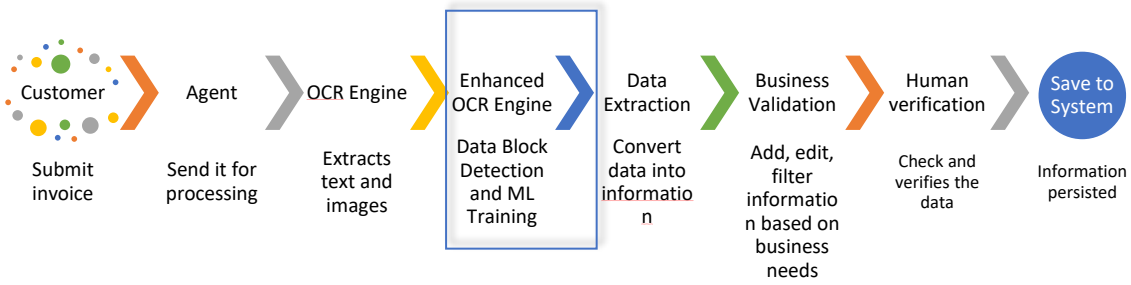


Figure 7: Invoice data extraction business process flow with enhanced OCR engine.
"Devised by the author".

This enhanced OCR has been designed using a GAN-based framework. The Faster R-CNN is used as the data extraction unit. The novelty of this research lies in the fact that the existing method, called TLGAN (Kim et al., 2020), identifies the table location but cannot extract any data. With the help of the GAN model, the image quality is enhanced by settings of the proper loss function. The Bi-LSTM model is used in spell checking to correct the extracted text even before the text is sent for data extraction. This is to ensure that the correct labelling of the data block is performed. As we are working on a large dataset, deep learning-based methods for data extraction and error correction are adopted. The study performed on this is explained in depth in chapters 3 and 4.

Therefore, we will understand OCR, machine learning methods, text extraction from OCR, processes required for better extraction, and the problems identified during this process. Interview(s) and survey will be performed with some businesses to identify challenges they might face in real life. Furthermore, in the later section of this chapter, missing gaps will be identified to resolve some of these challenges one at a time. OCR and machine learning techniques shall be used to develop an end-to-end model which can extract relevant information from scanned invoice-based images. Let us start by understanding existing technology and machine learning that can be used for data extraction.

2.2. Existing Technology

Machine Learning is a branch of Artificial Intelligence (AI) that enables computers to teach themselves without being explicitly programmed. Machine Learning allows systems to learn from data and, as a result, improve their performance automatically over time. In this system, computers apply various algorithms and use large data sets to "train"

their models. In this process, the systems teach themselves with learning and make further predictions. The process is based on learning and depends on finding patterns in the observable data contained within the training dataset. A test dataset is used to evaluate the model's performance by calculating various performance metrics. No observations that are derived from the training dataset are used for the test dataset. The reasoning behind this is as follows: if a test dataset contains examples from the training set, then it will become difficult to judge whether the algorithm was truly learning or simply memorizing. Machine learning techniques are applied in various fields for improved knowledge and training. Some applications include object detection and facial recognition. Text detection, speech recognition, and language translations are some of the popular applications which were possible only because of machine learning abilities.

Nowadays, almost every industrial OCR solution is based on a machine learning technique, with each technique performing better in some areas and worse in others. Many are open source, and many others are proprietary. Most of these OCR tools are generic and provide good text extraction accuracy. However, it has been demonstrated that the advertised accuracy rates are not met if the invoice structure is unfamiliar. It was evaluated that Amazon's Textract Engine achieves the best result. Textract provides generic text extraction, and invoice extraction features are offered as a separate service. Although the table data extraction capabilities are pretty remarkable, it struggles to extract 5-6 key-value fields properly. Below there is a list of some of the existing OCR tools and their respective comparison of accuracy.

Table 3: OCR application comparison.

"Devised by the author".

OCR Tool	Algorithm Used	Labels	Document Type	Dataset Format training	Accuracy	Drawbacks	Source link
Google OCR	CNN + Character Label Language Algo	All text in one label name as text and percentage according to Label	Support JPG, PNG, files	Image URL with text dimension in one CSV	80%		(Iddo, 2018)
Microsoft Azure	CNN	Extracted text line by line using text label. It generates an accuracy percentage of each line of text	Support JPG, PNG image files	Image files in a folder	Near about google OCR	In the case of photographs, it is unable to recognize the whole word because textual data is in the false positive category. As well as, when the photograph is without any textual information, the precision of model varies based on the type of image.	(Iddo, 2018)
Amazon Textract	Text Detection Algo	Extracted text one line by line using text label, Extract text as a form type using Key-value Pair, Extracted table data according to cell,	Support JPG, PNG, PDF files	They already trained on millions of Documents	90%	Amazon OCR API fails to recognize text from the table of bank fees properly: Textract failed to identify it as a table	(Amazon, 2020)

		percentage according to Label					
Sema Media Data	Currently Unavailable	Currently Unavailable	Currently Unavailable	Currently Unavailable	API not available	It cannot detect language automatically.	(Iddo, 2018)
Taggun	use Google or Microsoft API	Extract text as an Amount, Tax Amount, Date, Merchant, Address, percentage according to Label	Support JPG, PNG files	Not Available	82.5%	It is Not good for Scanned Invoice Documents.	(Iddo, 2018)
Cloudmersive	Microsoft API	Extracted text one line by line using text label percentage according to Label	Support JPG, PNG files	Not Available	80%	It is only helpful for Extract text from a book or paper full of text without a table.	(Iddo, 2018)
Expensify	Deep Reinforcement Learning	Extract text according to merchant name, amount, currency, a nd date label	Support JPG, PNG	Not available		It takes much time to generate and Extract data from the Invoice.	(Trepanier, 2019)
ABBYY	CNN + LSTM	Extract text according to a text label	Support pdf, jpg	Available	83%, according to Capgemini.	It recognizes field value based on position, so it is mandatory to create template formats for differently structured invoices	(ABBYY, 2020)

Rossum	Self-learning AI Neural Network	Extract text, table as a label user can add a custom label	Support PDF, JPG, PNG	Not available	98%		(Seguin, 2019)
Smart Receipt OCR	machine-learning algorithm	text	-	-	82%	1. The reader not scanning the correct information from the receipts. 2. Not able to fill in all the text from the receipt.	(Megan, 2018)
OmniPage SDK	Classification and optimisation algorithms are used; the exact algorithm name is not specified.	Data extraction, Sorting of different documents, layout analysis, business processes.	Support various format such as, pdf, html, bmp, gif, tiff, xml, .docx, xlsx, txt, rtf	-	100%	The major drawback of OmniPage SDK is related to the operating system. As the OmniPage has features of Window configuration, but when the user is using Linux, it is a tedious job. It gives also provide less accuracy when the documents have coloured or highlighted text/ background.	(OmniPage)

2.3. Literature Review

In this section, we will review the work done on OCR text extraction and, more specifically, invoice data extraction. The section has been divided into five subsections based on the area of study contained within. Fortunately, over the course of researching the topic, it became apparent that research analysis did not need to discuss invoice data extraction alone and that general text extraction research remained relevant. There are possibilities that a new methodology might have been created in some other area, but it is still related to text extraction and OCR. Therefore, we must study all these areas to find the latest work and identify research gaps. The first set of all research objectives that are from 1(a) to 1(f):

a) To determine the scope and relevant source material. Since invoice management, SCM and machine learning are interdisciplinary topics, relevant articles published in a wide variety of journals would need to be identified.
b) To investigate the complexity of data extraction from scanned bills.
c) To identify the problems and challenges that are existing in the current industry.
d) To explore various methods already applied for automated data extraction.
e) To identify the non-machine learning approach taken, issues and challenges found.
f) To further perform an industry survey to identify the importance of automated data extraction.") achieved in this chapter.

2.3.1. Written Languages and Scripts

OCR is a great tool for data extraction, but it is not always perfect. As with any system, there are certain limitations that must be accounted for or overcome. If the original document is clean (if it is laser printed, for example), OCR will typically read up to 97% of the words correctly. However, if the document contains images, special characters, or handwriting, the OCR's ability to read it suffers. If the original document is photocopied, faxed, or even printed by a dot matrix printer, the result is the same. Additionally, the OCR engine gets confused by lines and boxes because it is programmed to understand them as text features. (Khoddami & Behrad, 2010) performed script identification using the curvature space feature in which they tried to retrieve text from bilingual documents.

Their method was to identify scripts at the character level and generalise them into words, lines, and pages. However, a problem arose when small, common symbols were not removed in the pre-processing stage resulting in disconnection in Farsi fonts. (Singh et al., 2014) used a script recognizer for separating text and, later, a morphological method for segregating horizontal or vertical strokes was applied. Nevertheless, the threshold value was dependent on the classification and only valid for the machine printed documents.

Nowadays, digital collections are increasingly dependent upon various industry requirements. However, qualitative and quantitative research on the subject typically applies to many industries and applications. In digital libraries, a version of OCRed is used for indexing documents. (Chiron et al., 2017) researched the Gallica digital library from France National Library to find the error impact of OCR. This library consists of more than 100M OCRed documents and receives 80M annual search queries. Initially, they designed 12M characters in both English and French, and later they computed the OCR errors and submitted the queries to the Gallica portal over a period of four months. Their error indicator helped in identifying mismatched resources and observing the errors and identifying them in a large document set allowed for the enhancement of library services. Furthermore, statistics on OCR errors have been computed due to the novel alignment method introduced in this paper. They have taken the user-submitted queries from the Gallica portal over four months. It was then fed into the model to predict the risk. When the OCR error was present, it was mismatched with the query due to this OCR quality was underlining for accessing the data from the digital library. Their future work was concerned with the analysis of two approaches: query expansion and OCR post-correction using OCR error models.

Improvements to the processing capabilities of computers and to the availability of text recognition tools and technologies have led to the incorporation of computers in our day-to-day business activities. Many academic and commercial products have demonstrated the capability of computers to understand human language, be it printed, handwritten, or

scanned. With Natural Language Processing (NLP), computers can recognize and extract meaning from written and spoken forms of human language. The author proposed an automatic method for detecting and extracting Japanese alphabets within a manga comic page for offline and online real-time language translations (Arai & Tolle, 2011). The text was extracted vertically from balloons in images using blob extraction functions. The text was extracted from multiple constraints using OCR, which can further be used to translate in different languages. A similar approach was adopted by the authors for offline handwritten Telugu character extraction using OCR (Prameela et al., 2017). The proposed algorithm performed classifier feature extraction using both Support Vector Machine (SVM) and Quadratic Discriminate Classifier (QDA). On the same lines, OCR technology with object detection capabilities was applied to the post-examination data entry process (Rizvi et al., 2019). Controlling the post-examination process with an intelligent information management system (Rizvi et al., 2019), a tool was developed, which reduced human workload and increased efficiency.

In (Mao et al., 2002), a wavelet transform was used for text localisation. For finding different local energy variations, the Harr wavelet decomposition method was used. A binary image was acquired after the threshold by local energy variation. They then filtered out certain geometric aspects like size and ratio in order to do the text variation. To form the results, they merged the text regions into different scales. Later, (Prasad et al., 2008) used Hidden Markov Model (HMM) in OCR on Arabic script to improve OCR error accuracy. They used a combination of script-independent and script-specific techniques to glyph models and language models. The first step was to use context-dependent HMMs for modelling shape variations on the position in the word and then to estimate the HMM. Later, a higher-order n-gram for rescoring was applied for language modelling. With the dataset of 297 images consisting of newspapers, books and magazines, the word error rate was 9.6% with n-best rescoring. In future, the plan was to explore the Parts-of-Arabic-Words (PAW) character sequence as a modelling unit in the HMM.

In another paper (Rashid et al., 2012), a Multi-Layer Perceptron (MLP) and HMM was used in segmentation for text line recognition. The model was tested on 1086 lines with 56891 characters from a subset of the UNLV-ISRI dataset. The result reveals the character recognition accuracy to be 98.45%. Furthermore, (Sari & Sellami, 2002) worked on the Arabic language, where words generated by OCR were corrected with the help of a context-based method. The morpho-lexical analysis was used to correct the substitution errors in the case of the Arabic language. Based on the previous study (Magdy & Darwish, 2006), proposed a model which uses a word-based OCR correction approach for the Arabic documents. The information was retrieved from Arabic documents with the help of various indexing. The OCR model used the improved segmentation model by applying noisy channels. For testing, the OCR degradation was considered. For the result evaluation, short n-gram approach was used for indexing. Based on the analysis this model was used for other languages too. Even though the research evolves the improvement in scripting, (Kissos & Dershowitz, 2017), used the classifier which was applied on several images as well as on text. To check the correctness, the ground truth was used, and for comparison, a language model was considered. For experimentation, Arabic newspaper, Arabic Giga word were considered for training and 22000 words for testing. The network gave 50% reduction in word error rate.

Later, (Javed & Hussain, 2009) focused on Arabic script in the Nastalique writing style. Before the text recognition process, Nastalique-specific pre-processing methods were used. This included a page and line segmentation approach. With 500 high frequencies of words testing, they achieved an Urdu word extraction accuracy of 94%. In (Sabbour & Shafait, 2013), a related model (called Nabocr), which used feature extraction with contour extraction, was designed. Then it does shape context followed by page segmentation with line segmentation in a text. Around 20,000 ligatures were extracted from the Arabic text with an accuracy of 91%. According to (Javed et al., 2010), the Nastalique Script standard was used for the recognition of Urdu and Arabic scripts. However, the Nastalique Script was written diagonally, does not contain any baseline for

processing. So, to extract the character, two approaches were used, such as the segmentation-based approach and segmentation free approach. To extract the global features, HMM was used. The 1282 unique ligatures from that 5000 high-frequency words were extracted. The system gave 92% accuracy. Even (Sagar et al., 2008) has described an OCR system for the Kannada language document. They have first extracted images of Kannada scripts and then performed the image segmentation process in the form of line segmentation, word segmentation, character segmentation. Surprisingly, the accuracy stated was 100% using some database approach that was not defined properly. In future, the addition of SVM and a neural network with a segmentation process was proposed.

For Sanskrit scripting, (Avadesh & Goyal, 2018) proposed the Convolutional Neural Network (CNN) to extract the characters from low-quality documents. Segmentation algorithms were used to check the intensity of the characters. Furthermore, (Krishna et al., 2018) proposed a text correction approach based on post-OCR for digitizing texts in Romanized Sanskrit. The method was based on the seq2seq model, and the character sequence was applied as an input to the encoder-decoder framework, which was able to generate the sub-word-level characters in the encoder-decoder model. Validation was tested on a set of 430 images and the character recognition rate achieved was 80.08%. There was no comparison made due to the lack of an existing dataset or research on Romanized Sanskrit text. Recently (Dwivedi et al., 2020) introduced a framework for reading Sanskrit characters using an attention-based LSTM model. They used the CNN model to extract the image features; next, the output was fed to the BLSTM model, which encodes the input image. Based on the encoded features, the LSTM model was used to decode it with the help of a single-headed attention mechanism. Based on the test executed on 20000 lines from the different text, character error rate of 3.71% and the word error rate of 15.97% were achieved. As a gap, this review identifies the need to refine the attention LSTM model by decreasing the Word Error Rates (WER).

Recently (Yin et al., 2019) worked on OCR text extraction on Chinese uppercase characters due to their poor performance in text extraction. The paper proposes deep learning aided OCR technique where the dataset was trained on four neural networks: CNN, a visual geometry group, a capsule network, and a residual network. To further reduce the computational costs, a lightweight CNN method was developed to trim the network weight by 96.5% while reducing accuracy by no more than 1.26%. A character recognition accuracy of 97.70% was achieved over 15 kinds of Chinese uppercase characters in 480 images. In (Zhuang & Zhu, 2005), the OCR post-processing technique was used. The combinational model, statistical language model and semantic lexicon were applied. The candidate information was used to save the search space. Here 60.84% error reduction was achieved. For the evaluation, 1,96,009 Chinese addresses were used. This review identifies the gap as a need for a more precise language model and proper functionality in their research.

In this paper, (Jain et al., 2017) worked on identifying Urdu text using an end-to-end trainable hybrid CNN-RNN model. By applying a variation of the model, the author achieved an accuracy of 98.80% on the UPTI dataset, using the hybrid 7-CNN-RNN-fine model. The challenge was in cursive writing-based images. Furthermore, the authors proposed attention modelling into text recognition for better object detection and captioning complicated tasks. Later, in (Singh & Kaur, 2010), the OCR technique for Telegu character recognition using an artificial neural network (ANN) was implemented. To recognise a large number of characters, the backpropagation algorithm was used. (Krishnan et al., 2014) implemented a web-based OCR system for seven Indian languages which use a unified architecture. The architecture uses a segmentation free approach and addresses the issues of UNICODE reordering. It is capable of continuous user input and feedback. They used a BILSTM based transcription for text extraction. With 1000 pages of Hindi text, the character level error rate and word error rate were 1.80 and 5.72, respectively. In the future, they hope to extend their work to the Marathi and Urdu

languages and increase the word recognition rates. These represent gaps in their research in the areas of word recognition rates and multilanguage functionality.

In (Vinitha & Jawahar, 2016), a recurrent neural network (RNN) was used for the word classification that is to check if the word contains an error or not. A generic error detection method was applied to four different Indian languages. The datasets consisted of 5000 document images of each language, and the average performance of the model was above 80%. The future scope of the project is to detect real-world errors in OCR output. They plan to work on a variety of languages in future. In a related paper (Saluja et al., 2017), Indic OCR was used for correction. A Long Short-Term Memory (LSTM) based character-level language model was designed, which locates errors from discriminative language modelling with a fixed delay. The Four languages F1 Scores was above 92.4% but decreased in WER by at least 26.7%. The future performance scope of BLSTM models and character level attention models was discussed. Later, (Dave et al., 2020) presented a geometrical rectification framework for better image capturing, and Tesseract was implemented with an extended LSTM based on a recognition engine. The precision of this method was 85%. Developing an OCR that can take multiple font input, support multiple languages, and act as a translation device is an area of future research that their work recommends. (Paul & Chaudhuri, 2019) proposed OCR text extraction from Bengali and English text using a single hidden BILSTM-CTC architecture having 128 units. They have also not used any peephole connection and dropout in the BILSTM, which gave better accuracy. By training 47,720 text lines and with 20 different Bengali fonts, they achieved a character level accuracy of 99.32% and word-level accuracy of 96.65%. The future work was to improve CTPN with BILSTM-CTC for better accuracy.

Again, in the paper (Wickramarathna & Ranathunga, 2019), an OCR was used for character recognition. The system was designed to take the Bharmi character array as an input instead of word boundaries and later convert it into Sinhala sentences. The proposed model consists of networks such as two bigrams, one trigram and a translation model. For experimentation, 800 sentences were used, and an accuracy of 91% was achieved. Their

work is limited by its tendency to produce grammatically incorrect sentences, making the production of grammatically correct sentences a gap in current research. According to (Belay et al., 2020), an end-to-end Amharic text-line image recognition approach based on Recurrent Neural Network (RNN) was presented. On a dataset of 76850 sample images, they achieved a character error rate of 1.05%. Furthermore, in (Alshehri, 2021), the text method was implemented based on various distance algorithms for text extraction and recognition. For the text extraction centroid method was used, whereas text recognition was done with the help of weighted Euclidean distance as they want to implement a lightweight system which can further be used in the mobile application. The system accuracy was 99% and it was developed on mobile with reasonable execution time.

(Yeremia et al., 2013) used a backpropagation network algorithm along with various genetic algorithms to execute the model in less time. By training on each character of the English alphabet, the model achieved an accuracy of 90% in character recognition. Later, (Bissacco et al., 2013) proposed a system called PhotoOCR based on isolated character classification in machine learning. The solution was evaluated on public benchmark datasets of around 2.2 million characters. They achieved a word recognition accuracy rate of 82%, a promising result when to the ABBYY baseline of 35%. Following this development, an innovative method for translating printed English text into an equivalent Braille character set was introduced by (Chakraborty & Mallik, 2013). They contributed to the development of an application using OCR that extracts text from scanned images for further conversion. The extracted text included numbers, alphabets, symbols, and compound letters, which were translated to a six-dot cell Braille format which was saved for printing the document. In this paper, the author proposed a fast and accurate scene text detector. The proposed method uses Fully Convolutional Network (FCN) model that predicts text regions. The pipeline was flexible in producing either word or line predictions. The author claims that the proposed algorithm significantly outperforms state-of-the-art methods in accuracy and speed (Zhou et al., 2017a).

Table 4: Gaps Identified: Written Languages and Scripts.
"Devised by the author".

Research Gaps	**References**
High Word Error Rate	(Dwivedi et al., 2020)
To work on a precise language model	(Zhuang & Zhu, 2005)
To increase the word recognition rates and to work on Multilanguage's	(Krishnan et al., 2014)
To work on an OCR that can take multiple font inputs, support multiple languages, and act as a translation device	(Dave et al., 2020)
By using BILSTM-CTC, the character and word level accuracy were not good.	(Paul & Chaudhuri, 2019)
The system generates grammatically incorrect sentences.	(Wickramarathna & Ranathunga, 2019)

2.3.2. Real Scene Images

In this paper, (Chang et al., 1995) used content-based indexing for video servers and databases to extract visual information. The visual information was nothing but shape, colour, texture, background. For the texture extraction, different techniques were used, such as wavelet transform, Discrete Cosine Transform (DCT), and subband transform etc. For the colour, the histogram was calculated. For the shape, traditional techniques such as edge-based, region-based, or feature-based were used. The research was tested practically using Columbia University's Multimedia. Later, (Chun et al., 1999) used the combination of FFT (Fast Fourier Transform) and neural network for reducing the processing time. FFT computation has taken place for the reduction of overlapped segments of 1 × 64 pixels. The output where each segment is of 32 features. Labelling and noise elimination was done with the help of neural network output. Though the author has stated that the system can be used in real-time, the paper's processing rate was not reported.

Furthermore, (Chiang & Knoblock, 2011) used a set of raster maps with straight, curved, and multi-oriented text labels in various sizes on google map. The google map data was used to achieve 93.7% accuracy. Different kinds of methods were possible depending on

the image, but edge-based focuses on the contrast in between text and in the background. The edges of any text boundary identification were the first role, and then merging of data takes place. An edge filter was used for edge detection. According to (Smith & Kanade, 1995), the input image will be in different filtered with 3x3 horizontal to image and perform threshold for finding vertical edges. The smoothing operation was used to remove the unwanted broken parts and then connect the detached edge elements. To find characters of identical shape and texture, the final extraction of text strength histograms of each cluster was performed. In the final extraction of text, recording the intensity histogram of each cluster was proposed to get similar shapes and character textures. The traditional texture-based methods face the problem of computational complexity during the texture classification process, and it increases in the processing time.

However, it was also required to scan the input image to detect and localise text regions. (Sin et al., 2002) proposed a frequency feature-based model where the number of edges in a pixel was taken as a frequency feature. From the scene image, the number of horizontal and vertical lines indicates the frequency features. By assuming that many of the text regions were rectangular in the background, a Hough transform detection of edges was used. Still, it is not clear that these three stages will provide results.

(Beaufort & Mancas-Thillou, 2007) have presented an OCR correction system to recognize natural scene text. They used the Finite State Machine (FSM) algorithm to solve this problem of OCR confusions, capital/accented letters, and lexicon look-up. There are two primary interpretations in the FSM, the oriented graph of a label with arc and definitions of a class of regular language. With around 400 scenes of natural words, they achieved the correct recognition rate of 94.7%. In future, they propose to work on an additional machine for morphemes and a syntactic machine to correct real words. The inability to detect real words is an important research gap. Further research has been carried out in (Wang et al., 2011), which focuses on word detection and recognition in natural images. They used a two-stage pipeline that included text detection and a leading OCR engine. Later, based on previous work in generic object recognition, they have

shown that the latter-based approach achieves superior performance. By using Chars 74K dataset, they achieved an F-score of 40%. Furthermore, (Elagouni et al., 2012), for scene text recognition, a neural classification method without the traditional character segmentation stage was used. By using ICDAR 2003 database with 5689 sample images, the proposed method got an accuracy score of 66.19% in comparison to ABBYY FineReader and Tesseract, which received a score of 42.80% and 35% respectively.

In this paper, (Karaoglu et al., 2012) implemented a system for text extraction from natural images to aid visual classification. Saliency-based object recognition, scene text recognition, and object recognition with the aid of recognised text are among their methodologies for text extraction. Their tests were performed on the ICDAR 2003 dataset, and the results were 0.68. These results were compared to 0.37 and 0.00 for ABBYY and Tesseract, respectively. Based on the report by (Karanje & Dagade, 2014), a dynamic threshold was used in the detection of edges from the wavelet coefficient. However, for further effectiveness, edges were obtained using an alternative heuristic threshold by blurring the approximate coefficient. For final text extraction, the region of interest was used, and an evaluation of 80 pictures was done. Here the accuracy rate was 91.20%, which was beneficial for robust to noise form by using the methods like wavelet transform and ROI. Furthermore, (Busta et al., 2015) have designed a model for scene text detection called FASText Keypoint Detector, which has two keys that were used for segmentation, such as Stroke Ending Keypoint (SEK) and Stroke Bend Keypoint (SBK). For evaluation, ICDAR 2013 was used, and an F1 score is 75.9 was achieved.

According to this paper, (Islam et al., 2016) used text extraction from natural scene images for detecting the enhanced Maximally Stable Extremal Regions (MSERs) method. Later, the OCR was performed with an f-score of 77.47% on the ICDAR 2011 dataset, which was better than the previous method performance of 76.22%. Based on the need of the study (Tian et al., 2016), proposed a novel Connectionist Text Proposal Network (CTPN), which localizes text lines in natural images. Based on 1500 sample images, 0.74 precision rate on ICDAR 2015 dataset. Furthermore (Lee & Osindero, 2016) has

presented an approach for text extraction from natural scene images. The method was a recursive recurrent neural network with attention modelling for lexicon-free. In this technique, they used recursive CNNs for feature extraction, then a character-level language model for avoiding n-gram and soft attention mechanism to exploit image features in a coordinated way within a backpropagation framework. Using 647 cropped word images with 50-word lexicons, the text recognition accuracy was 96.3%. In the future, they hope to explore recursive fully CNN in order to overcome the gaps in their research relating to connecting extracted image features and the corresponding location on the input image. As stated in this paper (Shi et al., 2016), a novel neural network architecture called Convolutional Recurrent Neural Network (CRNN), was developed which is a coherent system that combines sequence modelling, feature extraction, and transcription. Using 251 scene images with text labelled bounding boxes, they achieved an accuracy of 97.6% on the IIIT5k dataset as compared to ABBYY, which received 24.3% only. Recognition of text from images is carried out by OCR. The fuzzy logic controller provides a technical interface mechanism where the transaction will be carried out from man to machine to man. This controller adapts new techniques due to the addition of existing processing methods. Recognition was the question that arises due to finding the complexity in normal consonants, which arises due to the position maxing with post level consonants.

Later (Bartz et al., 2017) designed and STN-OCR, semi-supervised neural network for text extraction from natural scene images. The proposed model was simple but complicated in training. The accuracy on a test dataset achieved was about 97%, but the authors accepts that the model is still not good enough for detecting arbitrary locations in the image. Another way, they proposed, “A single Neural Network for Text Detection and Text Recognition”, which was a semi-supervised neural network. For text detection, Spatial Transformer was used to find feature map, text recognition CNN followed by ResNet. The network was trained using Connectionist Temporal Classification (CTC) loss which is helpful in the case of prediction. For experimentation, publicly available

datasets were used, such as SVHN, French Street Name Signs (FSNS). Furthermore (Wang & Hu, 2017) has proposed the Gated RCNN (GRCNN) model for recognizing text in natural images. The proposed model outperforms the existing datasets, including the IIIT-5K, Street View Text (SVT) and ICDAR.

In (Kumuda & Basavaraj, 2017), the authors presented an algorithm for extracting and analysing text from complex scene images. Initially, edges are detected by DWT. Next, clustering and an AdaBoost classifier were applied for the text region localisation. For character extraction, heuristic rules were applied. Finally, OCR was used for analysis. They designed an algorithm that evaluates the various types of database images using four main steps (pre-processing, text localisation, text extraction and character recognition). Initially, the input image was taken and transferred to the pre-processing stage. The pre-processing stage performs two tasks: colour or grey conversion and median filtering. After that, DWT, Connected Component CC analysis, and text extraction with the help of morphological operation and Adaboost classifier were used to perform the localisation task. Morphological operations and heuristic filtering were carried out before the OCR derived the final text in the extraction phase. These are the overall steps they followed in the execution process. This algorithm also uses Haar discrete wavelet transforming with a soble edge detector to extract text from scene images. From given input to OCR, result analysis is displayed in a notepad. The algorithm extracts text efficiently in case of font, size, and orientation. However, this method fails when detecting text from window frame bricks, edges, and leaves, a significant gap in their research. Their future work was to overcome the above issues and the speed of extraction.

(Ren et al., 2016) proposed the ‘Faster R-CNN Inception v2 Pets’ model for use in building a text recognition model. This model was configured for ‘Oxford-IIIT Pets Dataset’ and used the basic CNN model. A CNN was a Deep Learning based algorithm used for processing an input image. The CNN algorithm was used to identify various critical objects or aspects, according to their importance, in the image and differentiated one from the other. CNN was a supervised machine learning method with high accuracy,

and it was used for image classification. R-CNN (Region Convolutional Neural Network) algorithm was used for object detection.

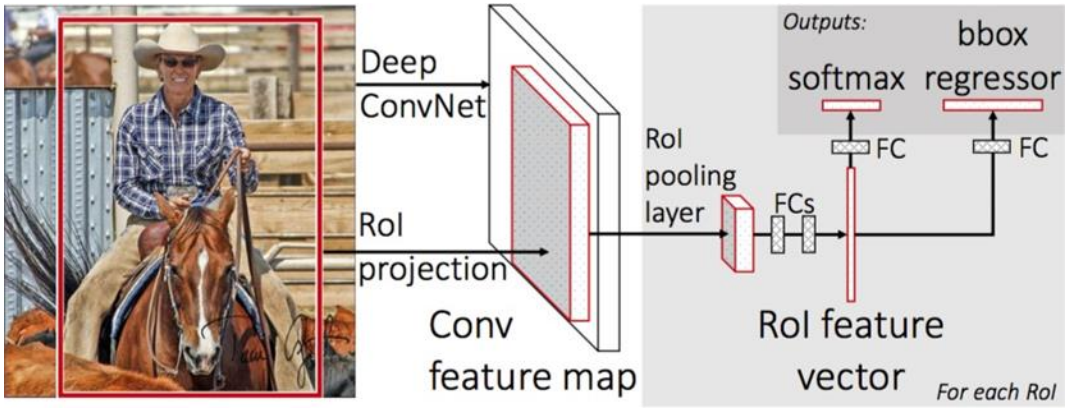


Figure 8: Fast R-CNN (Girshick, 2015)

The primary limitation of CNN was that it could not identify the location coordinates of multiple objects at one time. CNN can only identify the class of an object. When multiple objects were contained within the CNN bounding box's visual field, regression is limited. Hence, the bounding box location of multiple objects cannot be generated in CNN. R-CNN specializes in recognition of multiple objects with their respective location coordinates. R-CNN focuses on one region at a time, thus minimizing any interference. A selective search algorithm was employed by R-CNN for object detection. This algorithm was designed to be fast but with powerful recall capabilities. It was based on the hierarchical grouping of similar regions based on various features. These features can be based on the colour, texture of the object, as well as size and shape. All the regions were resized into equal size before they were fed to a CNN for classification and bounding box regression.

(Girshick, 2015) implemented the Fast R-CNN model to overcome the drawbacks of the R-CNN model. They used the CNN model for generating feature maps based on an input image. The region proposal network was used to map the features onto a square shape. The basic functionality of the RoI pooling layer was to reshape the square into fix size

and pass it to a fully connected layer. The Fast R-CNN model was faster than the R-CNN model because it directly generates the feature map with the help of a convolution layer. The selective search creates the problem in both models.

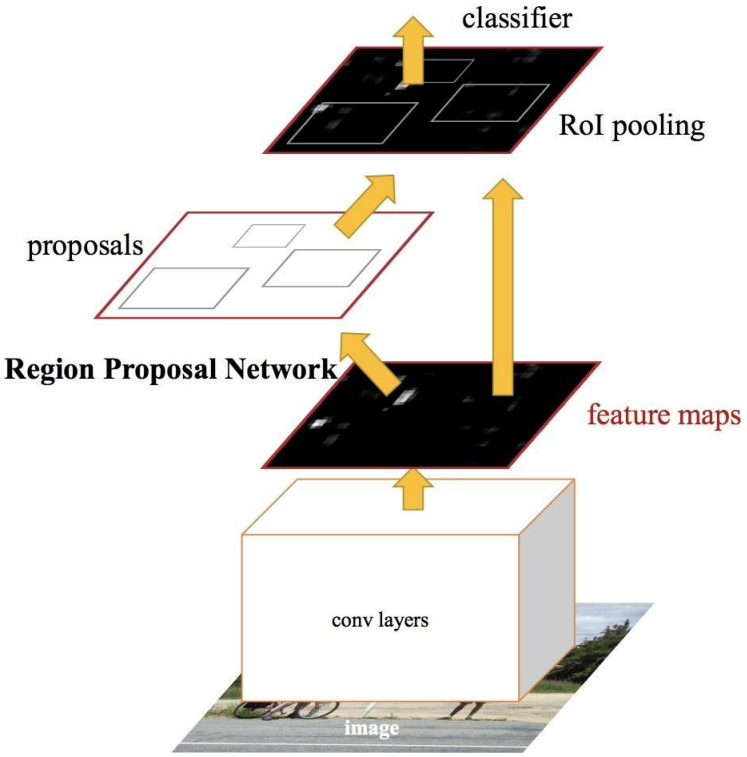


Figure 9: Faster R-CNN (Ren et al., 2016)

Region Proposal Network (RPN) (Ren et al., 2016) takes an image of any size because input and output are nothing but the set of rectangular object proposals, each with an 'objectness' score. A small sliding network was applied over the convolutional feature map output. To generate the region proposals, the last convolution layer was used. The small sliding window was applied to the input convolution feature map. The RPN architecture has been developed with the help of n×n convolutional layer followed by two siblings 1×1 convolutional layers. The object or non-object pattern and bounding box regression outputs were specified in relation to an anchor, which was a collection of reference boxes. To cover artefacts of various shapes, the anchors use several pre-defined

scales and aspect ratios. The anchors were calculated by multiplying the aspect ratio and scale (Ren et al., 2016). The ground-truth image bounding boxes were used to allocate the training labels to the anchors based on their Intersection-over-Union (IoU) ratios. If the value IoU was highest for the given ground truth box or had a value above 0.7, then the anchor was assigned with positive labels. Another condition if the IoU value was less than 0.3 for the ground-truth box, then the negative label was assigned (Ren et al., 2016; Lin et al., 2017)

Later, (Wang et al., 2017) proposed a "scene text recognition algorithm based on faster RCNN" to improve text recognition. The correct rate of faster RCNN was 90.4%, and the correctness rate was 88.9%. The results were compared with conventional detection models. This technique was helpful in the case of scene character recognition. Later, (Cheng et al., 2018) researched text extraction from natural images. They proposed an Arbitrary Orientation Network (AON) model that targeted irregular text images where the existing research had certain limitations. The experimentation was performed on the datasets like CUTE80, SVT-Perspective, IIIT5k, SVT and ICDAR. The proposed method achieved better performance in irregular datasets. (Borisyuk et al., 2018), proposed a solution called Rosetta for OCR text recognition from real scene images uploaded to Facebook. This method was implemented to detect the individual words by using Faster R-CNN, and to produces the lexicon-free transcription of each word with the help of a full convolution neural network. The dataset used for the evaluation was COCO-Text. The author has achieved +6.76% relative accuracy on synthetic training using CTC model. For OCR on a multitude of images, a system was developed by the author for identifying and extracting text from images (Borisyuk et al., 2018). This system utilizes Faster R-CNN's object detection techniques, and it supports text detection and text recognition. The system can also recognise text in a variety of languages on its own.

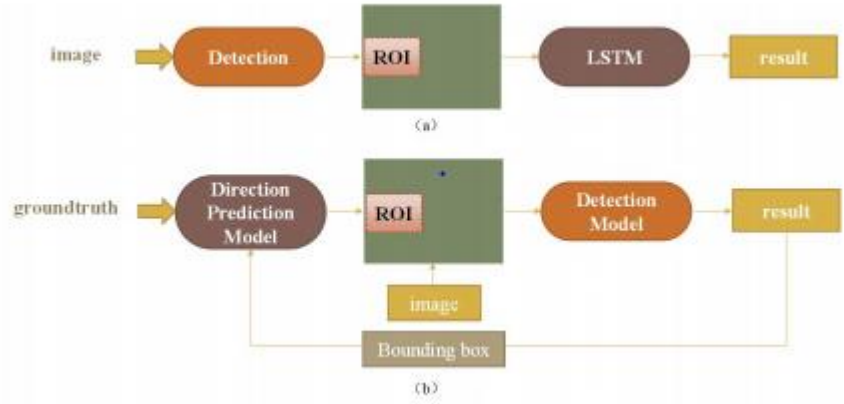


Figure 10: Workflow of ROLO (a) and our framework (b). The three pipelines: Direction Prediction Model, ROI and Detection Model. (Zhang et al., 2018)

(Zhang et al., 2018) proposed Object Detection and Tracking based on the direction prediction model constructed by using RNNs. They used Single Shot Detector (SSD) as the detection model and LSTM for the direction predictions. For the evaluation, OTB and VOT2016, two datasets were used. They resolved the issue of information vanishing by using the LSTM. The workflow of this method is shown in the figure above.

(Namysl & Konya, 2019) proposed a CNN encoder to extract features from the natural scene images. This was achieved by mixing different methods such as synthetic training data generation, data augmentation techniques and deep learning methods. Based on the training data using large text corpora and over 2000 fonts, they found a character error rate of 1.76%, which was less than the ABBYY "Fine Reader" character error rate. Afterwards, (Huang et al., 2019b) proposed a "Mask R-CNN with pyramid attention network for scene text detection" for enhancing the feature representation as compared to Mask R-CNN. One gap illustrated by this model is poor performance when it comes to long text lines. For accurate detection of text in a natural scene, a Faster R-CNN based approach was proposed by the author (Lu et al., 2019). They focused on challenges for small datasets. The proposed method was a multi-scale text feature extraction network. The network used a feature pyramid based on Faster-RCNN, which presented complex

features of natural scene images accurately. Later (Zhong et al., 2019b) used another approach to replace the bounding box regression module with the LocNet model, which will improve the localisation accuracy of a Faster R-CNN based text detection. The result was achieved on the superior multi-oriented (MSRA-TD500, ICDAR-2015) text detection and horizontal (ICDAR-2011, ICDAR-2013 and MULTILIGUL) datasets.

Furthermore, (Brzeski et al., 2019) proposed “Evaluating Performance and Accuracy Improvements for Attention-OCR”. To improve the results in the case of real-world data, the author has designed the Attention-OCR model. The accuracy of the model was improved by using various models such as dynamic RNNs, and Bi-LSTM. The Bi-LSTM gave better accuracy, and dynamic RNNs was used to provides less computational complexity. Based on the study, (Zhu et al., 2019) have presented “Rotated cascade R-CNN: A shape robust detector with coordinate regression”.

The limitations of object detection techniques avoided here by firstly using Locally Sliding Line-based Point Regression (LocSLPR) which defines the outline of an object and then Rotated Cascade Region-based Convolutional Neural Network (RCR-CNN) to target the object. Using this method, real-time object detection was not possible. Furthermore, in (Zhong et al., 2019a), an Anchor-Free Region Proposal Network (AF-RPN) was proposed for Faster R-CNN-based text detection. According to the author, the anchor's mechanism was ineffective for scene detection, and the proposed solution was to achieve a high recall rate on both horizontal and multi-oriented text detection benchmark tasks. In Faster RCNN and SSD, “anchors were used as references to predict the corresponding region proposals or target objects, and the label of each anchor was determined by its Intersection Over-Union (IoU) that overlaps with ground-truth bounding boxes” (Ren et al., 2015). The proposed solution gave a better result but was limited when the image has a low contrast pixel, and blur or text font was less than 12px. These are the known limitation in invoice data extraction, especially retail receipts where font sizes are small. The author has used a large dataset of raw images of digital meters (Kanagarathinam & Sekar, 2019). The research work highlights the reduction in

infrastructure cost issues and a smart metering method in measuring and managing electricity usage. An alternate approach using SVM was applied to Bangla language character which was based on OCR (Pervin et al., 2017). Additionally, the author has used feature fusion of two different feature vectors obtained by Zoning and Gabor filter by passing the features through a classifier. The recognition accuracy by individual features and feature fusion was compared, revealing that the feature fusion-based method performed better (92.99%) than a single feature extraction method (68.15% for Zoning, 89.73 p% for Gabor filter) during classification. Later, (Zhou et al., 2017a) proposed the pipeline structure with two Fully Convolutional Networks (FCN) and NMS merging stages. For the experiment evaluation, ICDAR2015, COCO-Text and MSRA-TD500 datasets were used. The proposed model achieved an F1 score of 0.8737 on ICDAR2015. Their future scope is detecting curved text, text recognition, and general object detection to overcome the primary gap in their research. After studying existing methods, (Li et al., 2018) implemented a novel Progressive Scale Expansion Network (PSENet) for scene text detection. The network was trained by using the Stochastic Gradient Descent (SGD) algorithm. For the experimentation, three datasets were used as ICDAR 2015, ICDAR 2017 MLT and SCUT-CTW1500. For the ICDAR 2017 dataset, an F1 score value of 94.34% was achieved.

(Goel et al., 2019) presented a model that uses Open-Source Computer Vision Library (OpenCV) and CNN to extract English text from images. The CNN model was designed to directly detect the characters in the scene images in the first step with the help of the two-stage pipeline architecture of a single neural network. The second stage is non-maximum suppression (NMS), which suppresses the multiple bounding boxes that were generated around the same text region to hold just one box. The proposed OpenCV implementation model gave the precision rate value of 88.3% and a recall rate of 76.8%. The results indicate that the model was not capable of producing significant false positives but may sometimes come across the false negatives, which may not detect regions containing text data. The model has some drawbacks. The algorithm was unable

to detect text when the text was not correctly aligned to the horizontal axis; thus, it was unable to generate the rotated bounding boxes. Additionally, when the text was embedded in a circular shape, a similar problem occurs. These gaps should be addressed with future research.

Table 5: Gaps Identified: Real Scene Images.
"Devised by the author".

Research Gaps	**References**
Proposed FSM unable to correct real words	(Beaufort & Mancas-Thillou, 2007)
To improve image feature extraction and text localization	(Lee & Osindero, 2016)
The method fails in detecting text, and the speed of extraction takes more time	(Kumuda & Basavaraj, 2017)
Mask RCNN fails in the detection of text when input is of the long text line	(Huang et al., 2019b)
Unable to detect the curved text, text recognition and general object detection	(Zhou et al., 2017a)
Unable to detect text when the text was not aligned	(Goel et al., 2019)

2.3.3. Text from Video

In the context of text extraction from video and moving images, (Qi et al., 2000) used a simple linear SVM-based method to identify text in videos and images. The sample data was based on, tv program of 2 hours and 21 seconds of MPEG-7 data, which used a CNN algorithm. The objective of this paper was to create an intelligent based tv browser based on text search, and a relevant frame was shown and HTML-based video browsing. They have received 91.3% accuracy for the ten most frequent categories and 85.5% overall 118 categories. Using SVM was a viable choice, but applying a simple method was not enough to solve the primary purpose for multiple data extraction and classification. Furthermore, the result was varied based on video quality, which was a list of images that OCR scans to extract text. Later, (Hua et al., 2002) have applied multiple frame information extraction methods to combine repetitive frames with similar text for better text extraction. The accuracy was improved, but again challenges were due to the quality

and intensity of the background of the frames. Furthermore, only text can be extracted with information extraction, like in invoice data extraction. In another paper (Hauptmann et al., 2002) used multiple-modal information retrieval for text extraction from images and as well as from speech. The results for combining OCR and speech recognition were promising and worked well when image quality was good.

Furthermore, (Chen & Bourlard, 2001) proposed the system for OCR video to extract the keywords. To identify the text region, SVM was used. To enhance the text regions, asymmetric filters were applied. For the experimentation, 27054 characters were used. The text recognition rate was 82.6% with the enhancement method and was 36.1% without the enhancement method. The system gave an error when two characters were close together. So, in the future scope character classification algorithm was proposed to be implemented for better classification. In the case of (Chen et al., 2001), they used an operator named a Canny operator to detect edges in an image. Edges of each text are enhanced in terms of the scale of information. Morphological dilation was used to connect all the edges into any cluster. The horizontal and vertical aspect was used in finding the filter out of non-text clusters. (Gllavata et al., 2004) designed a technique that automatically localises, segments, and binaries text appearing in an image. The methodology used was k-means clustering based on wavelet transform for text regions detection. Then with the use of connected components, exact text positions were found via a refinement algorithm. And finally, an unsupervised learning method was applied for text segmentation and binarization using a colour quantizer and a wavelet transform. Around 51 images with variety in background were tested and the result of the detection and localisation performance in terms of pixel-based recall was 82.6%. The authors hope to address the gaps in their research by working to automate the indexing of images and videos in order to get their content-based retrieval information.

Later, (Yang et al., 2011a) used a weighted, discrete cosines transformation text detector which used dynamic image contrast/brightness adaption to enhance image text quality. The F1 score of text detection using DCT Detector was 90% over 180 frames of video.

In the future, the authors have proposed dictionary-based post-processing with additional data to the DCT detector for improving the text recognition rate. In another paper by (Lyu et al., 2005), a network for video text detection and recognition was proposed. Two languages were considered, such as English and Chinese. The multiresolution text detection from the multilingual text characteristics was classified. The detection accuracy of the system was 90.8%. In the future, they would work on detecting the text in a non-steady environment as they did research on steady text.

A method for extracting text from an image captured by grabbing the content of the television screen was proposed by the author (Kastelan et al., 2012). This system was used to verify whether the television functions were working properly as per the text displayed on the screen or not. To see if the TV responds to remote control orders, the text on the grabbed picture was read. Furthermore, in the field of video indexing and analysis, (Yang et al., 2011b) had adopted a localisation and verification scheme and then performed text detection. Later, they have applied a method to delete false alarms from the text detection stage and analysed the slide structure by using SWT (Stroke Width Transform). To recognize texts, a multi-hypotheses framework consisting of multiple text binarization, OCR, spell checking and result merging processes were considered. The algorithm consists of geometrical information and text stroke width of detected text lines. With the dataset of 10,000 video frames and a title extraction, an F1 score of 90% was achieved. Furthermore, (Halima et al., 2012) used a technique for text extraction and recognition from Arabic video clips called the neuro-fuzzy system. They used colour and edges for extracting text, then localized the text, tokenized the text and then performed segmentation. On a 2,000 keyframe of Arabic video, they achieved a recall of 85% on Tunisia one tv video frame. The extraction rate was increased by about 91%, and text recognition was about 84%.

In this paper, (Ye et al., 2005) proposed a novel coarse-to-fine algorithm to extract the text information from the images as well as videos with complex backgrounds. The coarse detection was used to locate the pixels of text. The density-based region growing method

was helpful to connect the pixels with the region. The fine detection was used to extract texture features using the forward search algorithm. At last, SVM was used as a classifier. For the experimentation, 177 images from webs, broadcast videos frames, and Microsoft common test set containing 44 images were used. The proposed algorithm resulted in a 94.2% recall rate. The text was detected, but there was a lack to extract the text from the complex background and videos. One gap in their research was the systems inability to detect and recognize text in front of a complex background. (Poignant et al., 2012) have researched text extraction from video images as well as personal identification by performing several text detections, temporal tracking and semi-supervised parameter setting techniques on 59 video samples with a resulting F1 score of 77.3%.

Table 6: Gaps Identified: Text from Video.
"Devised by the author".

Research Gaps	**References**
Unable to automate indexing of images and videos to get their content-based retrieval information.	(Gllavata et al., 2004)
Fails to detect and recognize text clearly in the complex background	(Ye et al., 2005)

2.3.4. Non-Invoice Document

In this paper, (Baumann et al., 1997) used a prototypical analysis system, OfficeMAID, to extract and recognize the text from the message document. This system analysis was carried out for the features of daily work of purchasing, workflow etc. This model has three steps, including structural analysis, text recognition, and feature extraction from the information. For the evaluation, 500 business documents were used in such a way that the first training set has 91 documents, the next training with 176 documents and 229 for testing. Furthermore, (Ho & Nagy, 2000) have described a document specific OCR system by applying on faxed business letters. They used unsupervised classification of the segmented character on each page. The letter identities were assigned to each cluster for increased matches with a lexicon of English words without any shape of training. With

200 English business letters, the authors identified approximately 80% of the words with a lexicon in 2/3 pages. Additionally, the method was not fit for the (Ho & Nagy, 2000) and have an issue with low-quality images. Furthermore, (Jin et al., 2002) implemented a generalized content-based correction model, which works before the OCR correction model and boosts the retrieval performance. By using the Language Model (LM) instead of correcting whole word, a retrieval-based approach was used for the correction. The TREC standard was used for testing, and it shows that system has improved the performance for the correction. The future scope was to extend the model and use several features instead of one as they used rank information only.

Later, (Zhang et al., 2002) worked on a colour image instead of a binary image. The features were extracted from the colour image to check the variation of intensity during the normalisation. The Gabor transform was used to find local features and LDA for classification. The proposed algorithm was applied for Chinese sign recognition. The recognition accuracy of the system was 92.46%. For the evaluation, the Chinese national standard character set GB2312-80 dataset was used, which contains a total of 3755 different characters. Based on the need of the research (Nartker et al., 2003), the MANICURE system in combination with other post-processing systems was proposed to get quality results as that of ground truth. The word document was converted into text by removing the tables, figures, background manually. Seventeen documents were used for evaluation. The raw OCR output accuracy was 91.91%, and MANICURE output accuracy was 94.14%. Later, (Suzuki et al., 2003) used the INFTY method, which has four steps, including “structural analysis of mathematical expressions, manual error correction, layout analysis, and character recognition”, which was designed to get better performance. For the evaluation, 500 pages of mathematical documents were used to recognize the characters. The recognition rate in the case of text was 99.44%, and mathematical expression has 95.18%. (Zidouri, 2004) used ORAN method to understand and recognize the document. This technique classifies the document into three parts as text, picture, and graphics. The knowledge-based method was used to recognize the text

document. More than 6000 characters were used for testing, and the recognition rate was 97%, and the model document rate was 99.7%. One gap identified in this research was its inability to recognizes characters when they were remarkably close. The future scope is to work for a multi-font characters system.

Furthermore, (Jacobs et al., 2005) have described a method to extract text from low-resolution images. They used a CNN combined with language models using dynamic model programming. After training, the character recognizer used a set of 15 pages of training data and a dictionary of 3,64,778 English words. They achieved a text recognition accuracy of 80% from images that were taken by a 1024x768 APLUX camera. This method was slow but suitable for low-quality images. Later, a new approach was carried out by (Shivakumara et al., 2005). It was a new Boundary Growing and Hough Transform Based approach and was used for binary image document analysis. To improve the accuracy of the model, the Hough Transform Based Method (BGM-H1) was used. For skew detection from the binary image, the Hough Transform was used.

(Fataicha et al., 2006) developed a model for the information retrieval system based on the OCR. The proposed technique randomly collects error grams and correction rules for the query. The architecture was trained on 979 document images and tested on 100 scanned Web pages. The total character and word recognition rate was 94.83%. In the future, it was proposed that the effectiveness of retrieval could be improved by using different techniques. Furthermore (Laine & Nevalainen, 2006) tested the capability of a mobile camera without OCR software and added hardware interfaces. This system was implemented on Nokia 6630 camera in Symbian C++. This system was only capable of recognizing English letters written in black font on a white background, a gap that should be addressed with future research.

(Grover et al., 2009) developed an approach for embedded text in complex coloured document images. They used simple edge detection, thresholding techniques, and block classification. Here, the conversion of images from greyscale was performed by taking

the weighted sum of RGB components. For edge detection, simple greyscale conversion was performed using masks, which separates horizontal and vertical edges. After getting edges, they were divided into small overlapping blocks represented in terms of m-by-m pixels, where m indicates the image's resolution. Then block classification methods was used to differentiate text from the image with the help of a pre-defined threshold. Dataset from newspapers and magazines scanned at 150, 300 and 600dpi, were selected. In terms of sensitivity to colour fonts, this approach gave 99% accuracy. Good results in high sensitivity and low false alarm rate were achieved. There was a problem when the gradient of text and image were quite similar. In the future, to achieve high sensitivity, the generalized value of gradient in context to intensities of text for any image was required. Later, (Zhang et al., 2009) designed an OCR based android application using the generic framework. The issues related to capturing the images were resolved. Other two applications were developed, namely PocketPal and PocketReader, to check the performance. The binarization of this method gave 96.94% accuracy. The future scope is to develop a system that works when the image consists of a complex background and involves merging of multiple images. This review identifies these issues as gaps that should be addressed. Later, (Kluzner et al., 2009) used adaptive binarization, registration, and optical flow-based distortion compensation to extract text from historical books. An adaptive word recognition accuracy of 86.7% was achieved over 18,984 individual words. (Fabrizio et al., 2009) introduced a technique for detection as well as extraction of text from commercially taken screenshot images. For labelling, they combined two methods as blob extraction method (edge-based method plus connected component labelling method). The extraction process was done by the collation of a homogeneity detection filter and threshold number. Here the result of successful extraction on complex background was 94.66%.

Furthermore, (Packer et al., 2010) implemented a method to extract named entity recognition from scanned and OCRed historical documents. They have applied dictionary-based, rule-based regular expression, maximum entropy Marko model and

ensemble extraction methods to various types of noisy OCR data. With 12 titled historical documents, they achieved 89.65% precision of newspaper title extraction. A further plan was to add a supervised machine learning approach for better extraction of data. In the paper (Nagabhushan, 2010), have developed a canny edge detector for detecting edges for extracting text in the complex colour based background. The presence of dilation operation on the edges creates holes in the nearest components, which instead creates a character string. The components which were not nearest for the dilation were eliminated automatically. But here only connected sets could be identified. When there were more conditions based on the situation, then this method may not provide good accuracy. For eliminating other non-text components, the method used an analysis of standard deviation from connected components and computing. For performing segmentation, an unsupervised local threshold was reprocessed. In the end, text regions were found and reprocessed. The methods like canny edge detector, dilation operation, unsupervised local threshold, and the connected component analysis gave 97.12% accuracy in handling degradation as blur or wavy text format.

Later, the author proposed an algorithm for license plate recognition, which was applied in intelligent transportation systems (Wen et al., 2011). It used shadow removal techniques and character recognition algorithms like SVM. The article also presented some improved methods, such as image grey enhancement and image tilt correction. The algorithm quickly recovered against the view angle, variance in illumination, position, colour, and size of the license plates when working in a complex environment. In this paper, (Shinde & Chougule, 2012) proposed a model for text detection and recognition of seven-segment numerals of digital energy meters. Later, they designed a unique profile-based method for segmenting printed text and developed an algorithm for correcting skew angle produced during text document scanning. The text in a document image is separated into characters, lines, and sentences, using this algorithm and segmentation of characters in any text document. This method can find the total number of lines and words as well as count the number of words in a specific line.

In this paper (Lund et al., 2013) used multiple global threshold binarization techniques instead of single binarization for OCR text extraction. The test was performed on 1074 images, and the results achieved were 8.41% and 6.79% on Word Error Rate (WER) and Lattice Word Error Rate (LWER), respectively. A further plan was to focus on character classification. Later, (Tian et al., 2013) proposed a co-occurrence histogram of oriented gradients (Co-HOG) to recognize the text in scenes. This technique was useful for capturing the spatial distribution of neighbouring orientation pairs instead of just a single gradient orientation. The ICDAR dataset consisting of 11000 characters gave character recognition accuracy was 83.6%. In future, they would add a global feature with a co-occurrence histogram of oriented gradients to enhance the character recognition. With the study of existing approaches, (Mithe et al., 2013) have presented a character recognition method using OCR technology and an android phone with a higher quality camera. Firstly, to identify the individual glyphs, a binary input image was segmented. Later, feature extraction was done to generate a vector of numbers from each glyph that can be used as input features for an ANN. This paper does not discuss any finding, but only proposes a system with further work on OCR mobile application by using table boundary detection and post processing techniques. (Leon et al., 2013), have developed a model to extract the caption text with the help of a hierarchical region-based image model. This model has used two features such as geometric and texture-based. For the texture information, Haar wavelet decomposition and regions identification were proposed with the help of geometric features. The system has 85.21% accuracy. In future, they would work to get all the text in the textured area. In (Breuel et al., 2013), a bi-directional LSTM model to extract handwritten text was designed. On 20,000 text lines, the system achieved an error rate of 0.6 as compared to 1.3 and 0.85 of Tesseract and ABBYY, respectively.

According to this paper (Esser et al., 2013) designed an approach to find automatic indexing of documents based on the entity extraction, i.e., generic positional in terms of indexing. Finding document indexing was based on a standard full-text search as the automatic indexing of scanned documents was designed. This would help to get an easier

retrieval of an index number by finding the index number automatically, which would make the searching of any document easier than searching the whole library. The purpose behind automatic indexing was to lead a paperless office where any retrieval, exchange or related task would be performed digitally. In the case of the domain, these digitalized documents create a vital role, like tagging any document by using previously or predefined vocabulary makes subsets smaller. The subsets were used at the time of the retrieval process, which makes it easy to find out the document. As the main objective was to extract data in terms of dates, amount from the business document, and use it further for indexing. Though automated processing can be done, they proposed a graphical approach by using data positions of indexes that already exist in the database. However, the problem was the indexing error that arises during the data search by the user. So, they have prepared a template where already documents were indexed. According to their method, it would cluster the entered data and store it according to precision, then the indexed data position of the cluster would be used in data extraction. Usually, any company generate business documents by predefined template and fill them with relevant information. The flow of their model towards document extraction via an information extraction system follows three basic functions. After extraction of the document, template detection was carried out, which leads to templates documentation and index data extraction. Finally, the extracted data will be sent to the user for feedback purposes. They performed the task using the template detection method where the documents can perform queries along with search index. Wordpos describes document feature types along with relevant word positions as calculated by the OCR. These combinations will lead to a much simpler extraction process. Their solution process is only applied to the data, which are independent of structure. Extraction of amount, data, doctype, docnum, recipient, sender, subject from 4000 static documents while comparing mobile vs scanned gave F1 score from 81% to 73% in mobile capture, i.e., reduced in OCR in case of mobile. Their future work was to combine their template approach along

with the text-based extraction method to get pros from both techniques. For improving extraction results, they will follow self-learning systems.

Furthermore, in the following paper (Bartoli et al., 2014) has developed a template-based data extraction system based on semi-supervised wrapper choice. The idea was to select a template or wrapper based on the source. The PATO system first executes a classifier (KNN and SVM) to define whether the wrapper is known. When no wrapper was found, it shows the document to a human operator, which then uses point and clicks GUI based selection to help in creating a new wrapper. When a wrapper was identified, it triggers data extraction from invoices based on identified wrapper choice. According to them, the wrapper-based concept has gained considerable attention in recent years for data extraction from online sources. In online sources, it was much easier as the syntactic structure of HTML help a lot. However, in printed invoices, the scenario was different. Firstly, this was due to structural differences; the image has a flat set of blocks that has geometric information and textual information only, for example, block width, position on the page, text content, height and so on. In the second part, the image sheet contains a lot of white noise, both in textual and geometrical features. These are due to OCR conversion errors, sheet misalignment, stamps, staples, and other irregularities. Furthermore, the authors have also addressed other features applicable to online information extraction that may not work well with printed data. The invoice dataset test includes schema with nine elements: invoiceNumber, date, total, taxableAmount, vat, customer, rate, issuerVatNumber and customerVatNumber. Based on the test carried on more than 600 datasets, the human processing time was tested to be reduced by 66 seconds. The extraction accuracy achieved on average was around 88.91%. No gap was being discussed in this paper. Additionally, general fields were targeted. No table-level extraction was evaluated. The result looks good but still require a lot of human intervention to add more templates. Furthermore, enhancement in the form of better invoice classification can be done for better data extraction.

(Cristani & Tomazzoli, 2014) have discussed the automatic document classification process. A new approach of linking image and text-based contents together has been proposed instead of normal tagging for classification. The test was conducted on the dataset of around ten years of daily issues of local newspaper articles, each having 64 pages with an average of 4 articles per page. Based on a total of 800,000 documents, the results declared were better. Both image and text gave an F1 score of 0.0428. Additionally, no future work or gaps were discussed. Furthermore, the test data set was related to research articles; whether this works on invoice data extraction is still an open topic. Another paper (Klampfl et al., 2014) was studied related document classification of research articles by analysing using a variety of unsupervised machine learning techniques and heuristics, as well as meta-data extraction; a PDF document was developed without any pre-trained model. Here too, the structure of the articles will be fixed as there are certain predefined sets of the publisher in the world. At the same time structure of the invoice differs.

An ANN was a mimic of the human brain, which imbibes the learning pattern like the way the human brain works. ANN has several neurons, like the Feed-Forward Neural network, that work forward without a feedback mechanism. The ANN has three layers such as input, hidden and output.

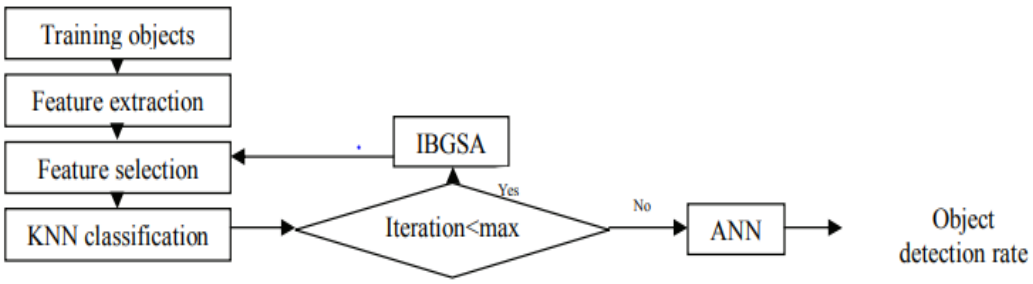


Figure 11: IBGSA method feature selection flow chart Source: (Pourghahestani & Rashedi, 2015)

(Pourghahestani & Rashedi, 2015) proposed an improved binary gravitational search algorithm and ANN model for the detection of objects from image data. To segment the

images and extract the object, the Watershed algorithm was used; KNN was used as a classifier. For the experimentations, only six objects were some hand tools like nipper, meter, spanner, tongs, and screwdriver. The recognition rate of this model was 63.82%. This approach in future would be used in the robotic application. The fast R-CNN (Girshick, 2015) and SSD (Liu et al., 2016a) was used to address the text detection problem, but still, there lies a gap in test detection and extraction. They tried to address some of this gap and understand what has been done to overcome the challenges faced. RCNN works by creating a bounding box on images for object detection. For creating a large amount, the regional proposal model was used which uses a selective search algorithm (Uijlings et al., 2013). The classification was finally done by using the SVM. Faster RCNN only speed up the process by adding a technique called RoI pooling.

According to (Pitou & Diatta, 2016), before the text extraction process can begin, text localisation, place determination, and recognition or text (identification) were required. So, a spatial positioning method (i.e., a method for determining two or more objects' positions relative to one another) was devised. Data segmented by the K-means cluster algorithm was used to create the prototype region. The formal context was derived by establishing the prototype region by using the concept of lattices (i.e., the probable relationship and connection between different components). For the data set, thousands of one-page, A4 sized, and scanned invoices amalgamated by 18 service providers in the auto repair and towing industry were used. The assumption for the region-based extraction approach for finding synthetic data sets was to consider the A4 document in 4 coordinates which is not possible in the real world. They have also mentioned and was not sure about the region (A4 document in terms of height and width, i.e., x & y pixel) representation in four coordinates, so here the approach itself is confusing, so there is less surety for getting synthetic data sets by using region-based extraction approach here. For textual information extraction, a free OCR engine named Tesseract was used, which cannot recognize the net amount in invoice images mentioned in the paper also. Tesseract is available under the Apache 2.0 license. It could extract printed text from images and

works with a broad range of languages. Tesseract may be used for a wide range of programming languages and frameworks. The experimental result was not as accurate as adding an unjustified and assumed pathway, which only enhanced the per cent from 2-10 in between correct information among a total number of detected and sought information. Although all the conditions were not performed, they have only considered the assumptions, but those were not efficient for text extraction by using the lattice concept.

In the next paper (Henge & Rama, 2016) proposed a fuzzy neural hybrid system methodology for the classification of connective consonants (along with numerals) and storing of outcomes for identification. They designed five layers of the fuzzy neural closed-loop feedback system, which was a hybrid system for the identification process of consonants and numerals. The process gets started by getting the characters of the input image, which was classified in two ways, i.e., the first method depicts general consonants, while the second method depicts conjunct consonants. The neuron was performed as input by the fuzzy neural hybrid closed loop method. This method carried out the extra set of tracing, which helped identify in-between mixed connective consonants controller. Their proposed algorithm was intended to gain versatility while implementing the data flow in constructing an OCR system. The algorithm can scan the text layer-wise with different directions and orientations. Though it was a hybrid system, the first will identify mixed hand-written letters and other consonant characters, and the second will identify low-quality written conjunct (mixed). As the five-layered methodologies of fuzzy neural hybrid, a closed-loop can only be implemented using MATLAB and LabView VI GUI. Their future work was to improve the mixed and non-mixed conjunct consonants' classifier to achieve a better recognition rate.

In this paper, (Kooli & Belaid, 2017) designed an entity recognition system for documents that was completely based on OCR image document recognition. The proposed system depends on the graph matching technique, and the database was designed to describe the entity records. Here, input documents were labelled by entity attributes. Identification of labels was categorized by score leads to select candidate entity set. Each entity label was

modelled by a structured graph. The graph method was used to match it with the modelled graph. These graph matching functionalities were useful in integrating dissimilarities in features. Then the validation process was carried out to segregate the mislabelling and recognition task. Here, the datasets show variations between 88.3% and 95% for recall and between 94.3% and 95.7% for precision. Their future work is focused on the position of local structures as guides for searching along different types of string-matching combinations to enhance the recognition process. (Rigaud et al., 2017) tested the OCR system for text extraction from comic books. The results were better than open-source Tesseract. Furthermore, the idea of this paper was to target and test text extraction in comic books and how the existing methods work. Later, (Shaker & ElHelw, 2017) proposed an OCR system for multi-class labelling, which used the recurrent attention model to localize individual digit and deep convolutional neural networks for actual character recognition. In this work, the SVHN dataset was used to recognise individual digits and incomplete street numbers accuracy.

Furthermore, (Zhang et al., 2017a) studied a new lossless function for training CNN based model to solve low image resolution problems in OCR text extraction. The test results gave an accuracy of 78.10% on the ICDAR 2015 TextSR dataset. The method assigned more weight to the edge regions, allowing the CNN to focus on high-frequency image details. It does an effective image padding and conduct model combination for improving the performance. The planning was to enhance the model. The study between these two techniques by using tree-based data fusion was to predict the data. (Griffin & Kurup, 2017) has combined two techniques, i.e., regression trees and model trees. They designed such a hybrid model to develop the decision level, fusion techniques, and bootstrap aggregation. After giving training and by testing, those predicted values for OCR were implemented in interpretation methods. Due to the fusion, it enhances the decision level of the regression trees. Overalls they found less errors in OCR due to the fusion of models. In real life finding scene text in images have many applications. Starting from license

plate detection to visually impaired people, these scene texts was playing a vital role as image consists of so much deprivation like blur image type, noise, odd lighting etc.

In this paper (Dong & Smith, 2018) have discussed a sequence-to-sequence model. By applying the attention method for single-input correction and a new decoder method with multi-input attention to the correct post-OCR error was implemented. Based on 1.7 million text lines, they achieved the character error rate and a word error rate of single and multiple decoding as 0.11 and 0.09, respectively. Even (Mei et al., 2018) system was used for error detection and correction. The model consists of the following submodels such as candidate correction generation for the detected errors, candidate ranking and error detection from text based on the features. The dataset used was Birds of Great Britain and Ireland, which has 274 pages. The results of this method were achieved with the precision value of 99.04 and recall value of 98.98. The future scope was to increase the features to achieve optimal performance. Later, (Coustaty et al., 2018) proposed the regression approach for post OCR text detection. Various fields in the regression model include weighting and candidate generation based on an adaptive edit distance. The candidate generating, candidate ranking, and candidate scoring used the Language Model (LM), and candidate ranking is also used for feature extraction. For evaluation, ICDAR 2017, 666 training documents and 41 testing documents were used. The network gave 43% better results than the other 4 top companion teams. (Arroyo et al., 2019) proposed CNN based model. The first step was a text map being a generation using the OCR techniques, and then this generated map was fed to the CNN model. The dataset used for experimentation was Nielsen Brandbank; out of that, more than 10000 images were used for training and 2000 for testing. The research has been done only for specific cases. Later, in the next paper (Guo et al., 2019), proposed Entity-aware Attention Text Extraction Network (EATEN) to extract entities from images without doing the post-processing. The network consists of a CNN model to extract the features, entity-aware attention network was used to find the layout of the image and decodes the contents.

Almost 0.6 million images in three real-world scenarios (train ticket, passport, and business card). The mean entity accuracy value was 95.8%.

Furthermore, (de Jager & Nel, 2019) have discussed Business Process Automation (BPA) to solve the problems of industry related to key-value text matching. The predefined labels of tax characteristics that were related to key/field names of the company were used for processing. The values/field-values were the relative tax values corresponding to a specific key. The proposed project had three steps (de Jager & Nel, 2019):

1. The text from the tax image was read using GCV OCR. The ground truth image was manually developed. Exact string matching (ESM) was used for matching the existing file data and the GCV file output.
2. Index pairing was introduced to improve results in cases with different image type formats.
3. Finally, Approximate String Matching (ASM) was introduced to decrease the impact of OCR reading errors.

For the evaluation, publicly available 19 tax clearance certificates were used that was gained by registering with organizations from the South African Revenue Service called as a dataset. The documents were scanned and stored in JPG and PNG format. With the help of the third step that was ASM, the results of the model achieved were 90.06%. A potential gap in this research deals with its relative weakness with field-value identification for the regular expressions.

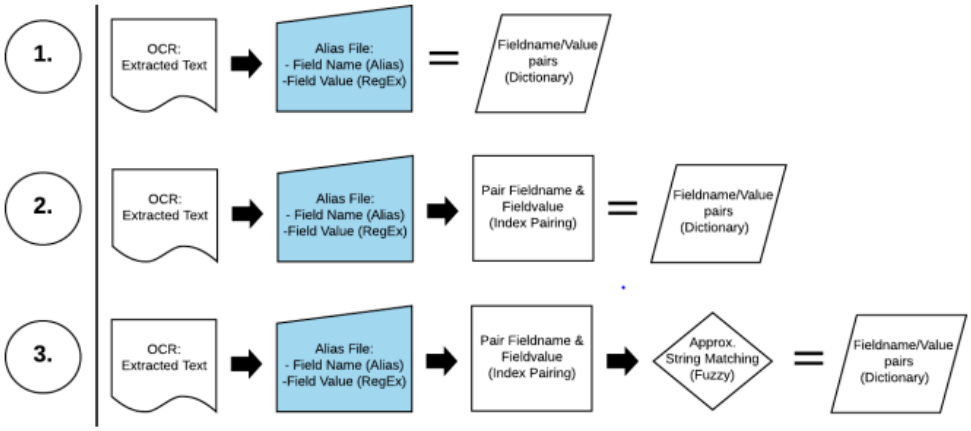


Figure 12: The flowchart of accuracy measurement with the latest functionality. (de Jager & Nel, 2019)

Article (Grönlund & Johansson, 2019) have applied Faster RCNN to detect detection and OCR on steel. The results were promising apart from the issue in reading some characters which were read incorrectly. In this paper (Arora et al., 2019), a DNN- / HMM-based model was designed, which has an open vocabulary sub-word text recognition system. In this research, they have overcome the vocabulary words and time delay issue with the help of the sub-word lexicon and sub-word language model. The test was performed on 60,000 lines of text based on the IAM dataset with an error rate of 10.0%. Later, (Agarwal et al., 2019) proposed a method to extract text from Handwritten Character Recognition (HCR) using CNN and TensorFlow. A Soft Max Regression method was used for assigning the probabilities to handwritten characters. With 4900 Sample images used, the authors achieved 90% accuracy in handwritten character recognition. They also claimed that feature extraction methods such as diagonal and direction techniques were far superior to many conventional vertical and horizontal methods in terms of producing high accuracy performance.

Furthermore, in (Jun et al., 2019), Faster RCNN and Regional Proposal Network (RPN) technology in (Girshick, 2015; Ren et al., 2015) were designed to extract text automatically and classify information from scanned images in the form of an icon, text, table noise and other objects. Based on 522 image samples (training and testing), they

have divided the methodology into the construction of RNN and RPN. The objective of using RPN was to reduce the number of late anchors boxed, let say from 6000 to 2000, thus increasing the calculation speed. After this, classifier layers were constructed based on ROI pooling. Finally, the result was divided into three categories (stamp, character, and page number). It was analysed that the positioning of stamp and page numbers were accurate, and the score was extremely high. However, due to a large amount of text. Few places were missed. Therefore, text detection is still a gap, and the identification of characters with accuracy remains a challenging question. According to (Purwantoro et al., 2019), OCR error correction was performed on the old newspaper by suggesting the new words if the error was found. The error detection task was completed with the help of the dictionary KKBI. Two approaches were used for error correction and detection as isolated word-based and context-based error. For evaluation purpose KOMPAS articles between 1991 and 2017 were used. The results obtained were 95.73% as correct words and 87.68% as errors. In (Jatowt et al., 2019), the novel approach for error detection in the case of character level and word level with the help of features was developed. The experiment analysis was done on the ICDAR 2017. The F1 Measure for monograph value was 79%, and the periodical value was 70%. The plan was to improve the network, which gives better results in terms of error correction. The study of (Paliwal et al., 2019) presented the TableNet approach to detect table and structure recognition. The pre-trained VGG-19 architecture was used. The rule-based semantic approach was used for row extraction. For the evaluation, the datasets such as ICDAR 2013 and Marmot are used. The F1 score achieved for the TableNet was 0.9151.

(Weng & Xia, 2019) designed an image processing module for mobile devices based on the characteristics of CNN for text extraction. They trained CNN with greyscale images for feature extraction. They developed a traversing features methodology by calculating the similarity between texts. Based on the test performed on 50,000 data samples of images, an accuracy of 93.3% on character recognition was achieved. The tasks of image processing can be divided into three categories:

- Image acquisition, storage, transmission: This step involves the digitisation of the image, the resizing of the image by compression, and the performance of encoding or decoding (if required).
- Enhancement and Restoration: Sometimes, the original invoice might be blurred or smudged, leading to an unclear scanned image. In this stage, the image is processed or enhanced using various tools like smoothing, removing blur, de-skewing, and noise reduction. To improve the readability of the printed text, the image is converted to greyscale using an image processing library, and an applied threshold increases the clarity of the image. Through this process, the image becomes easier to read, and text recognition becomes more accurate.
- Information Extraction: The last stage is extracting information from the image. The information could be text, numbers, or image characteristics. This information is then stored for further analysis.

Furthermore, the same algorithm was applied by (Yang et al., 2019) on handwritten recognition on 200 images, with an accuracy as the average character segmentation rates of the word up to 99%, the average character segmentation rates of the letter up to 95%, and average character recognition up to 97%. Although handwritten recognition of text is not in our scope of the study, still its looks interesting to know that the same methodology can also be applied to extract handwritten character. Furthermore, (Liu et al., 2020) have discussed table detection in document analysis. It uses Faster RCNN with a featured pyramid structure. The test was performed on ICDAR, UNLV and TableBank datasets. The F1-measure of 92.59% in UNLV was achieved. As such, Faster RCNN seems to be the best choice for table layout detection and to perform better data extraction from tables. Furthermore, according to (Arora et al., 2020), a model was implemented with three networked such as OCRXNetvv1, OCRXNetv2, and OCRXNetv3. In the OCRXNetv1, basic image processing techniques and Tesseract was used. The OCRXNetv2 was based on object detection technique, and OCRXNetv3 used two models, such as EAST (Efficient and Accurate Scene Text) detector and CRAFT (Character Region Awareness

for Text detection) in the pipeline. For the experimentation, they used identity documents like Aadhar card, PAN etc.

This paper, (Ast, 2020) used the knowledge-based approach where the document was converted into a semantic image. The semantic image was used for further processing, where RNN was used for extracting the regions from the semantic image. The extracted regions were classified based on the textual map. Later, in the next paper, (Geetha et al., 2020) worked on text recognition and text extraction on formatted bills using deep learning. The proposed deep learning model was used for text detection and extraction by using the EAST algorithm to investigate the letter and word from images or scanned documents into machine-readable form. The overall architecture was divided into the three models such as pre-processing, text detection and text recognition. For the real-time analysis, the Open-Source Computer Vision Library (OpenCV) was used. This proposed system aimed to investigate how to convert handwritten text and invoices into an excel format. They have not done the result analysis, nor they have shown any experimentation results. (Geetha et al., 2020):

A. Image pre-processing

In the image pre-processing, unwanted data was removed from the image by applying the threshold function. The original image colours were converted into grayscale for the operation. A noise removal procedure was initiated to improve the image quality, thus simplifying the segmentation process.

B. Text detection

To detects the text in the image, the text detection network was used for further recognition. The EAST detector used for text detection implemented four layers: an input layer, a pooling layer, an output layer, and a convolutional layer.

C. Text Recognition

In this module, the EAST algorithm's output feature map was used to recognize the test using RNN. For a single timestamp, the feature sequence had a total of 256 features; the useful information was passed through this sequence by RNN (Geetha et al., 2020).

(Poncelas et al., 2020) described a tool called the Language Model (LM) that could be used after the Tesseract to correct errors by suggesting alternate words. The dataset they used was "An Essay Towards Regulating the Trade and Employing the Poor of this Kingdom", which has a total of 576 lines and seven words per line. In total, 63% of errors contained in the dataset were corrected. Future work on this model should seek to address its research gaps by developing a method for automatically identifying characters and improving the LM model in large data sets.

In this paper (Anand et al., 2020) proposed Loan Application Process (LAP) documents that use the deep learning model titled Loan Operations Data Extraction System (LODES). This model extracts the required information from the large document. The accuracy of the model was 100%. The future scope was that bug reports should be improved for business documents. Furthermore, (Pham et al., 2020) have developed a system as a pipeline structure, which consists of data pre-processing steps like to deskew scanned documents (for that Hough transforms algorithm was used), table detection to detect the vertical and horizontal lines or table structure, and document layout analysis (for that X-Y Cut algorithm was applied). The evaluation uses the Vietnamese documents dataset, which consists of 120 documents that contain 40 images with table and 80 without table. The result Similarity score was enhanced by 0.23. Later (Yu et al., 2020), the deep learning-based algorithm was used due to huge achievement in OCR. The Key Information Extraction (KIE) architecture was implemented, consisting of the encoder, graph model and decoder. KIE handles overly complex layout documents and extracts the textual as well as visual features from the documents. Different datasets were used, such as Medical Invoice (2,630 images), Train Ticket (2k real images and 300k synthetic images), SROIE (626 receipts). The mean entity F-1 (mEF) was 98.6% for train ticket dataset. Later, (Martınek et al., 2020) proposed, "Building an efficient OCR system for historical documents with little training data". The proposed method was divided into two steps as page layout analysis and OCR. For the segmentation, Fully Convolutional

Networks (FCN) was applied, and for OCR, RNN was considered. The dataset was created from Porta fontium portal. The research was only limited to historical documents. Furthermore, (Yindumathi et al., 2020) designed a method that was useful in the case of alphanumeric recognition. Machine learning-based algorithms were used to extract the text such as CNN. The paper research only identified the challenges, and no new method was presented.

(Shehzad et al., 2020) have presented a paper, "Named Entity Recognition in Semi-Structured Documents Using Neural Tensor Networks". This paper includes feature engineering to perform layout-specific extraction of information. They implemented the method without considering layout documents to extract the specific data. The Continuous Bag-of-Words (CBOW) architecture was used to convert words into a vector, or simply the output of CBOW was derived in vector format. A neural network tensor was different from the Fully Connected Neural Network (F-CNN), as the F-CNN uses the two-dimensional weight matrix, but the neural tensor uses the three weights such as value, label, and entity vector. They have got an average accuracy of 92%. In future work, they would test the method for a variety of documents such as invoices and tender documents to check the performance and effectiveness.

In this paper, (Rabbi et al., 2020) proposed an enhanced super-resolution GAN (EESRGAN) generator that uses RRDB (residual-in-residual dense blocks), a discriminator that uses the VGG19, and a detector that uses the Faster R-CNN and SSD. For experimentation, the OGST dataset was used, and the average precision of the network was 95.5%. In the future, they would work on the creation of more accurate Low Resolution (LR) images for training. A multi-level residual network with dense links exists in the RRDB. During the training process, dense links are used to maximise network ability and residual scaling to avoid unstable conditions. The generator network was trained on two losses such perceptual loss (Lpercep) and content loss (L1). The discriminator was implemented by using the VGG19 architecture. The discriminator network was trained with the help of two-loss functions, classification (Lcls_frcnn) and

regression (Lreg_frcnn). (Rabbi et al., 2020) also used the ResNet-50-FPN as the backbone network for Faster RCNN. In the SSD module, the VGG16 was used as a feature extractor. They have trained networks two way, separately, or jointly in an end-to-end way. In the separate training, the generator and discriminator network were trained together, whereas the detector was trained separately. The detectors loss was not backpropagated to the generator model. The generator has only feedback from the discriminator network. Furthermore, they trained the whole architecture at a time; it means the detectors loss was applied to the generator model. This means the generator receives the data from the discriminator and detector.

Recently, based on the success of GANs in image-to-image translation and super image resolution, it has been used in many research works. Such as (Zhang et al., 2020b) used the DetectGAN for text detection. The generator model was implemented using the UNET architecture. For the experiment analysis, DTDR and SROIE datasets were used. The F1 score for the SROIE dataset was 98.74%. For character recognition, they used convNet architecture (Zhang et al., 2017b). In future, they have planned to design to integrate DetectGAN with the text recognizer to correct each other simultaneously in the detection and recognition phase, which would improve the performance.

Generator model:

They implemented the generator network using the UNET model. The network has U shaped Encoder-decoder unit. The input image is downsampled five times, and a feature map was collected. The output of the generator model has a text score map and the direction map.

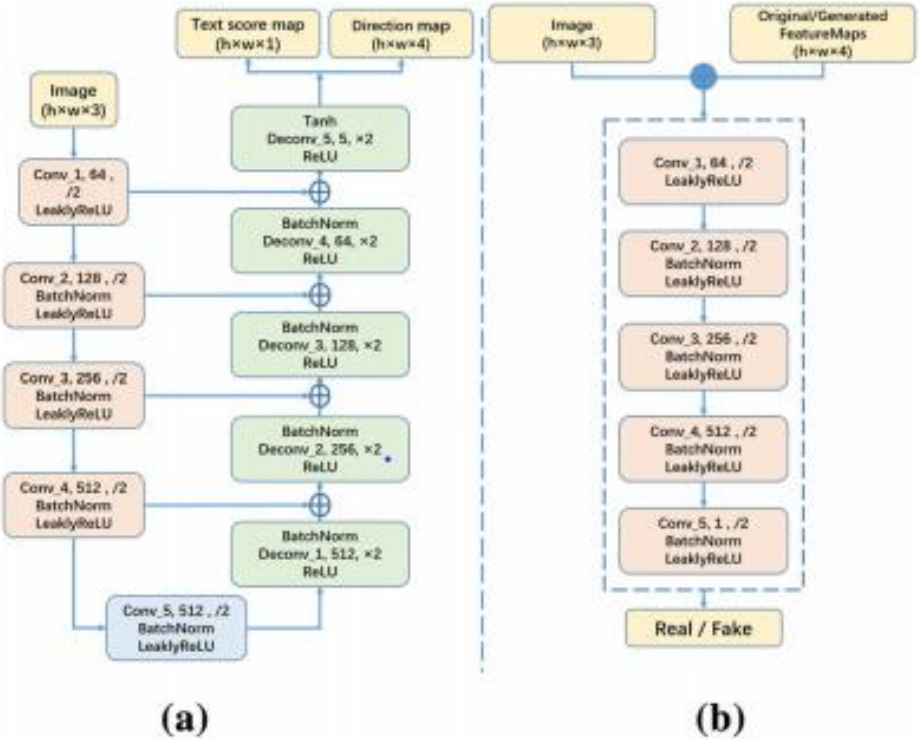


Figure 13: The overall architecture of DetectGAN where a) represents the Generator framework and b) represents the Discriminator model. (Zhao et al., 2020)

Discriminator model:

The discriminator model was implemented with the help of patch-based FCNN. The discriminator model was able to distinguish between the directional score map generated from the generator network or the original image.

Training of DetectGAN:

For training DetectGAN, training and validation images (of size 256x256) were cropped with stride to 64 pixels. The Adam optimizer was used with a learning rate of 0.0002 and mini-batch size SGD.

(Zhang et al., 2020b) designed the Multi-task Generative Adversarial Network for Detecting Small Objects in the wild. The end-to-end Multi-Task Generative Adversarial

network (MTGAN), where the generator network would be able to generate the super-resolution image. The discriminator unit uses the patch-based convolutional neural network. For the evaluation, COCO and WIDER FACE datasets were used. The generator network uses the five-residual block with kernel size 3x3. The activation function used was RELU. The generator used the pixel-wise MSE loss to calculate the difference between the ground truth image and the image generated by the model. The Adversarial Loss method was used to generate a realistic image. ResNet-50 network was used in the discriminator network. The architecture has ResNet, three fully connected layers, and the last average pooling layer, which was the backbone of the model. This model was trained on Classification Loss and Regression Loss.

In a related paper, (Wang et al., 2020) used Generative Adversarial Networks-Knowledge Distillation (GAN-KD) for the one-stage object detection. The generator network has ResNet50 as its student net and ResNet101 as its teacher net, whereas the discriminator model uses the SSD-HEAD network. For the evaluation, PASCAL VOC 2007 dataset was used. They achieved a performance gain of a 5% map on the COCO dataset model, and the proposed model was compared with MobilenetV1.

(Liu et al., 2019a) proposed the GANs for Small-Data Object Detection. They proposed the DetectorGAN, in which the generator uses the ResNet, nine blocks, and the discriminator uses PatchGAN. The RetinaNet detector was used to generate the real and synthetic labelled images. For the experimentation, the NIH Chest X-ray dataset was used. They have improved 50% of the localisation accuracy.

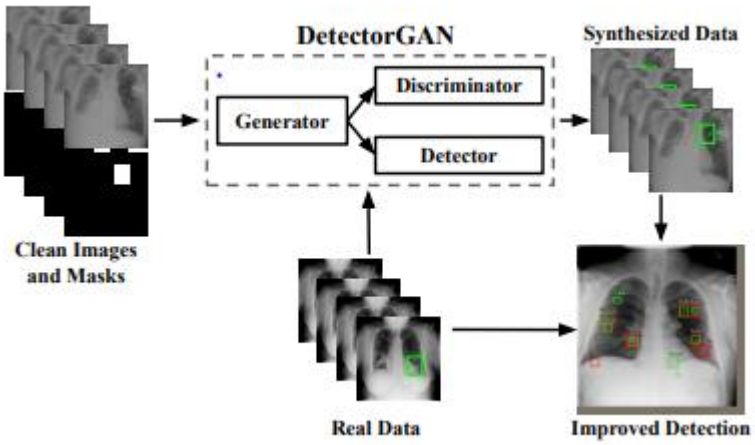


Figure 14: DetectorGAN example. (Liu et al., 2019a)

In this paper, (Charan & Lina, 2019) implemented GAN-DO for Detection of Objects on Images with Varying Quality. The SSD300 and RetinaNet50-400 were used for feature extraction. The PASCAL VOC2007 and PASCAL VOC2012 were used for evaluation. They achieved 68% accuracy in the case of low-quality images. (Wang et al., 2019) designed an open event extraction from online text using a GAN. In this paper, they have addressed the limitation related to Bayesian graphical models. To overcome that limitation, they introduced the Adversarial-neural Event Model, which is an event extraction model based on Generative Adversarial Nets (AEM). To capture the patterns underlying latent events, the generator model was used with the help of Dirichlet. Moreover, a discriminator was used to distinguish documents reconstructed from the latest events and the original documents. They have evaluated the model on two Twitter datasets, such as the FSD dataset, which contains 2499 tweets, and the Twitter dataset has 1000 tweets and a news article dataset from the Google dataset, which has 11,909 news articles. They have got a 90% recall rate for the news articles. They intend to investigate integrating external knowledge (for example, word-related information available in word embeddings) into the learning system for event extraction in future research.

Later, (Kundu et al., 2020) proposed Text-line extraction from handwritten document images using the GAN model-based network. Based on the study of the existing system, they found that when document images have several overlapping words/characters, distorted text lines, and non-uniform inter-line space, methods usually fail. To solve these problems GANs model was used. They implemented the generator network with the help of U-Net architecture and PatchGAN architecture for the discriminator network. For the experimentation, the HIT-MW dataset and ICDAR 2013 Handwritten Segmentation Contest datasets was used. They achieved the accuracy of the proposed model as 99.63%.

Recently, (Pegu et al., 2021) designed an end-end framework for Visually Rich Documents (VRDs) to extract the information which would be used further for verification and compliances; the proposed framework had five steps such as, number of sequence detector, QR code detector, text identifier, logo detector, and corner detector. For the evaluation purpose, a publicly available in-house dataset with 1500 card types was used. The system gave 99% accuracy. The future scope was on testing that should be possible on other real-time products. Finally, (Tian et al., 2021) designed the Faster RCNN for Financial Ticket Faster Detection network (FTFDNet). The FTFDNet method uses the Resnet101, Inception RPN, ROI pooling. For the evaluation, 4184 VAT tickets and 16146 ROI regional data was used for the detection and recognition model. The recognition accuracy of the model was 97.4%.

Table 7: Gaps Identified: Non-Invoice Document.
"Devised by the author".

Research Gaps	**References**
Unable to recognize when characters were remarkably close	(Zidouri, 2004)
Only capable of recognizing English letters written in black font with white background	(Laine & Nevalainen, 2006)
Not capable of handling complex background, merging of multiple images	(Zhang et al., 2009)
Fails at the time of the field-value identification for the regular expressions	(de Jager & Nel, 2019)

In case of a large amount of text, few places were missed, detection and identification of character remains a challenging task	(Jun et al., 2019)
automatically identifying such as characters and to improve the LM model in case of large data	(Poncelas et al., 2020)
Image resolution was the main issue in text detection	(Rabbi et al., 2020)

2.3.5. Invoices Document

In this paper, (Kosiba & Kasturi, 1996) proposed a combination of the graphical features of image and text analysis as a way to analyse the structure of a page. The first step dealt with the identification of line intersections with blocks found in the image and, later, keywords such as item number, quantity, and total were searched. Connected component analysis was used to search for a valid keyword match. The invoice system consists of two parts, classification of both known and unknown document types and data extraction. If the documents type is known, then data extraction is done directly. If it is not known, then a detailed analysis is performed. An issue like a rounded box corner might be challenging to identify. Incorrect keyword to box position correlation might also lead to issues. In the future, an updated model needs to be created to accommodate new invoice features and types of analyses. Additionally, non-box-based invoices need to be researched.

In their article (Bayer & Mogg-Schneider, 1997) offered a solution for extracting data from invoices. The authors did not consider template-based extraction rather than the data extraction from invoices whose formats were not known in advance. This system consists of an OCR tool that only extracts text from images with relevant meta-information of that text and an information extraction model named Frame Representation Language for Structured Documents (FRESCO). According to the authors, invoice data extraction cannot be solved by knowing geometric layouts as invoices significantly vary in their layouts. The test conducted was based on 497 invoices related to a healthcare company. After the OCR, the text was sent to FRESCO, which extracted data in two forms., key-

value and table line items. For table line items, the header was searched in the text, and then data below the header was considered as a value for that column. Based on the test conducted, the recognition rate of key-value fields was 74% for the amount field and between 25-64% for the rest of the four fields, i.e., id, diag-code, birth, and period.

Furthermore, for table line-item extraction, accuracy was around 23.9% only. It was also found that most of the error was due to OCR errors caused by incorrect segmentation or incorrect classification. Whereas in the case of a table, it was either due to the missing description or missing table entries. Only 92 invoices were recognized correctly, i.e., a 20% recognition rate. However, the authors claim that this 20% accuracy would correspond to 98.3% if an average of 96 characters' recognition is considered. As such, the error rate of the FRESCO system is below 1% only. From a processing time point of view, the system took 4.83s to process each document. In future, the system was further planned to be optimized in terms of accuracy and run-time rate.

The following paper (Köppen et al., 1998) was designed to detect and recognize the price printed in the invoice. The proposed model had several stages like text stripe extraction and detection. Text stripe extraction uses a multilayer backpropagation network which acts as a classifier unit. For the detection, the genetic algorithm was used, and recognition was done with the help of OCR. The dataset used 200 wholesale invoices. They have got a 93% OCR recognition rate. They would work in future on the challenges faced in real-time such as quality, handwritten remarks, stamps, a small number of entries in the table. Furthermore, text extraction is becoming the most challenging and essential need to make the digitalisation process more effective. (Cesarini et al., 1998) proposed a flexible form reader system that was used to extract textual information from documents like bills or invoices. To categorise the type of invoices and the easier retrieval process, they designed a model consisting of relational graphs. This kind of model was carried out by a hybrid model. This hybrid model was based on morphological operations, connected components and instruction fields. In the end, they have connected it with connectionist models. The document layout which they used was attributed relational graph. The

presence of such an attributed graph gave the user the ability to specify document structure. Here, the document structure was being described by only the features of the object used in the registration and information field of any location. For adding more unique activity, they used a registration technique that was completely based on a verified paradigm. This solution leads to provide a variety of layouts, which contains objects like logos, keyword, or lines. The location of the keyword was caught by associator neural networks along with the OCR approach. The words were recognized by way of string edit distance done by dynamic programming. Overall, this approach shows maximum accuracy that leads to the further improvement of such a model.

In the following paper, (Kieninger & Dengel, 2001) proposed the T-Recs table recognition method to recognise block segmentation, location and structure analysis. This system works on the output of OCR, which provides the word bounding boxes geometry to the text as well as T-Recs++ prototype was used for location and to check performance quality. The authors (Delie et al., 2002) designed a Chinese financial invoice feature recognition system. A linear whole block moving method for vertical line segmentation and for slant lines fast algorithm was used. The blank invoice form was generated from the recognized invoice features for real-time. In (Belaïd & Belaïd, 2004), a morphological tagging approach for document image invoice analysis was developed. After performing POS-based analysis on 276 invoices corresponding to 1704 articles, the accuracy achieved was 91.02%. In this paper, (Chien & Lin, 2009) designed the system, which consists of two parts such as invoice number detection and candidate invoice number segmentation. By using the various thresholding techniques, the invoice number was detected for example adaptive thresholding, and HSV colour model. The candidate invoice number segmentation, the morphological operations, run-length smearing algorithm (RLSA) are applied to fill the gaps. For the experimentation, 191 invoice images of resolution 640×480 were used. The performance of the system with correct extraction was 98.42%.

Later, (Liu et al., 2016b) used the bag of words concept for business invoices. The SVM gave better results than the Naive Bayes. The training error was 8.89%, and the testing error was 13.99%. In total, 97 raw invoices were used, and from those, 8000 features and 2095 tokens were generated. The future scope of research will be to collect high-quality invoice images. In another paper, (Alippi et al., 2005) proposed the automatic invoice document classification system for office documents. The classification system was used for the analysis of the graphical information of that document and classified into two separate groups, such as open worlds, which indicates some of the classes will vary as per operation, and closed world means number of classes were fixed. The evaluation of invoice documents of real company correct classification was 99% in case of closed world and 79% for an open world.

Considering the error correction perspective, the article (Agarwal et al., 2007) has studied noise in the different textual sources. “Spelling errors, abbreviations, non-standard phrases, false starts, repetitions, missing punctuation, missing letter case details, pause-filling words (like um and uh), and other text and speech disfluencies were common in the text generated under such conditions”. In context to OCR, the result was not great. Accuracy was reduced at every level of noise added or if image quality was less. Noise test at four different levels, 0%, 40%, 70% and 100%, did not impact severe change as long as corpora of words were large enough. Even 40% noise did not affect the classification. According to the author, meticulously designed label sets are needed to overcome label noise. This review identifies the need for enhanced evaluation capabilities for real-world data sets like quickly written notes or summaries and improved attention to label noise as gaps in the current research.

Furthermore, (Hamza et al., 2007) implemented a case-based reasoning approach for invoice structure extraction and analysis. The CBR-DIA method works without pre-trained data by analysing similar documents in the database and executing a graph probing and editing distance program. The authors achieved an accuracy 85.29% over 950 invoices documents of known classes and 76.33% for documents of unknown classes.

Subsequent work was done to add database indexing functionality with the goal of fetching similar documents in a short amount of time. This concept was widely used but underperformed on real-time invoices due to too many variations when a new invoice format was presented. Additionally, case-based is more like rules-based and can be used as an alternative approach after the ML technique was already applied. Furthermore, (Bart & Sarkar, 2010) worked on repeated structures from the document which was extracted, such as names, price, and quantities. They used a probabilistic framework that extracts the repeated information. For the experimentation, 10 synthetic invoices for training and 15 invoices for testing was used.

In this paper, (Medvet et al., 2011) proposed the first step in identifying problems and challenges in data extraction from scanned invoices and receipts. That "printing documents lack any syntactical structure: they are made up of a flat collection of blocks with only textual and geometrical characteristics, such as page position, block width and height, text material, and so on". Furthermore, due to sheet misalignment, OCR conversion errors, staples, stamps, and other factors, the representation of a document obtained from a paper sheet typically contains some noise, both in geometrical and textual features. The authors proposed an approach for data extraction from printed documents based on probability by applying maximum likelihood in finding a similar document. Later, using the semi-supervised approach for classifying the document was performed. If a completely new document class was found, then a human operator can help to update the document class or create a new class. The operator merely uses a GUI based screen to point and click on the required text key-value pair. Later, this clicked position is saved as coordinates for futures document classification. This system was put to the test on 807 multi-page printed documents from three separate domains: invoices, patents, and datasheets for electronic components and was further divided into 85 different classes. The result achieved was effective, even for smaller training sets. The main findings show that the success rate tends to be more than 90%. Although a final check is required by a human for data consistency and integrity, it still reduces the effort required to enter data

into the system by a good deal. Additionally, less time would be required to rectify incorrect data.

According to (Shin, 2012), a novel invoice feature extraction model was designed, which validate the information automatically in the form of the table header detection and classification of columns. For the experimentation, they used 11 datasets which consists of a total of 2,307 invoice images. Furthermore, to add to that, (Sorio et al., 2012), in an information extraction system, suggested a method for automatically detecting and correcting OCR errors from printed invoices. The algorithm proposes corrections based on domain knowledge of potential character misrecognition; it then uses knowledge of the type of derived information to perform syntactic and semantic tests to validate the proposed corrections. The proposed solution could achieve accuracy up to 86.8% and 61.23% for datasets D1 and D2, respectively, based on the real-world data set. Although the result was good, a critical comparison with research methods was not made to get a better evaluation. As future work, the authors proposed a further level of error corrections. Later (Chu et al., 2014) proposed an E-Invoice Framework for the taxation system based on a data-oriented structure. The system was able to generate and transmit E-invoices with fewer mistakes and accurate invoices. The authorities have direct access to this system, which took lesser time to get the required data.

Concerning financial supply chain and cash flow management, as per (Tangsucheeva & Prabhu, 2014), “an accurate cash flow forecasting is crucial for good business management, and it is even more important when demand and credit conditions are unpredictable. A company's short-term obligations could not be met without adequate cash flow forecasting, putting it at risk of bankruptcy”. Another interesting study was conducted by (Paul & Wang, 2015) on the U.S. Department of Agriculture (USDA) to account for uncertainties in demand, supplier, and carrier bid prices. Historical invoice data detailing demand information by product type, destination, supplier, and carrier bids were analysed. Now, this may work if records are in a structured format. However, data from many others sectors and industries will be missing where invoices are in paper

format. Thus, there is a need to automate invoice management is seen clearly. Furthermore, (Gupta & Dutta, 2011) did a study on the flow of money between receipts and payment of invoices. The idea was to understand the payment schedule with the constraints of money receipt, without delay in payment and thus no payment of penalty. According to them, the problem was a bit complex, and it is difficult to define the dynamic nature of in and outflow of money. They proposed an integer programming model to represent static problems by solving using a heuristic approach. And from the insight derived from the static problem, apply another two heuristic approaches to solve the various dynamic problem. Later, (Seppälä et al., 2014) did a case study to understand the critical significance of transfer pricing for determining the value-added distribution around the globe. The authors focused on a specific product and data at the invoice stage. The writers have captured the costs incurred by a company using a simple global supply chain. The authors have shown how corporate intra-firm transfer pricing decides which business units and locations benefit. Now, assuming if all information were captured automatically than further analysis could have been done much faster.

Furthermore, (Aslan et al., 2015) designed automated invoice processing and information extraction system. The proposed solution was a two-phase process for optimisation of structure and invoice classes elimination. According to the authors, the system can handle any kind of invoices, and the result was promising. In between two phases, the first phase uses detectors for individual invoices such as SVM. The first phase continues for finding maximum entropy and produce candidates of different types. HOG is responsible for producing various types of candidates from invoice parts. The second phase was responsible for parsing an invoice. Here the invoice should be arranged in a composition of deformability. They used the PTM method for the optimisation process, which helps in handling any kind of invoices. Apart from the optimisation process, they have also proposed four different types of models, i.e., sequencing, serialisation, invoice, data tag, sender, recipient, customer tax and firm logos field models. They have reviewed all these processes and methods from different papers but did not make any comparison between

papers. In the initial stage, before applying any methods of optimisation, they only reviewed other methods. The second phase consists of a two-level invoice analysis where part based modular system (PTM) was used, and optical character recognition was used for displaying patterns. They performed an area match score to evaluate performance, which indicates the matching amount of area indicated by the system as a performance metric. Due to the PTM method, they found the system's performance in all areas as 80%. This optimisation technique was efficient as PTM works independently of the invoice classes and consider each invoice as a new one. Their future work would be extending the method with a new field of finders without making any change in the base system. The extraction was good except in the case of table extraction. For future work, different optimisation methods have been planned to increase performance.

Furthermore, in another paper, (Gao et al., 2015) proposed a new two-step verification process and compared this with an earlier method like the RANSAC algorithm. According to the authors, due to the high proportion of "outliers" - valid matches corresponding to other instances - RANSAC algorithms, which were commonly used in many one-to-one matching applications, may fail. The object of this new study was to locate document structures specific to invoices. The results were 6-14% better. The only issue was a performance where this new approach takes the most time. The test was conducted on an invoice dataset consisting of 4109 images from 249 providers. There are three main gaps in this research. First, this paper does not explicitly discuss text extraction. Second, the system does not scan whole documents, only specific regions (for example, headlines, shopping item records, address blocks, etc.). Third, the extraction of table items was not discussed in this article.

According to (Younes et al., 2015), research was carried on invoice related transactions of the Canadian construction industry in 2012, which states that a typical construction company processes 10 thousand invoices for payment annually. One of two significant challenges was the cost of the delay of invoice payments. Because of the variety, amount, and unpredictability of received invoices, ensuring on-time payment of invoices was a

difficult task even when funds were sufficient. These facts make invoices a pressing issue that must be addressed to prevent loss of profit and damaged reputations for both contractors and owners. The study aimed to define and rank bottlenecks to identify and prioritise opportunities for process improvement, resulting in a zero-overdue invoice-processing strategy.

Furthermore, (Roychoudhury et al., 2016) have developed a framework based on a rule which enables automatic verification of any document. They used real-life case studies like credit letters or commercial invoices. The maker-checker process was related to the transaction where the maker generates the transaction initialization by entering data, and the checker gives the transaction authority by verifying the data. They found that one complete maker-checker based transaction takes 2-4 hours, which was time taking. So, they planned to automate document verification by using ontological representation and extraction as that would be done by applying specific steps in natural language text. For rule extraction from natural language text (NLT), they have only represented NLT in SWRL (Semantic Web rule language) by using a reasoning engine for automatic verification also followed for extraction. In the first step, grammatical parsing of the statement rule from natural language was done by the CoreNLP tool suite, which uses annotations like Named Entity Recognition (NER). In the second step, they have accessed the parse tree in the XML file. The third step consists of partitioning the parse tree into subtrees where heuristics were applied for assertive (VP subtree for verb phrase) and conditional statements. The fourth step was extracting verb phrases and the SWRL operator from the VP subtree. The fifth step was to extract data properties and classes using n-grams, where they used 3-grams for extraction purposes. For assigning appropriate subscripts to extracted entities, they have assumed that the weight in the bill of landing should not be less than the weight limit in a letter of credit in the sixth step. Here, they should consider the reverse assumption, where the weight of the landing bill will be greater or equal to a weight limit in a letter of credit. Then, the analysis can be completed in terms of these conditions. After getting the extracted entities, finally, these

were assembled by using SWRL syntax. They got 88.57% precision, 86.11% recall, and an F score was 0.8732, which is desirable for any extraction algorithm. The future work was to use the document Verification-as-a-Service (dVaaS) framework to achieve more degree of precision and recall so that the verification time of the existing document was efficient in terms of time.

(Rahal et al., 2016a) have attempted extraction of the entity and its correction based on the token structure model generation. For the entity labelling, they used OCRed invoices. After successful labelling by combining labelled entities with the geometric and semantic relations, the generation of the token structure model has taken place. This model was used for entity extraction and correction of mislabelling by avoiding superfluous tokens (identified in labelling). The steps they followed for complete extraction of entity are invoice image, which would be processed by an OCR engine, after processing of images, entities will be labelled in OCRed invoice images. After the detection of the label, the local structure of each entity will be detected. However, they have not mentioned the process of finding the local entity structure (individually) from different labels. The next step was to create each entity structure, which was done by a token model generation that will help eliminate superfluous tokens (generated from the labelling step). Finally, for the concatenation of two consecutive tokens, an incremental algorithm will be used. Extraction modules consist of the method, i.e., tokenisation, token filtering and non-contiguous graph building. Correction modules consist of the method of relevant token clustering and sweeping. In the tokenisation method, the need for white spaces and their distance in a set of tokens is not explained. In the case of token filtering, they have assumed to put the tokens in a bounding box, but while using the filtering algorithm, they have only mentioned the coordinate of the upper left corner, height, and width. Another problem where the coordinates were not taken into consideration, so the result of the algorithm may not be accurate. Entities are labelled by using regex. Structure correction evaluation was performed by a set of regex patterns that are used in conjunction with the relations between labels. This step was taken place for the elimination of superfluous

tokens. By using the method along with module correction, the experimental result was 97.50% accuracy in precision. This test does not include account extraction of tables items. It also does not discuss the issue with OCR engine incorrect character read. Although a search of entities can be done using regex, the challenges were more when the text was incorrect. Furthermore, this would also be a challenge when the invoice was not structured in a localized fashion. Even multi-page document extraction might fail if the structure were divided between pages. The authors have not discussed any shortcomings or the scope of future work, but this review identifies the potential failure of multi-page document extraction as a gap in their research.

Later, (Rahal et al., 2016b) designed a method to extract valuable entities of tokens from semi-structured documents (administrative). They tried to portray this model as a mislabelling correction. Initially, the entities were labelled. Then every entity was modelled by considering the nodes, which represent tokens and arc, which represent the distances. A clustering algorithm was applied for the incrementation and concatenation of relevant tokens by ignoring noisy ones. Token filtering and the incremental algorithm was used for processing cluster analysis. Their proposed model generated the results when the invoices with fields were tagged. Once the tagging process was done, the tokenisation algorithm is used to label the elements. Due to the iterative nature of the algorithm, it removes the noisy labels in RW tokens. After that, a graph is built by considering candidate tokens of the entity. In that graph, nodes are represented by KF tokens, and the distance between them was represented by arcs. In the end, the incremental algorithm was applied for the concatenation of two consecutive tokens. These consecutive tokens were fixed empirically by a certain threshold. All the above method was used to handle node and arc for irradiating redundant content. The experimental results also show an improvement in both performance and accuracy. As they have not used any model matching operations or algorithms, it still proved robustness and portability. They got 97% of accuracy, which was acceptable, and a novel model was also used that gives an impact of faster use.

Furthermore, (Gupta et al., 2016) used the concept of rule-based information in business invoices. A descriptive rule was proposed, which helps in grouping related information, and thus extraction can be more accessible. According to them, invoices were interchanged between business organisations on a day-to-day basis, and they all contain a similar kind of information, i.e., What is the name of issuing company? To whom is it issued? What is the amount of the invoice? What is the mode of payment? Capturing and structuring this information can help in the critical supply chain and cash flow by assisting market analysts in making better decisions. They recommend annotation as a structure for specific rules to add more structure to business invoices to simplify and standardise the storage and retrieval of business information as a solution. The rule is still to be adopted, although, in the future, a query-driven approach has been proposed that maps tags to fields in a relational database model, allowing unstructured data to be converted to structured data. Although this paper discusses many papers which were meant for actual invoice data extraction, this paper does not discuss any related solution. It rather proposed rules which would be helpful in the better mapping of already extracted data. Most of the paper was based on theory about Big Data and its three forms, that is structured, unstructured and semi-structured.

(Miloudi et al., 2016) reviewed the fundamentals of working capital and techniques of the supply chain. They developed a mathematical model based on the collected amount for invoices and total money collected from downstream customers at a particular period. This model also evaluates its performance on fictitious instances. They aimed to determine planning models for payment to generate cash and minimizing cost before considering borrowings (short-term). The two main methods for funding and optimisation of working capital (supply chain finance and collaborative finance) were adopted. Over the course of ten periods, 20 invoices were expected to be obtained and 15 supplier invoices to be paid. Their model does not account for overdue payments from customers or scheduled payments from suppliers. As a result, the next step was to formulate a model that would take into account the two scenarios described above.

This paper (Koo, 2016) used the text-line detection algorithm for camera-captured document images. They implemented Maximally Stable Extremal Region (MSER) algorithm to extract text components. Machine learning based on the Markov Random Field (MRF) model was used for the text line classification. For the experimentation, natural scene images were used. They designed their dataset for evaluation. The F1 score of the proposed system was 0.9329.

In (Liu et al., 2016a), a single-shot multibox detector was used for object detection. SSD gives better results than the single-stage method. The datasets were used for evaluation as PASCAL VOC, COCO, and ILSVRC. The future scope is to detect and track the objects from video. (Jenckel et al., 2016) implemented a method for sequence-based learning for text extraction. The method was based on segmentation and utilized clustering on individual characters, i.e., LSTM architecture. They achieved a character error rate of only 7.33%. (Ma & Hovy, 2016), designed the neural network architecture for End-to-End Sequence Labelling to word-character automatically with the help of Bi-directional LSTM-CNNs-Conditional Random Fields (CRF). They implemented named entity recognition (NER) for two datasets, part-of-speech (POS) tagging and CoNLL 2003. The model's accuracy for the POS dataset was 97.55%, and the F1 score for CoNLL was 91.21%. In future, they wanted to explore a model for multi-task learning. By replacing single RPN with multi-RPN (Nagaoka et al., 2017), better text detection compared to the original Faster R-CNN was performed. This method helped detect different text size-based images with better accuracy as each RPN was trained differently with many ROIs. Now, because the invoices have text with different font sizes, it will be interesting to apply multi-RPN and evaluate the result. Later in (Shi et al., 2017), a text detection method was proposed. They used the pretrained VGG-16 model to extract the deeper features. For the segmentation, the box-oriented model was used. For the experimentation 3 publicly available datasets were used such as ICDAR 2015, MSRA-TD500 and ICDAR 2013. The F1 score for ICDAR 2015 Incidental Text was 75.0. In future they wanted to extend SegLink for end-to-end recognition system.

Furthermore, (Raoui-Outach et al., 2017) worked on receipt image scanning using DCNNs. The method was tested on 5000 images (3000 receipts and 2000 non-receipts). The method performed well in classifying images into a receipt or not a receipt. The method was based on the fusion of text and logo identification. Later, (Berg Palm et al., 2017) proposed an invoice analysis system called ClouScan that was based on zero configuration or upfront annotation. It was not based on template matching; instead, it learns from a single global model to work on unseen invoices. The system was evaluated by extracting 8 fields from a dataset of 3,26,471 invoices. The RNN and baseline model achieved average F1 scores of 0.891 and 0.887, respectively. The system was not validated against publicly available software or dataset, so it was hard to validate the accuracy of this system. Furthermore, as seen in much other research, extraction of few fields may yield higher average F1 scores, but it completely kills the purpose of using a suitable extraction system that can extract all relevant fields instead of focusing on few selected fields.

In their work, (Teunissen, 2017) used the CSP method for field extraction for Dutch invoices. During the implementation of CSP, the variables and constraints are checked first. The next task was to find the relation between the variables and the ground truth. For the experiment analysis, 241 Dutch purchase order invoices in image format were used. The overall system accuracy was 85.1%, and system accuracy without the OCR error was 89.1%. The future scope was to minimize the system error. Additionally, they want to extent research for graphical areas as well. Furthermore, (Gilani et al., 2017) have developed table detection using the deep learning method for the layout analysis in case of documents having tables. The deep learning-based; RPN followed by an F-CNN network was developed for table detection. The UNLV dataset was used for the experimentation, which consists of a research paper, documents, magazines, etc. This technique gives better results than Tesseracts. One gap in the research is its inability to extract table structure and table contents, which the authors hope to address in future work.

(Rahal et al., 2018) worked on data extraction of Arabic and Latin scanned invoices. The result was evaluated on 1050 real invoices. The method first labelled the images, and later by combining the logical and physical structure, a local graph model was built for entity extraction. Based on 7 entity that was extracted they achieved an average accuracy of 92%. Major failure in the model was due to the OCR issue in Arabic invoices and thus required more future work. Furthermore, (Jiang et al., 2018) proposed FCN-biLSTMs, which automatically processed and recognised the invoices data. With the help of invoice layout information and text characteristics, the line text was extracted, which gave higher accuracy. The approach was further extended by (Lohani et al., 2018), where they proposed a mode-free invoice reading system using GCNN. The framework divides invoice data into different entities and then uses a graph structure to learn structural and semantic knowledge. With an overall F-measure score of 0.93, the machine could read up to 27 entities without any template matching or configuration.

(Reza et al., 2018), proposed advanced layout analysis for invoices for removal of table cell lines and merging text lines. They have integrated the new solution in the existing anyOCR solution to validate the accuracy of the solution. In the process, before applying binarization, all line graphics were removed, then binarization is applied. After that, THE existing text line extraction method was changed and merged with text line segments in the same row. Finally, text was recognized for each line. Based on the sample size of 29 images, the solution outperformed anyOCR and ABBYY. They achieved recognition accuracy of 83.34%. Later, (Wang et al., 2018) proposed an OCR technique for Value-added tax invoices. The network consists of deep CNN followed by the residual network (ResNet). They achieved a testing accuracy of 99.38%. Furthermore, research was carried out by (Sidhwa et al., 2018), where they proposed text extraction from invoices and bills using OpenCV and Tesseract OCR. The paper only discussed end to end solutions to scan and extract tests. It does not share any details of the accuracy and performance of the solution.

(Katti et al., 2018), have discussed the Chargrid technique, which was used for 2D document analysis. The proposed network consists of the fully convolutional encoder-decoder network, which predicts a segmentation mask and bounding boxes. This model was useful to extract the data, classify and recognize the named entity. The Chargrid model extract entities such as “Invoice Number, Invoice date, Invoice amount, Vendor Name, Vendor Address, Line-item Description, Line-item quantity, Line-item amount” (Katti et al., 2018). The proposed model got maximum accuracy for the invoice date as 84.28%. (Ha et al., 2018) proposed a model called OCRMiner, for extracting indexing metadata for a structured document. Their system was based on text analysis and layout features analysis for the identification of text blocks in the invoice document. The system was tested using open OCR software, and it achieved an accuracy of 80.1%.

(Zhang, 2018) focussed on online invoicing system design for QR code recognition and enterprise information cloud storage. The system only requires the taxpayer number or the QR code instead of the invoice and reduces the waiting time to check the invoice. In this paper, (Gui, 2019) used the Hough transform, which was applied to the scanned invoice. For the character verification and recognition, template matching and geometric feature extraction methods were used. The recognition accuracy rate was 95%, and the invoice identification rate was 99%.

Furthermore, (Chung et al., 2019) proposed the pattern recognition method designed for lottery and invoice number and character recognition. This system was used for Arabic numerals. The region of interest was calculated for the captured images, and the algorithm gives outstanding accuracy. To compare the results of the proposed system, OCR was applied. Later (Yi et al., 2019) proposed Gaussian Blur and Smoothing–Convolutional Neural Network Combined with a Recurrent Neural Network (GBS-CR) method to save money and stabilize work efficiency for the medical invoice recognition. For the training Alexnet–Adam–CNN (AA-CNN) model was used. The average increase in recognition rate was 10 to 15 percentage points. The future scope was to improve the pre-processing methods for restricted Boltzmann machines (RBMs) and applying the approach to more

areas like 3D object recognition and MRI images. Hospitals were required to commit a significant amount of staffing to manually entering medical invoice data into the medical system (approximately 300,000,000 medical invoices each year). The breakpoint font in medical invoices can be fixed using a novel pre-processing approach called Gaussian blur and smoothing (GBS to improve the identification rate of the breakpoint font in medical invoices, a CNN and RNN combinational model was used. The semantic revision module was implemented using RNN. The designed model achieved a text recognition accuracy of 83.46%.

(Riba et al., 2019), proposed the system consisting of Graph Neural Networks (GNNs) used to describe the orientation of the tables in the invoice document. The network consists of CNN, Residual block, adjacency matrices to learn the weights. For experimentation, CON-ANONYM with 960 documents and RVL-CDIP being 4,00,000 grayscale images used. They have got a table detection Accuracy of CON-ANONYM dataset was 97.2% and 83.9% for RVL–CDIP dataset. The future scope was to use the practical architecture in all the structures of the table. Later, (Holeček et al., 2019) designed a structured table-level data extraction from invoices and bill using a graph over the word for position identification, and then training was performed using a novel neural network. Based on the sample size of 3554 pdf invoices, they achieved the accuracy of 93% of line-item detection. Although table-level data extraction was challenging, the biggest issue was with unstructured format and text position, which this paper lacks to provide any solution. Therefore, in real life scenario, where format of the invoice line-item table was not known, it will be difficult to adapt only this model.

Furthermore, (Liu et al., 2019b) proposed the graphical convolution network for Visually Rich Documents (VRDs). The VRDs are encoded with the help of graphical convolution. The Graph embeddings were used to segment the text and extract the entity. With the help of the BiLSTM-CRF model, the results were extracted. The limitations or the future scope of the NER technique was to achieve multitask learning. For the experimentation, two datasets were used, such as VATI and IPR. The F1 score for VATI was 0.881, and the

IPR score was 0.849. In the future, they hope to work on document classification tasks. (Bhatt et al., 2019) have taken the next step in invoice data extraction to overcome the challenges in duplicate invoice payment. First, the images were converted in the structured template and then data was extracted as key-value pair. Based on certain parameters and value matching, the invoices were marked as duplicates. Based on 80,000 invoices based on production data samples, and with 8 fields that were identified to compare the value, an average F1 score was around 90% was achieved.

In this paper (Meng et al., 2019a) designed a smartphone-based reimbursement system for invoices. They used a Hough transform for image tilt correction and the YOLOv3 model for target positioning and extraction. Finally, OCR was applied. Experimental results gave localisation accuracy of up to the 92.5% and identification accuracy of up to 97.5%. Later, (Tarawneh et al., 2019) used the deep CNN method for invoice classification. The features were classified using various machine learning algorithms, namely including Random Forests, KNN and Naive Bayes. The best result achieved was 98.4% using KNN. The test was performed on 45,000 invoice images. The future work plan was related to image enhancement for better classification. (Tang et al., 2019) proposed the deep fusion analysis method based on K-means and Skip-gram for the electronic invoice. For evaluation, 39,668 samples were used. The model's accuracy is 76% in the inter-enterprise association analysis and 85% in the inter-user association analysis. Furthermore, (Tang et al., 2020) performed research on a machine learning-based anomaly detection algorithm that detects irregularities found in e-invoices. The depth fusion analysis method was using the k-means and Skip-gram to check the abnormal behaviour. This system was designed for security purposes.

Later, in this paper, (Blanchard et al., 2019) implemented the model, which will be able to generate the invoices automatically. There are two types of generation: invoice was generated by random variation in the actual invoice, and the second one was that just invoice elements by adding inter and intra-element relations. The output was in standard format as XML-GEDI form. For the recognition graph convolutional neural network was

used. The dataset contains more than 3000 images. As stated in this paper (Reza et al., 2019), the advanced machine learning model proposed that used Table Localisation and Segmentation using GAN and CNN for invoice data. The conditional Generative Adversarial Networks (cGAN) was designed for table area localisation and segmentation using SegNet. In the generator network, they used the pix2pixHD architectures for table localisation. They have named the generator network a Coarse-to-fine generator, a generator pair that is a global network and a local network. The Multi-scale discriminators was used. The global generator model was trained to achieve a good output. The SegNet architectures were used for table segmentation. The SegNet architecture was encoder-decoder, uses the VGG style. For the experimentation ICDAR 2013 table competition dataset, which contains 238 document images were used. The precision of the result from the proposed table localisation model was 98.29%. The authors have identified the need to develop architecture to allow the table to detect original-size images and investigate column overlap for better segmentation as a gap in their research.

(Patel & Bhatt, 2020), proposed the Abstractive Information Extraction from Scanned Invoices (AIESI) using the End-to-end Sequential Approach. The information of various fields was extracted, such as payee name, total amount, address, and date. This method was helpful in the case of database exploring, document searching. The Key Invoice Parameter Extraction (KIPE) was used to extract the pattern of text. The BiLSTM acted as a multiclass classifier. For the evaluation, ICDAR, 2019 and SROIE, which has 1,000 scanned receipts, 876 were used for training and 347 for testing. The performance evaluation of the proposed model in the case of the SROIE dataset was 88.2%. (Loginov et al., 2020), proposed a codebook that was generated from visual word documents using the Bag-of-Words method for text recognition. The dataset consists of business documents like receipts, identity cards, invoices. They created a dataset and achieved an accuracy of 91.8%.

Later, (Zhang et al., 2020a) designed an end-to-end text reading and information extraction network to read the text and extract the information. The three models that

make up their architecture are the text reading module, multimodal context block, and knowledge extraction module. The text reading model was used to locate and recognize the text in an image. The model uses the Feature Pyramid Network (FPN) (Lin et al., 2017), which consists of the concept of the sliding window and for feature extraction RPN (Ren et al., 2015) was applied. Furthermore, the region-based detector Fast R-CNN (Girshick, 2015) and residual network that was ResNet50 was used to extract the shared convolution features. The multimodal context block acted as a bridge for text reading and information extraction. In multimodal context block, instead of working on a single feature that is textual or positional, they worked on multi-features. This block gave the relation between the visual context and textural context. Next was the information extraction module, where the context and textual features were required to extract the entity. The BiLSTM model was used to extract those entities. For the experimentation, they used three datasets such as taxi invoices, Resumes, SROIE. For the taxi invoices dataset, they extracted the 9 entities with the F1 score value of 93.26%. For the Resume dataset, they have extracted 6 entities; for that, the F1 score value was 76.3%. For the SROIE dataset, they have extracted the 4 entities, and for that, the F1 score value was 82.06%. The overall TRIE model accuracy was 93.26%. For comparison purposes, the pipeline of SOTA text reading and information extraction was used. Due to the lack of existing systems work that mainly worked on text detection and recognition, they have reimplemented the Chargrid, NER, GCN model for a fair comparison. The primary limitation of the model is that it fails when extracting information from multi-modality features; that is, the text reading module gave text features only at the time of entity extraction; due to this, the layout information was loosed. This is a gap that needs to be addressed.

(Li et al., 2019) implemented a GAN-based feature generator for table detection. The proposed novel network was used to generate layout features for table text to improve the performance of less-ruled table recognition. This feature generator model was similar to the GAN, and it was considered to extract similar features for both ruling tables and less-

ruled tables. They used the network structure of VGG in the generator model, and the activation function rectified linear unit was used. They have named their model “FGAN”, where they merged the UNET architecture and feature generator for semantic segmentation. They have trained the model on ICDAR 2017 dataset. They have divided the dataset as 549 images with 699 tables and 243 images with 317 tables into training and test data, respectively. They have got a 95.9% accuracy. No future scope or limitation was discussed in this paper.

(Le Vine et al., 2019) have also proposed extracting tables from documents using conditional GAN and genetic algorithms. They used a top-down approach, first using a GAN to map a table image into a standardised ‘skeleton' table shape denoting the approximate row and column borders without table material, and then using a distance measure optimised by a genetic algorithm to match renderings of candidate latent table structures to the skeleton structure. The generator architecture was designed with the help of the UNET model, and a convolutional PatchGAN architecture was used for the discriminator network. With the help of a genetic algorithm, the optimization of the model was done. This approach was to be paired with an algorithm to detect the table area in documents surrounded by text and an OCR algorithm to extract text into the data structure in the future. They have not discussed the experiment analysis as well as the accuracy of the model was not mentioned in this paper.

Furthermore, (Anand & Khan, 2020) proposed the W-A net model for text detection. The architecture has a W shape, the middle branch used Atrous convolution. The Atrous convolution was nothing but a dilation operation. For the experimentation, 3 datasets were used, such as ICDAR 2015 dataset, ICDAR 2019 SROIE dataset, CTW1500 dataset. The F1 score for the SROIE dataset was 89.65%. In future, they wanted to work on memory management and time reduction. (Kim et al., 2020), designed the TLGAN to localise the text of the document. The generator network finds the image features using an ImageNet pre-trained VGG19 network. For the experiment evaluation, they used the Scanned Receipts OCR and Information Extraction (SROIE) and TLGAN achieved 99.83%

results. The gaps exposed by this model's limitations were that it was unable to detect the specific font and the resolution due to the residual network or image features were based on the kernel size. More training computations and memory were required due to the VGG16 network.

(Rastogi et al., 2020) proposed using Formal Concept Analysis (FCA)-based template detection and knowledge graph rule induction to extract information from document images. In the first section of this paper, they have discussed the challenges related to information extraction. A knowledge graph was used to capture document structure, and a relational rule learning framework was used to generate extraction rules on the knowledge graph. A knowledge graph with a predefined schema that captures the spatial and semantic relationships between the detected elements was proposed to extract the information. They used a publicly accessible shared dataset of 1400 scanned bank trade finance documents to test this approach. The dataset contains 200 documents from seven different models, including CMB, DBS, Axis, Citi, DutchBangla, Hangseng, and Shanghai. They have extracted a total of 9 entities such as “Acc number”, “Amount”, “Date”, “Phone”, “Ref”, “Swift”, “Tenor”, “Drawee”, “Drawer”. They have got overall accuracy of 93.5%. The future scope was to explore the system which was able to extract entities from various documents.

According to (Kumar & Revathy, 2021), an automated invoice system was designed with the help of OCR to reduce the cost and staffing. It was helpful in real-time applications to extract information such as alphanumeric recognition from invoices. The objective of this system was to extract information such as invoice number, date, and payment details. Later, (Tutica et al., 2021) proposed the machine learning-based LGBM and random forest model for analysis of business payments. After the evaluation, it was observed that the LGBM gave better results in a business invoice payment document. (Cinti et al., 2020), proposed the Online Approximate String Matching (OASM) for FPGA implementation. The priority rules were defined with the help of a search algorithm. To overcome the drawback of ASM, which produces shadow hits due to

the tolerance threshold, the OASM was implemented. They have made future scope in case of FPGA hardware implementation.

Furthermore, (Krieger et al.) designed a graph neural network approach for datasets with high layout variety for the invoice documents. They have introduced a graph-based approach to information extraction from invoices and apply it to a dataset of invoices from multiple vendors. The figure below represents the proposed method architecture. The model takes document graphs as input, in which each node represents a word in the document. Syntactic, positional, and semantic features were attached to each node, derived from the word the node represents. The edges in the document graphs represent the relative positional relationship between the words. Key items were then extracted via node classification. They performed the character level encoding with the help of the Gated Recurrent Unit (GRU) model.

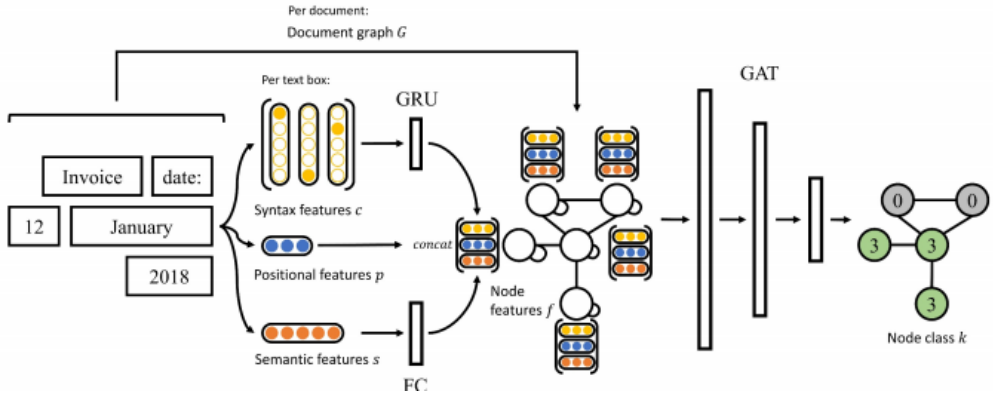


Figure 15: The node features are embedded using fully connected (FC) and recurrent (GRU) layers and are attached to the document graph, which is passed into graph attention layers (GAT) for node classification. (Krieger et al.)

In this paper, the authors have mentioned that there were no publicly available sets of labelled invoices that were of sufficient size and variety to train the machine learning and deep learning models. They have collected a set of invoices by an audit firm, which was composed of 1129 English one-page invoices from 277 different vendors. They have divided the dataset into three parts, 903 invoices were used for training, 113 for validation and 113 for the testing. They have extracted the information for the following fields as

“invoice number”, “total amount”, and “invoice date”. They were able to achieve 90% overall accuracy. In the future, the authors hope to address some of the gaps identified in their research, namely, including more key items to be extracted. Line items especially represent an interesting area of investigation. They want to explore this model to extract key items from other document types such as receipts and purchase orders.

(Meng et al., 2019b) proposed the smartphone aided intelligent reimbursement system using deep learning for invoice images. Firstly, the distorted images were pre-processed before being subjected to the Hough Transform Accumulator (HTA) algorithm, which adds an accumulator based on the Hough transform to obtain tilt correction and image recovery for the distorted image. The You Only Look Once-version 3 (YOLOv3) algorithm was used to accurately locate, segment, and intercept the main information areas on the invoice picture in the natural scene to eliminate unnecessary information. The imported text information block areas in invoice images were detected with the connectionist text proposal network (CTPN), and the detected text was identified with densely linked convolutional networks (DenseNets). For the Densenets network, the Connectionist Temporal Classification (CTC) algorithm was used to align the input and output formats of the text, and accurate OCR was performed on the intercepted block area invoice image. Finally, a new algorithm, Regular Matching and Recursive Segmentation (RMRS), was developed, which achieved standard formatted output on misaligned or offset information using recursive segmentation of regular matching.

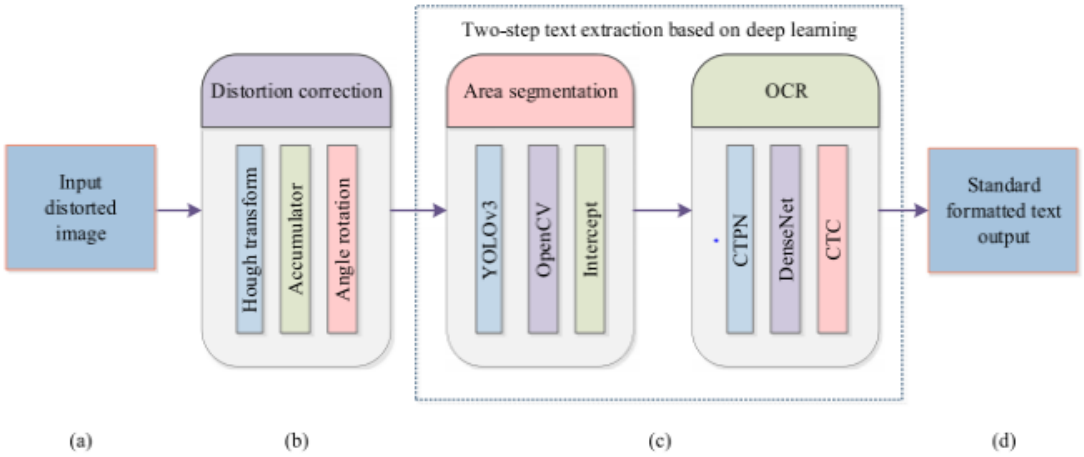


Figure 16: The system architecture (Meng et al., 2019b)

The dataset had 5,990 Chinese characters, English letters, numbers, and punctuation marks. The proposed system had an average recognition accuracy of a maximum of 99% and a minimum of 96%. They did not mention any limitations with their proposed system.

Table 8: Gaps Identified: Invoice Document.

"Devised by the author".

Research Gaps	**References**
Fails in the handwritten text recognition	(Agarwal et al., 2007)
Not discussed text extraction and extraction of table items	(Gao et al., 2015)
Multi-page page document extraction might fail if the structure were divided between pages.	(Rahal et al., 2016a)
Unable to extract the contents from the table as well as the structure of the table.	(Gilani et al., 2017)
It will be challenging to adopt only this model when the invoice line-item table format was not known.	(Holeček et al., 2019)
Unable to detect the original size of table and investigate column overlap for better segmentation.	(Reza et al., 2019)
It fails in the case of information extraction in multi-modality features; the text reading module gave text features only at the time of entity extraction; due to this, the layout information was loosed.	(Zhang et al., 2020a)

Unable to detect the specific font and the resolution due to the residual network or image features were based on the kernel size.	(Anand & Khan, 2020)
To include more key items to be extracted, especially line items represent an interesting area of investigation. As well as to explore the model for extracting key items from other document types such as receipts, purchase orders, etc.	(Krieger et al., 2021)
Only able to identify the table does not extract text	(Kim et al., 2020)

2.3.6. Summary Table

This section outlines the summary of all the literature review done in the prevision section. This is a small tabular representation for our readers to reference the research done in context of text extraction quickly.

Table 9: Chapter 2.3 Literature review summary.

"Devised by the author".

Area	Method	Maximum Accuracy	Gaps identified	References
Written Languages and Scripts	Curvature Scale Space Features BILSTM-CTC Morphological approach Hidden Markov Model contextual-based method word-based LM Nabocr CNN Seq2seq model Attention-based LSTM model Segmentation algorithm Deep learning model combinational model Hidden Markov Model Line segmentation approach CNN-RNN RNN Unified architecture	100%	• Design a model for Parts-of-Arabic-Words (PAW). • To improve classification model and neural network. • There was no comparison made due to the lack of existing dataset and research on Romanized Sanskrit text. • The proper functional analysis with the help of a precision language model. • Future scope on complex image text recognition. • The future scope is to detect real-world errors from the OCR output. • In future, do the work on Marathi and Urdu language and increase word recognition rates too. • The work will be extended to minimize the system error and work in the graphical area as well.	(Khoddami & Behrad, 2010) (Paul & Chaudhuri, 2019) (Singh et al., 2014) (Prasad et al., 2008) (Sari & Sellami, 2002) (Magdy & Darwish, 2006) (Kissos & Dershowitz, 2017) (Sabbour & Shafait, 2013) (Sagar et al., 2008) (Avadesh & Goyal, 2018) (Krishna et al., 2018) (Dwivedi et al., 2020) (Yin et al., 2019) (Zhuang & Zhu, 2005) (Javed et al., 2010) (Javed & Hussain, 2009) (Jain et al., 2017) (Vinitha & Jawahar, 2016) (Krishnan et al., 2014)

	CSP method Long Short-Term Memory Bigram, Trigram Weighted Euclidean distance Backpropagation network PhotoOCR Segmentation algorithm OCR-RCNN Geometrical rectification framework RNN FCN Wavelet Transform novel alignment method comic frame Quadratic discriminate Classifier ANN post-processing techniques		• The model performance should be improved. • To do the work on grammar. • The model will be extended for handwritten Amharic document image recognition. • Future scope on local energy variation analysis for text detection. • To improve character recognition and automatic conversion of Japanese comics into international comics.	(Saluja et al., 2017) (Wickramarathna & Ranathunga, 2019) (Alshehri, 2021) (Yeremia et al., 2013) (Bissacco et al., 2013) (Zhu et al., 2018) (Dave et al., 2020) (Belay et al., 2020) (Chakraborty & Mallik, 2013) (Zhou et al., 2017a) (Mao et al., 2002) (Chiron et al., 2017) (Arai & Tolle, 2011) (Prameela et al., 2017) (Singh & Kaur, 2010) (Lehal & Singh, 2002) (Rizvi et al., 2019)
Real Scene Images	Faster RCNN Pyramid attention network Saliency-based CTPN MSERs CNN Semi-supervised neural network GRCNN	94.7%	• Valid only for scene text recognition. • Applicable for short text. • The system should give better results in the case of the real word. • This is used for object detection. • They will locate the location on the input image. • The real-time object detection is not possible.	(Wang et al., 2017) (Hua et al., 2002) (Huang et al., 2019b) (Lu et al., 2019) (Beaufort & Mancas-Thillou, 2007) (Zhong et al., 2019b) (Karaoglu et al., 2012)

	AON Data augmentation techniques Generic object detection BiLSTM, RNN LocSLPR AF-RPN Survey paper Hough transform Content-based indexing Segmentation algorithm Gabor filter FFT and neural network FASText Keypoint Detector FCN PSENet OpenCV and CNN Neural Tensor Networks Faster R-CNN Fast R-CNN		• The scope is mentioned as text detection, extraction, segmentation, and recognition from natural scene images to work on various languages. • The future scope is to detect curved text, text recognition and general object detection. • The algorithm cannot detect text that is not horizontally aligned as it cannot produce rotated bounding boxes. • It cannot detect text that is embedded circularly.	(Tian et al., 2016) (Islam et al., 2016) (Lee & Osindero, 2016) (Bartz et al., 2017) (Wang & Hu, 2017) (Cheng et al., 2018) (Pervin et al., 2017) (Namysl & Konya, 2019) (Borisyuk et al., 2018) (Wang et al., 2011) (Brzeski et al., 2019) (Zhu et al., 2019) (Zhong et al., 2019a) (Kanagarathinam & Sekar, 2019) (Karanje & Dagade, 2014) (Sin et al., 2002) (Kumuda & Basavaraj, 2017) (Chang et al., 1995) (Smith & Kanade, 1995) (Chun et al., 1999) (Busta et al., 2015) (Zhou et al., 2017a) (Li et al., 2018) (Goel et al., 2019) (Geetha et al., 2020) (Ren et al., 2016)

				(Girshick, 2015)
Text from Video	coarse-to-fine algorithm Recognition and detection algorithm SSD k-means clustering SVM based method Region-based, filter	90.8%	• Only the text was detected, but there is a lack to extract the text from the complex background and videos. • Research the case of moving text. • The future scope is to detect and track the objects from video. • The classification should be improved.	(Ye et al., 2005) (Lyu et al., 2005) (Liu et al., 2016a) (Gllavata et al., 2004) (Qi et al., 2000) (Chen & Bourlard, 2001) (Yang et al., 2011a) (Kastelan et al., 2012) (Yang et al., 2011b)
Non-Invoice Document	Information Retrieval (IR) Symbian C++ Error detection and correction model Regression approach Isolated word-based Language model Novel error detector Fully connected neural RNN EAST algorithm Hough transform algorithm TableNet approach Key Information Extraction Prototypical analysis system CNN Global threshold binarization technique	100%	• To improve the effectiveness of retrieval. • To work in case complex background. • Increase the features to achieve optimal performance. • To work on a large-scale dataset. • Future scope on table structure and extraction of contents. • The bug reports should be improved for business documents. • To design an android application to capture an image • The research has been done only for specific cases by using the CNN model. • The research is only limited to historical documents.	(Fataicha et al., 2006) (Laine & Nevalainen, 2006) (Zhang et al., 2009) (Mei et al., 2018) (Coustaty et al., 2018) (Purwantoro et al., 2019) (Poncelas et al., 2020) (Jatowt et al., 2019) (Anand et al., 2020) (Ast, 2020) (Geetha et al., 2020) (Pham et al., 2020) (Paliwal et al., 2019) (Yu et al., 2020) (Baumann et al., 1997) (Arroyo et al., 2019)

	Co-HOG ANN CRNN RPN Faster RCNN MANICURE INFTY Gabor transform, LDA RNN Maximum entropy Markov model End-end framework OCRXNet EATEN region-based model Fuzzy Neural Hybrid system Entity Recognition SVM Intellix edge detection algorithm Hough transform A probabilistic approach PATO open-source Tesseract Attention model Edge-based feature Unsupervised local thresholding method Text-based		• A further plan is to add a supervised machine learning approach for better extraction of data. • The future scope of the testing should be possible for other products as well. • To design a model for invoice data extraction. • Future work improves the classifier for the mixed and non-mixed conjunct consonants to achieve a better recognition rate. • To work on the real-time handwritten text. • To design a model for complex documents that documents are having graphs, tables, pictures. The results were promising apart from the issue in reading some characters which were misread. • Future scope on foreground graphics and missing single character due to the hole. • The results were promising apart from the issue in reading some characters which were misread. • The method does not fit well with digits, special symbols, and punctuation and has an issue with low-quality images. • To design such a system that will apply to non-English scripts also. • To work on detection of text in textured areas.	(Lund et al., 2013) (Tian et al., 2013) (Mithe et al., 2013) (Shi et al., 2016) (Zhang et al., 2017a) (Weng & Xia, 2019) (Jun et al., 2019) (Jacobs et al., 2005) (Liu et al., 2020) (Nartker et al., 2003) (Suzuki et al., 2003) (Zhang et al., 2002) (Martınek et al., 2020) (Kluzner et al., 2009) (Packer et al., 2010) (Pegu et al., 2021) (Arora et al., 2020) (Guo et al., 2019) (Pitou & Diatta, 2016) (Henge & Rama, 2016) (Kooli & Belaid, 2017) (Agarwal et al., 2007) (Esser et al., 2013) (Shivakumara et al., 2005) (Bartoli et al., 2014) (Rigaud et al., 2017) (Dong & Smith, 2018)

	Edge Based Segmentation Approach segmentation method based on morphological operator, SVM classifier CNN CAE SVM Deep recurrent attention model Faster RCNN Unsupervised classification RCNN SSD SVM text recognition technique generic indexing system Faster R-CNN Segmentation algorithm MLP Language Model ORAN method DNN-HMM Multiple-modal information retrieval DCT SWT holistic based approach BLSTM-CTC architecture Gabor filter Bi-directional LSTM model		• To improve the performance by adding a number of features. • The limitation of this approach when characters are remarkably close unable to recognize. • In future, to extract the handwritten character. • The future scope is to work on field value identification. • Extend the work, which will be helpful for the robotics application. • Work on the loss function to achieve better results. • To explore incorporating external knowledge (for example, work-related information available in word embeddings) into the learning framework for event extraction.	(Grover et al., 2009) (Nagabhushan, 2010) (Cristani & Tomazzoli, 2014) (Fabrizio et al., 2009) (Yindumathi et al., 2020) (Grönlund & Johansson, 2019) (Wen et al., 2011) (Shaker & ElHelw, 2017) (Tian et al., 2021) (Ho & Nagy, 2000), (Girshick, 2015) (Liu et al., 2016a) (Uijlings et al., 2013) (Chiang & Knoblock, 2011) (Leon et al., 2013) (Borisyuk et al., 2018) (Shinde & Chougule, 2012) (Rashid et al., 2012) (Jin et al., 2002) (Zidouri, 2004) (Arora et al., 2019) (Nashwan et al., 2018) (Breuel et al., 2013) (Agarwal et al., 2019), (Yang et al., 2019) (Griffin & Kurup, 2017) (de Jager & Nel, 2019)

	Convolutional neural network Segmentation Tree-Based Data Fusion Techniques ASM a binary gravitational search algorithm RNN EESRGAN MTGAN GAN-KD DetectorGAN GAN-DO TLE AEM			(Shehzad et al., 2020) (Pourghahestani & Rashedi, 2015) (Zhang et al., 2018) (Rabbi et al., 2020) (Zhang et al., 2020b) (Wang et al., 2020) (Liu et al., 2019a) (Charan & Lina, 2019) (Kundu et al., 2020) (Wang et al., 2019)
Invoices Document	Graph model DCNN T-Recs++ prototype Morphological approach Classification algorithm probabilistic framework thresholding techniques Feature extraction model E-Invoice Framework Bag of words concept Recognition model FCN-biLSTMs Hough transform	98.42%	• Major failure in the model was due to the OCR issue in Arabic invoices and thus required more future work. • To implement the android application for real-time use. • The challenges faced in real-time such as quality, handwritten remarks, stamps, a small number of entries in the table. • The actual solution to the problem is not given; that is the information extraction from the invoice. • For future work, different optimisation methods have been planned to increase performance.	(Rahal et al., 2018) (Raoui-Outach et al., 2017) (Kieninger & Dengel, 2001) (Belaïd & Belaïd, 2004) (Alippi et al., 2005) (Bart & Sarkar, 2010) (Chien & Lin, 2009) (Shin, 2012) (Chu et al., 2014) (Liu et al., 2016b) (Jiang et al., 2018) (Gui, 2019) (Tang et al., 2019)

	Deep fusion analysis CNN Alphanumeric Recognition End-to-end Sequential Approach Bag-of-Words method LGBM Anomaly detection algorithm Pattern recognition method Genetic algorithm RNN OCRMiner Segmentation algorithm AA-CNN Graph Neural Network Residual network Tesseract OCR Novel neural network GCNN Template matching phone-based reimbursement system CBR-DIA Faster RCNN textual and graphical processing NLP Based Framework FRESCO Knowledge-based approach morphological operations		• They were used for specific area extraction, not all information from the invoice. • In future scope large scale dataset to improve the performance. • This model fails in delayed customer payments and anticipated supplier payments cases. • The future scope is extended to multiple echelons involving several currencies. • To extend SegLink for an end-to-end recognition system. • To work on memory management and time reduction. • Future work on complex background images. • To work on the document classification task. • The future scope will explore the model for multi-task learning. • The aim of the system is to explore the task of classifying handwritten text and invoices into an excel format. • The limitations of this model are unable to detect the specific font and the resolution due to the residual network or image features are based on the kernel size. • The training computations and memory required is more due to the VGG16 network.	(Blanchard et al., 2019) (Kumar & Revathy, 2021) (Patel & Bhatt, 2020) (Loginov et al., 2020) (Tutica et al., 2021) (Tang et al., 2020) (Chung et al., 2019) (Köppen et al., 1998) (Berg Palm et al., 2017) (Ha et al., 2018) (Reza et al., 2018) (Yi et al., 2019) (Riba et al., 2019) (Wang et al., 2018) (Sidhwa et al., 2018) (Holeček et al., 2019) (Lohani et al., 2018) (Bhatt et al., 2019) (Meng et al., 2019a) (Tarawneh et al., 2019) (Hamza et al., 2007) (Nagaoka et al., 2017) (Ren et al., 2015) (Kosiba & Kasturi, 1996) (Roychoudhury et al., 2016) (Bayer & Mogg-Schneider, 1997)

	A clustering algorithm Rule-based information SVM, HOG RANSAC algorithms Unsupervised machine learning Markov Chain Approach NLP-techniques supply chain finances techniques Markov chain model, Bayesian model Dynamic heuristics MSER algorithm SegLink W-A net model DetectGAN Chargrid technique graphical convolution network FPN Bi-directional LSTM-CNNs-CRF EAST and RNN RMRS GBS-CR TLGAN CGAN Segmentation Algorithm Deep Learning model RNN FGAN		• To explore the system which was able to extract entities from various documents. • To include more key items to be extracted, especially line items, represent an interesting area of investigation.	(Sorio et al., 2012) (Medvet et al., 2011) (Cesarini et al., 1998) (Rahal et al., 2016a) (Gupta et al., 2016) (Aslan et al., 2015) (Gao et al., 2015) (Klampfl et al., 2014) (Younes et al., 2015) (Brauer et al., 2008) (Paul & Wang, 2015) (Rahal et al., 2016b) (Miloudi et al., 2016) (Tangsucheeva & Prabhu, 2014) (Gupta & Dutta, 2011) (Seppälä et al., 2014), (Koo, 2016) (Shi et al., 2017) (Anand & Khan, 2020) (Zhao et al., 2020) (Katti et al., 2018) (Liu et al., 2019b) (Zhang et al., 2020a) (Ma & Hovy, 2016) (Meng et al., 2019b) (Yi et al., 2019)

	cGAN FCA Graph Neural Network			(Kim et al., 2020) (Reza et al., 2019) (Delie et al., 2002) (Gilani et al., 2017) (Zhang, 2018) (Li et al., 2019) (Le Vine et al., 2019) (Krieger et al.)

2.4. Survey Review

In addition to academic research and analysis, there is value in identifying real-world challenges that various industries face when extracting data from invoices and receipts. Research performed in a purely academic context tends to be out of touch with many aspects of day-to-day operation. As a result, some studies that appear promising in an academic context prove useless in a practical context. This is undoubtedly true for data extraction research. For example, data extraction accuracies presented in the literature review appear promising, but the same methods fail to perform well on real-world invoices. This is challenging for the industry. Furthermore, the standard set of published data is used repeatedly by different methods and looks promising for method comparison. But in real-world conditions, the format and quality change over time. Also, many of the literature reviews used industrial software to compare respective methods. Therefore, we also need to understand real-time industry data and its utilisation in a standard manner. For this, the invoice data collection and further gap identification from the survey method were performed to gain more insight. Three financial data extraction companies were contacted to provide this insight and sample data set.

After identifying significant problems in the literature review, certain sets of questionnaires were presented to the companies. The questions were primarily based on quantitative methods. There were a few qualitative questions to capture if anything was assumed to be missed out. The interviewers were asked about missing gaps and challenges in the process of invoice data extraction. For instance, what kind of tools they are using? What kind of challenges is being faced by them?

The set of questions are presented to several types of users working at different levels like managers or data entry experts. For this, the question related to end-user designation was added. The stakeholder designation or user was added to understand the different viewpoints based on the experience and responsibilities. Now, based on the survey feedback and analysis performed, 33.33% of the users belongs to the data entry team, 22.22% are the data analyst, 16.67% accountant, 16.67% holds the director designation, and 11.11% are the managers. All the information related to designation is represented in the following figure:

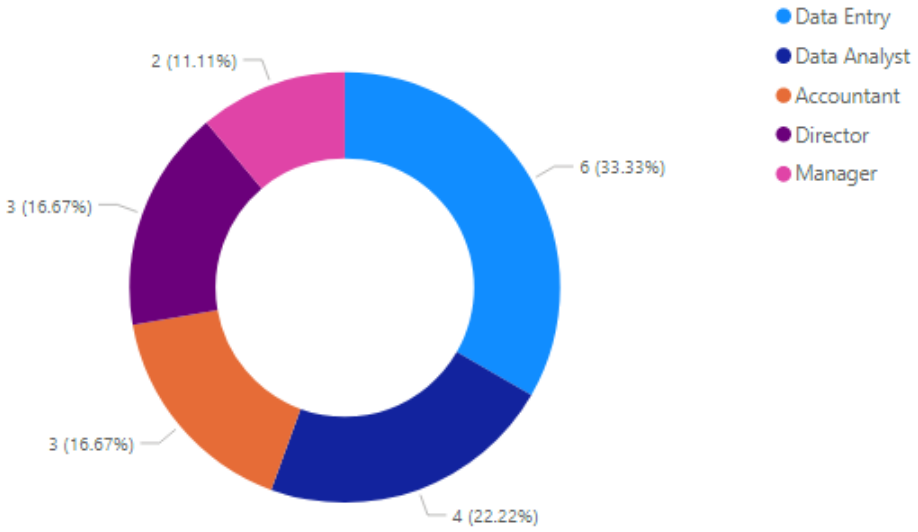


Figure 17: Survey feedback - Your designation in the company?
"Devised by the author".

In a typical scenario, the stakeholder business provides an automatic invoice data extraction solution to their client. They have an in-house OCR and data entry software where clients can use to upload the invoices. For OCR and text extraction, they have relied on OmniPage OCR API based solution. Once the client submits the invoices in the form of pdf and images, it is sent to the OCR engine. Which, after processing, returns the text and location of each text in the form of XML file. Later, this file is read, processed, and text is extracted. The extracted fields and their respective values, such are dates and tax, are then presented on the UI screen to the backend data entry team for validation for each entity. The team's job is to compare the value with the invoice image and correct it if needed. In other words, they are responsible for validating the results. They had a good system in place for this and provided automatic invoice data extraction and posted the extracted data into various accounting software programs.

Now, our objective was to enhance the existing system, apply machine learning and a unique rule-based engine so that more data can be extracted. According to the business,

the cost of validation was becoming expensive. Although research and the existing OCR industry does talk about high accuracy, in a real-life scenario, they were not getting this. Therefore, there was a constant need for an enhanced rules-based engine and the latest machine learning model, which can be combined so that templates of any unknown invoices are generated in real-time and gets enhanced in real-time. Thus, the accuracy increases with time. Furthermore, the system should be dynamic enough to adapt any kind of invoices across the globe, and all relevant fields can be extracted with few custom changes.

After carefully analysis, the discussed challenges and gaps were added in the survey and questionnaire section. With this, the following questions were presented in the form of google form and the link that was shared with the respective stakeholders:

Table 10: Survey Questionnaires.

"Devised by the author".

Invoice / Receipt Scanning Software	
Optical Character Recognition (OCR) is a process by which we can extract data from scanned documents. We are researching OCR, and we would like to collect information regarding the importance of text extraction from invoices, receipts, and bills in your business/ or individually. We would like to understand if you have used any OCR products, how much data you can extract without errors and the challenges you have faced while using such a tool available in the market. Although many companies claim that they can extract the data using OCR with 90-100 percent accuracy, the claims are usually false. **Note:** Box represents multiple selections. Circles represent single choice selection.	
1. Your designation in the company?	1. Director 2. Manager 3. Associate 4. Data Entry 5. Other
2. Are you using any invoice/receipt scanning software?	o Yes o No
3. If Yes, what is the name of the software/tool?	o None o Google OCR o Microsoft Azure OCR o Amazon OCR o Sema Media Data

	○ Taggun ○ Cloudmersive ○ Expensify ○ ABBYY OCR ○ OmniPage ○ Rossum ○ Smart Receipt OCR ○ Kofax ○ Nanonets ○ Others
4. Does the existing tool work as expected?	a) Yes b) No c) Maybe
5. What all information do you usually extract / or willing to extract from the scanned documents (Invoice, Receipts, Bills, PO)?	□ Document No □ Invoice/Receipt Type □ Date □ Due Date □ Currency □ Country □ Zip Code □ Telephone □ Fax □ Tax Number □ Website □ Email □ Bank Account □ New Amount □ Carriage □ Tax Amount □ Total □ Tax Rate □ Discount □ Item/Table Rows Extraction
6. In which of the fields do you get more than 90% accuracy in data extraction? Kindly select the options where you get / would like to get more than 90% accuracy of the extracted data.	○ Document No ○ Invoice/Receipt Type ○ Date ○ Due Date ○ Currency ○ Country ○ Zip Code ○ Telephone ○ Fax ○ Tax Number ○ Website ○ Email ○ Bank Account

	○ New Amount ○ Carriage ○ Tax Amount ○ Total ○ Tax Rate ○ Discount ○ Item/Table Rows Extraction
7. What is the average accuracy you get after the text is extracted? Between 0-100%?	
8. In general, what is the level of accuracy expected in your business? In general, people are fine with any percent. What do you think between 0%-100%, you will be happy to start using the software?	
9. Does the quality of the image impact the extraction of text?	A. Yes B. No C. Maybe
10. Where does OCR error correction or spell check mostly fail?	a) Abbreviation / Short form of words b) Full form of words c) Too long sentences d) Printing gaps in between words e) All of the above
11. What are the sources of scanned images?	1. Camera 2. Scanner 3. Email 4. Other
12. How much time (in minutes) does it take to validate the extracted data manually? * Suppose someone asks you to fill all fields manually by looking into the invoice. Then how much time you might take.	
13. How much time (in minutes) does it take to validate the extracted data automatically using the software?	
14. Does your software learn quickly from a past mistake? i.e., does it become more accurate with time?	• Strongly disagree • Disagree • Neutral • Agree • Strongly agree
15. What do you think could be the reduction of cost (in percentage) if a 5% increase in accuracy levels in data extraction is achieved?	
16. Do you think this software can help in gathering data that further helps in forecasting the demand and supply of the products/services?	1. Strongly disagree 2. Disagree 3. Neutral 4. Agree

	5. Strongly agree
17. Do you think this software can help in better spend classification of your company vendors?	1. Strongly disagree 2. Disagree 3. Neutral 4. Agree 5. Strongly agree
18. Where do you think the text extraction software can be helpful or applied in your existing business?	
19. Would you like to share some suggestions, feedback, or expectations from such data extraction software/tools?	

After collecting the feedback from different stakeholders, some of the results have been consolidated, analysed, and presented in the next section, along with the gaps that were identified from the academic literature review section. On the question ‘is the business using any invoice/receipt scanning software?’, the stakeholders responded with either a “Yes” or “No”. 83.33% used the existing data extraction software, and 16.67% were not using such type of software. The graphical representation of the data is shown in the figure below.

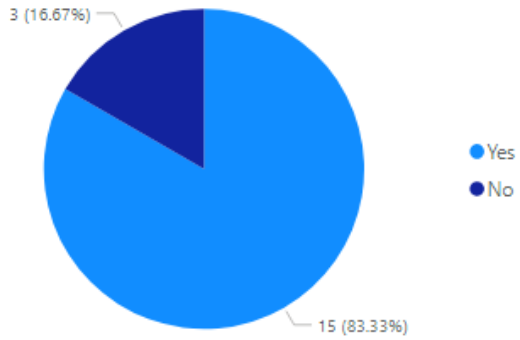


Figure 18: Survey feedback - Are you using any invoice/receipt scanning software?
“Devised by the author”.

For the question of whether the existing software/tool can help in better spend classification of your company vendors? The users responded to this question as agree, strongly agree, neutral. The 44.44% of users agreed that the software is useful for the

classification, 22.22% were neutral, and 33.33% strongly agreed, as shown in the figure below.

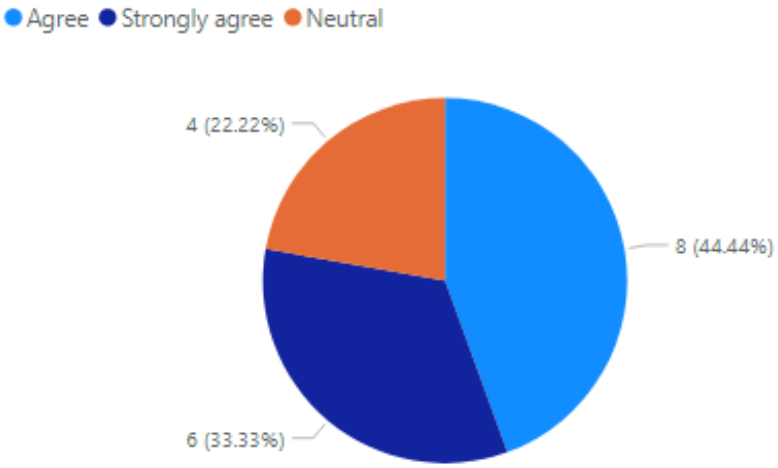


Figure 19: Survey feedback - Software can help in better spend classification of your company vendors?
"Devised by the author".

The extracted text information always helps in forecasting future demand and supply of the products/services. The user responses showed that 38.89% agreed, 22.22% were neutral, and 38.89% strongly agreed to the question shown in the figure below.

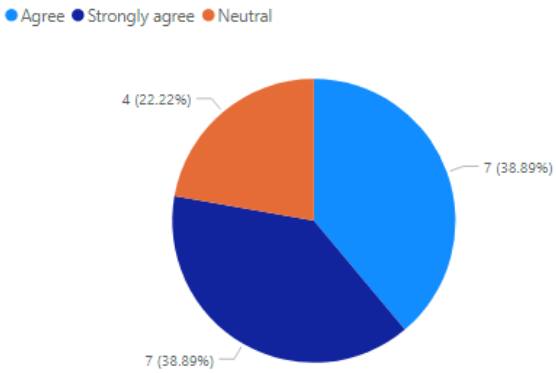


Figure 20: Survey feedback - Software can help gather data that further helps in forecasting demand and supply of the products/services?
"Devised by the author".

Furthermore, as the business is paying for the existing software/tools to extract the data, the idea was to understand if that existing software works as expected? For this question, 35.29% of users said “No”, 35.29% said “Yes”, and 29.41% said “maybe”. That means half of them accepts that still the software does not work as expected. The representation of which has been shown in the following figure.

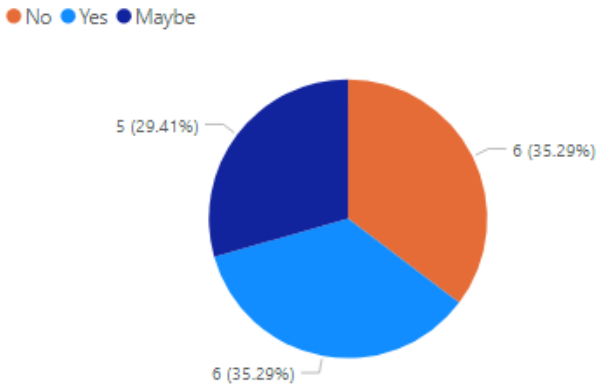


Figure 21: Survey feedback - Does the existing tool works as expected?
“Devised by the author”.

Finally, different users gave different points to help us improve the model or information about some extra features where the current system fails. According to one of the users:

> *“I think there should be an option to send money and an option to request money. There should be a list of all transactions, report etc. One user can easily find all relevant data in one place. There should be an option to fill in data manually and automated. If data extraction matches 100%, it will be good.”*

For the existing software/tool user commented that:

> *“The text extraction software can have huge application in the field of medical science; however, there is no such tool which can extract the information with 90% accuracy. Also, if some tools can be developed which can extract information from the handwritten text, that will be*

super useful in our business." And "Software should be applicable to real-time use without any constraints such as quality of image, time of computations and the extraction accuracy should be above 95%".

In another survey feedback, related to the impact of cost reduction (in percentage) if a 5% increase in accuracy levels in data extraction is achieved. The users replied that 38.89% of people responded as they will have a 10% reduction in cost; similarly, 22.22% said 5%%, 16.67% as 15%, 11.11% as 30%, 5.56% as 25% and 5.56% reacted as much as 40% reduction in cost. The 40% reduction was stated by those who were not using end to end OCR software solutions. Thus, overall, this is concluded that even a small change in accuracy does affect the outcome of data extraction and therefore helps in cost optimisation.

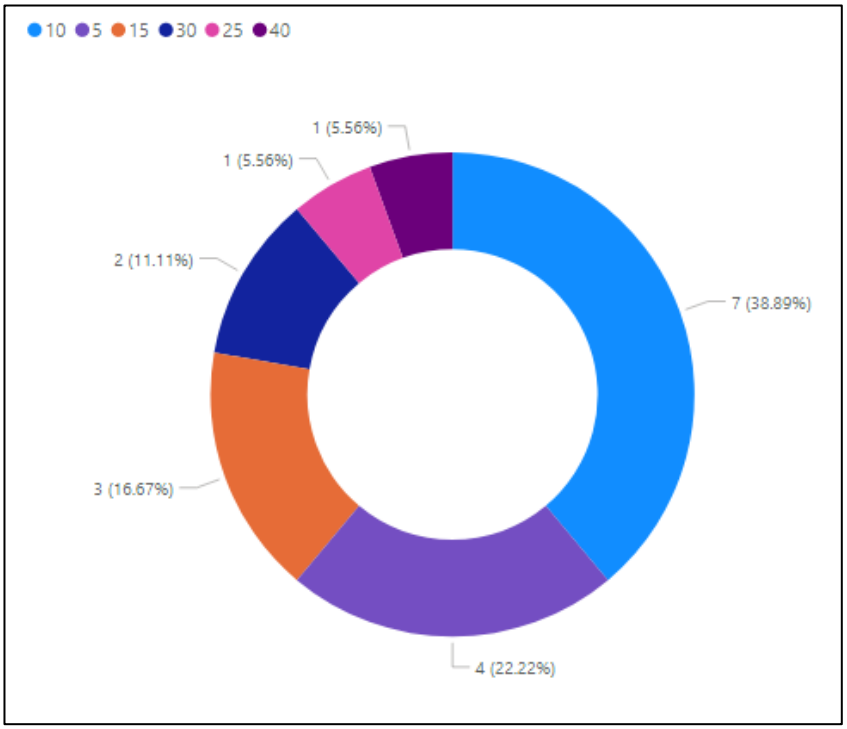


Figure 22: Survey Results - What do you think could be the reduction of cost (in percentage) if a 5% increase in accuracy levels in data extraction is achieved?
"Devised by the author".

2.5. Research Gaps

Based on the literature review presented above and the interviews conducted with relevant companies, a list of gaps has been identified. Each was either identified in an analysis of the literature or identified through interview responses offered by stakeholders within the financial sector. The objective is to understand these gaps well enough to identify viable solutions which will help to optimize the financial supply chain process in any given industry. The issues and challenges faced when processing and converting scanned invoice images have been listed in the sections below.

1. Image Quality Issues

The image types present within financial sector datasets are incredibly various. They range from grayscale to coloured, from compressed to uncompressed, and from centred text to text-oriented along the side. Images also come in various languages and fonts. It goes through so many hands that the quality gets affected. So, it is difficult to extract text from images due to the discontinuity in colour, size, or orientation and various other factors. Thus, the quality of the image affects the outcome of OCR text. If the image was not taken in a good background with enough light, quality reduces due to darkness (Köppen et al., 1998). If the camera flashlight gets onto the image, then the text becomes invisible or flashy. Furthermore, when the image is stored in low resolution or a compressed file format, the quality is still reduced. Also, when a page in a pdf format is extracted for OCR, the extraction quality determines the output of OCR text (Zhang et al., 2009). The image quality should be improved to achieve better results (Liu et al., 2016b). The model (Jacobs et al., 2005) works for the low-quality image, but not all text is extracted correctly. The model should work for both the super-resolution and low-resolution images for detection (Rabbi et al., 2020). The framework should also work for the real-time Visually Rich Documents (VRDs) without attempting an error in case of detection of text and extraction (Pegu et al., 2021). Even as per (Zhu et al., 2019), the real-time detection of text was not always possible.

As per the survey, 100% of the respondents stated an image quality issue, and the existing tools do not work as expected. To find the required pattern or extracting the text creates several issues. If the quality of the image is less or simply blurred, the existing models fail to extract the text. Therefore, the model should be trained well and smart enough to identify specific OCR related issues.

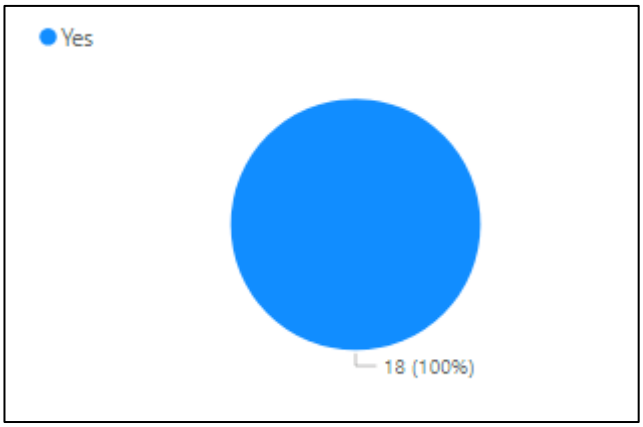


Figure 23: Survey feedback - Does the quality of the image impact the extraction of text?
"Devised by the author".

Furthermore, to extract text from the image of invoice/receipt, several sources are available nowadays, such as a scanner, camera, and email. Based on the response from the survey, 44.44% uses the camera, and 55.56% uses the scanner to take a photo of a particular invoice. So again, there is an issue with the resolution as it is different for every model. Illumination and distance also play a vital role at the time of capturing a photo. So, the text extraction depends on the source of capture also. Moreover, moving forward, the usage of the camera is increasing day by day.

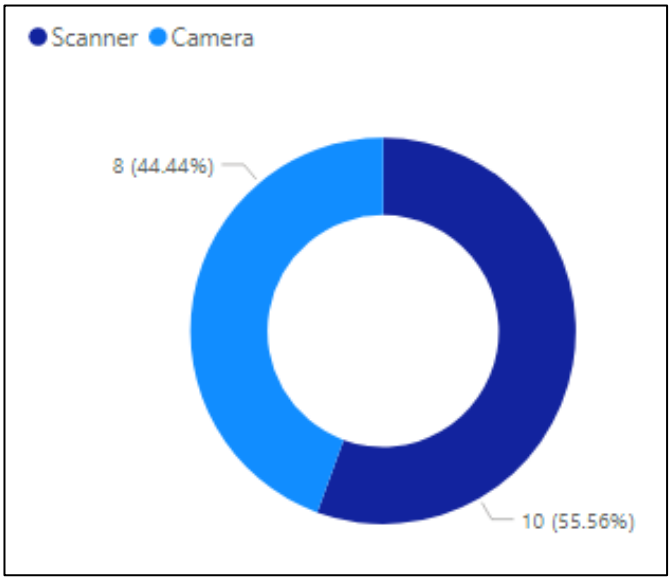


Figure 24: Survey feedback - What are the sources of scanned images?
"Devised by the author".

2. OCR Engine

The OCR engine uses an adaptive classifier that is font sensitive. When a suitable word is identified within the phase, it is transferred to an adaptive classifier as training data. The baseline normalisation makes it easier to distinguish upper-case and lower-case characters as well as improving tolerance to noise specks. However, different OCR engines behave differently, and further customisation is needed for better extraction. Also, OCR engines are generic in nature and do not adhere to all kinds of business requirements. Thus, it becomes challenging to use the same configuration-based OCR for all the work (Miloudi et al., 2016). Either the existing OCR can be used for text extraction, or a machine learning-based enhanced OCR engine can be used. The industry is already using machine learning-based methods in many OCR software, but the research lacks to show a significant amount of field identification and extraction. The real-time error detection from the output of OCR is also not possible (Vinitha & Jawahar, 2016). In the existing market number of tools are available, and those are mostly the paid software. The name of the software is such as Google OCR, Microsoft Azure OCR, Amazon OCR, Sema Media Data, Taggun, Cloudmersive, Expensify, ABBYY OCR, OmniPage,

Rossum, Smart Receipt OCR, Kofax, Nanonets etc. Based on the survey carried out, 33.33% people are using the ABBYY FRE tool, next 33.33% extracts the data with the help of Microsoft Azure OCR and rest 33.33% uses the OmniPage.

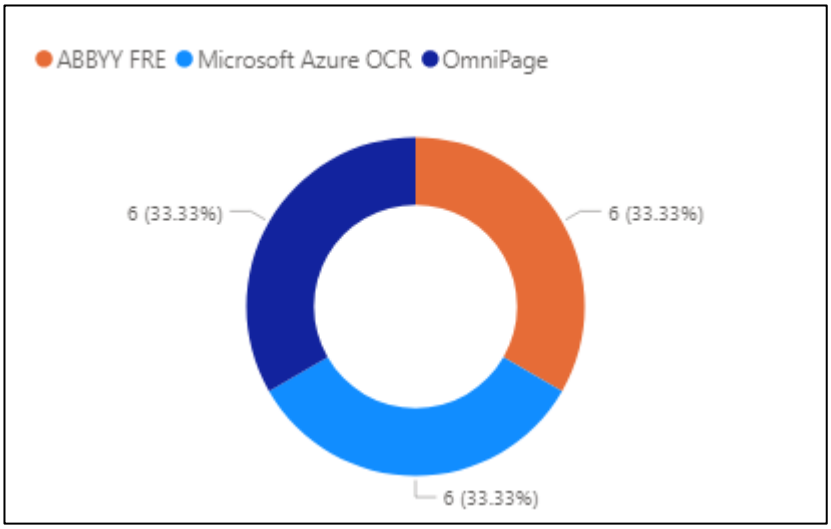


Figure 25: Survey feedback - What is the name of the software/tool?
"Devised by the author".

Furthermore, it was accepted by the survey users that the existing system still lacks expected accuracy.

> *"The existing tools/ software accuracy does not seem to be as good as promised by the company. On enquiry, they will justify and say we will get back to you."*

This means the existing tools are not providing good accuracy and not resolving the problem faced by the client. So, accuracy is one of the challenging tasks for the research and entirely depending on OCR software will not solve the problem.

3. Data Block Combining Problem

Attention modelling was proposed for text recognition to improve object detection and to caption during complicated tasks (Jain et al., 2017). When printed or due to print alignment issue, OCR returns a block of characters that is closed enough. These returned

OCRed text blocks are then combined with nearby blocks to form a composite data block for better extraction. Later, these blocks are tagged with a field key (Rahal et al., 2016b) by which kind of information was extracted. Once tagged, this block cannot be used for another field extraction. Now, there are possibilities that two or more field keys text might have been combined as a single data block. Therefore, data extraction becomes an issue due to multiple tagging. Furthermore, sometimes two related word blocks are quite far away in an image and thus do not combine. Then extraction again becomes challenging (de Jager & Nel, 2019). As such, enhanced understanding and then separation of the data block for vicinity and value is required.

4. Supporting Data Knowledge for Better Extraction

Continuous addition of new supporting knowledge data is also a bit tedious and challenging task. Some of this knowledge is common, and some are business-specific. Applying different logic based on image type, business needs, and industry is a huge problem. In one of the survey feedbacks for the question, “Does your software learn quickly from a past mistake. i.e., does it become more accurate with time?” The users responded to this question as strongly disagree, disagree, neutral, agree, strongly agree. This question was asked to check whether the supporting data extraction knowledge is required to get better extraction. The graphical representation below shows that 44.44% of people agreed, meaning more accessible data extraction knowledge is needed. It also shows that 5.56% of respondents disagreed, 11.11% strongly disagreed, and 38.89% responded neutrally.

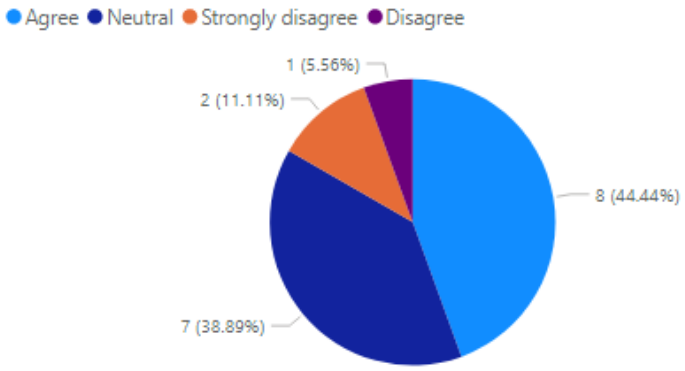


Figure 26: Survey feedback - Does your software learn quickly from a past mistake. i.e., does it become more accurate with time?
"Devised by the author".

5. Variance in Business Requirement

There was frequent variation in input data and different expectations for the output data within the same business due to differences in invoice templates and business requirements. Input invoices may not follow specific criteria or standards for some clients and while for others, it was submitted in a well-defined format. Even multiple invoices might be scanned on the same page. First, invoices need to be split and read separately. Additionally, some invoices run across multiple pages with no separation of start and end within a single file. Proper standard input was a need but did not work all the time (Esser et al., 2013). The bugs report should be improved for the analysis (Anand et al., 2020). In the feedback, one user is commented as

> "*In financial markets, there is a big shock to the world of trade where LIBOR is phasing out and new IBOR will be applied to existing contracts so, if the software is able to gauge the usage of LIBOR in the contracts like Pricing, Collateral Payment, Valuation etc. If that happens, this could be a huge win. Currently, all that is done manually, and it takes huge human effort.*"

Therefore, there is a need to improve the existing model so that the finance industry can store the required data with a single click and without doing much manual entry. So, there is scope for text extraction in the finance industry with complete automation.

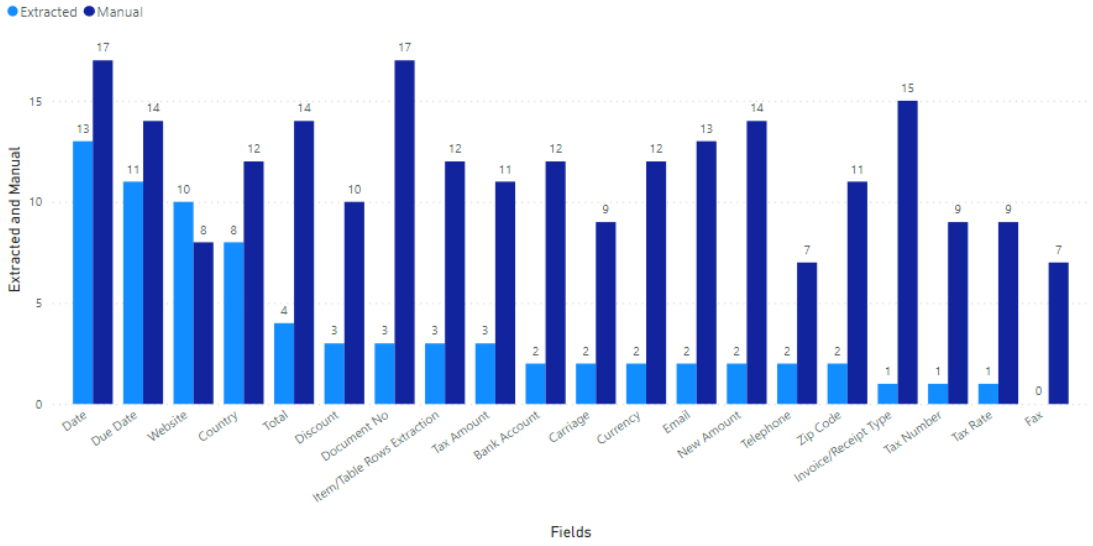


Figure 27: Survey Result: Manual time vs Automatic time.
"Devised by the author".

The above comparison chart shows that how much time is required to extract the named entity. The users responded that different fields, such as extracting the "date", on average would take around 13 minutes with the help of software and around 17 minutes to do it manually. For each of the fields, automatic extraction time versus manual time is represented in the figure above.

6. Multiple Similar Keywords

Similar keywords were used to store multiple pieces of information in the same scanned document. For example, total, grand total, net total, due date and bill date may exist in a single document (Dave et al., 2020). Again, similar keywords were used in key-value field items as well as in table level data. Extraction becomes difficult in documents that exhibit this (Chiang & Knoblock, 2011). For example, if the word 'grand' or 'net' is not read correctly, the key risks being read and interpreted as 'total'. When this occurs, the

wrong value is extracted (Grönlund & Johansson, 2019). Understanding nearby key relations and value matching between them is a need in the research.

7. Line / Table Item Extraction

There was a gap in the extraction of line items from the table. The extraction becomes challenging due to variation in line-items, multiple column-based data structures, short-hand table header/data, and no proper alignment in columns/rows (Köppen et al., 1998). Even if a column and table style was identified, data rows printing was used (Medvet et al., 2011). The extraction of table contents (Aslan et al., 2015) and the structure of table identification are not good (Gilani et al., 2017). Still, the table data extraction has an issue (Liu et al., 2020). As per (Reza et al., 2019), there is a need to detect tables in original size as well as to investigate the overlap of columns to improve the model further. (Krieger et al.) have mentioned that there was a scope to explore the model for line-item extraction to extract more key items. As well as model was able to extract key items from further document types such as receipts, purchase orders, etc.

As per the survey feedback, the existing software/tools are facing challenges in *"reading and processing documents"* correctly. In every receipt/invoice, there is a different layout style, which creates a significant problem when processing and maintaining the same quality of the image. So, to write the logic which will satisfy all the constraints is a little bit challenging task.

8. Top-Down versus Left Right Matching of Key-Value

In key-value pair matching of text, data can be found on the right side of the key or bottom of that key. It works well if the input template was known. If the template was unknown, then considering which value to consider was again a challenging task (de Jager & Nel, 2019). When several string values in the OCR output file follow a regular expression pattern, key-value matching becomes more difficult (Cinti et al., 2020). None of the current financial reimbursement systems can accurately describe the condition depicted

in the chart, which depicts a discrepancy between the original main fields and the invoice image's equivalent details.

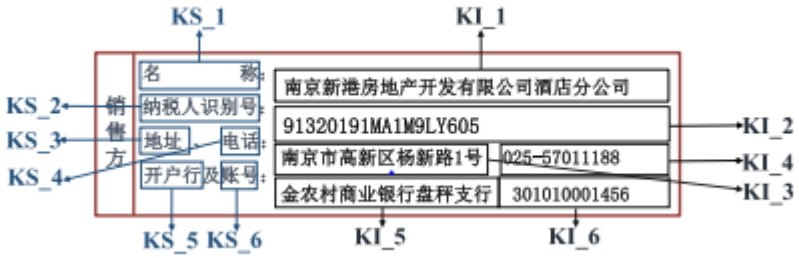


Figure 28: The block diagram of key segmentation model. (Meng et al., 2019a).

9. Printing Position or Short Form of Words

An invoice's text is not always printed according to a pre-defined invoice template. As such, invoice date, the key 'Date:' and the value '01/01/2017' may not occupy a single line. In the top-down presentation, the value was not precisely below the key (Medvet et al., 2011). Due to a large amount of text. Few places were missed. Therefore, detecting text was still a gap and identifying characters with accuracy was an open question to the researcher (Jun et al., 2019). There is another major issue where OCR error correction or spell check fails. The short-form words such as 'Qty', 'dt.', 'inv', 'tot amt', special symbols, punctuation marks, digits may not pass successfully if there are any issues with OCR text identification (Ho & Nagy, 2000).

Based on the questionaries asked in the survey review, one challenging question related to spell check failure. The multiple choices were given to this question, 55.56% responded as the OCR error correction fails in all, 5.56% said in too long sentences is a significant factor in extracting key-value pair, 27.78% replied that the short form of words in sentences creates issues, 5.56% commented for the printing gaps between the words, and at last 5.56% voted for the complete form of words also creates a problem. It means the structure of the sentence, length of sentences, printing gaps are still challenging tasks. Even if the researchers have commented that they resolved such issues to a reasonable extent, it still fails in a real-time application.

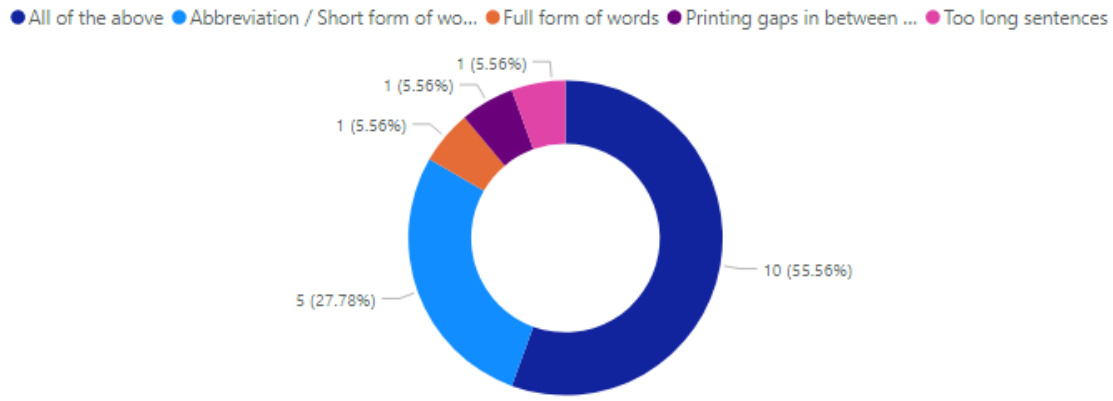


Figure 29: Survey feedback - Where does OCR error correction or spell check mostly fail?
"Devised by the author".

10. Locale Support

As invoices are shared across geographical boundaries, the existence of multiple currencies and addresses becomes a challenge in identifying locale that needs to be recorded for a given invoice (Teunissen, 2017). Many research-based datasets are based on Chinese characters. This dataset does not always perform well in English based invoices and receipts.

11. Handwritten Data

Certain invoices also include handwritten notes which were either not read correctly or add additional issues to printed text in the forms of overlay and overwriting (Köppen et al., 1998). If the OCR engine was configured to read printed information alone, it did not read handwriting properly. If the engine was configured well, then overall OCR text extraction accuracy becomes another problem (Agarwal et al., 2007). The handwritten text classification should be improved so that the invoice data extraction in a digital format becomes easy (Geetha et al., 2020). In survey feedback, one of the users commented,

> *"The text extraction software can have huge application in the field of medical science; however, there is no such tool which can extract the information with 90% accuracy. Also, if some tools can be developed which can extract information from the handwritten text, that will be super useful in our business."*

Another user responded, saying:

> *"As of now, its extracts data from electronic receipts efficiently. But if the receipt is handwritten, then it is unable to interpret. I would like to see in future that can interpret handwritten receipts also efficiently. My rural clients mostly submit manual receipts. And sometimes, it is challenging to understand the data for the software. I would also like to see if this software can be enhanced for scanning handwritten receipts of local Indian languages."*

Therefore, these gaps are also required to be worked upon and need further investigation.

12. Invoice fold marks

If the invoice and bills paper is folded, skewness and raised areas may create an issue that makes OCR engines return text less accurately (Kundu et al., 2020). Furthermore, if the folded line location was too dark or created a mark in the invoice, text gets damaged (Shivakumara et al., 2005).

13. Grammatically Incorrect

Due to the lack of dataset availability, many models fail to generate the expected output. In many cases, the model generates grammatically incorrect text (Wickramarathna & Ranathunga, 2019). In this regard, setting accurate parameters at the time of model building plays a vital role. It may generate or predicts the wrong character or word. There should be a proper end-to-end mapping between the encoder and decoder to avoid the wrong mapping of data.

14. Structure of text

Now, because the invoices have text with different font sizes (Kim et al., 2020), it was interesting to apply multi-RPN and evaluate the result (Nagaoka et al., 2017). These were the known limitation in invoice data extraction, especially with retail receipts where font sizes were small (Zhong et al., 2019a).

To conclude, many limitations in the existing methods have been identified in this section. The goal is to reduce the gap further and resolve as many challenges as possible. The following two chapters will highlight the solution and resolve the maximum challenges faced by the researchers and current data extractor tools/software in the industry. The detailed solution is explained in chapter 3 for many data extraction-related problems, and chapter 4 is specific to the error-correction problem.

2.6. Dataset Identification

After going through the various research paper and even collecting existing data from the industry, the following dataset has been identified and finalised. In the following chapters, these data sets will be used for comparative study.

1. **SROIE Dataset:**

(SROIE, 2020) is a publicly available dataset that is related to receipt information extraction in ICDAR 2019 challenge. The dataset has 626 receipts for training and 347 receipts for testing. Each receipt has 4 entities such as company, date, address and total (Huang et al., 2019a), as most of the existing academic approaches used the SROIE dataset for text detection only. This dataset was considered to extract the entities of tax invoices such as date, invoice number, vendor name, vendor address, and email. The below table summarizes the use of the SROIE dataset in recent years.

Table 11: SROIE Dataset Summary.

"Devised by the author".

Area	Method	Dataset with Accuracy (F1 score)	Gaps identified	References
text-line detection	MSER algorithm SegLink W-A net model DetectGAN	Own dataset= 0.9329 ICDAR 2015=75 SROIE= 89.65% SROIE= 98.74%	1. To extend SegLink for an end-to-end recognition system. 2. To work on memory management and time reduction. 3. Future work on complex background images.	(Koo, 2016) (Shi et al., 2017) (Anand & Khan, 2020) (Zhao et al., 2020)
scene text detection	FASText Keypoint Detector FCN PSENet	ICDAR 2013= 75.9 ICDAR 2015= 0.8737 ICADAR 2017= 94.34%	1. The future scope was to detect the curved text, text recognition and general object detection.	(Busta et al., 2015) (Zhou et al., 2017a) (Li et al., 2018)

Furthermore, the following table shows a comparative accuracy evaluation done on the SROIE dataset by different authors. The DetectGAN (Zhao et al., 2020) presents the best F1 score.

Table 12: Performance comparisons (F1-score) on SROIE dataset (SROIE, 2020).

Method	Precision (%)	Recall (%)	F1 score (%)
Koo's (Koo, 2016)	68.53	72.81	70.61
Fastext (Busta et al., 2015)	69.69	81.89	75.30
EAST (Zhou et al., 2017a)	87.65	79.5	83.37
SegLink (Shi et al., 2017)	89.02	92.75	90.67
PSENet (Li et al., 2018)	96.70	92.40	94.34
W-A net (Anand & Khan, 2020)	83.60	86.10	84.83
DetectGAN (Zhao et al., 2020)	98.98	98.51	98.74

2. **VATI Dataset**:

VATI dataset (VATI, 2021) has a total of 5000 images of taxi invoices. These invoices contain different entities such as number, distance, amount, waiting, price, and code. This dataset is also used in recent research and shows field-level accuracy.

Table 13: VATI Dataset Summary.
"Devised by the author".

Area	**Method**	**Dataset with Accuracy (F1 Score)**	**Gaps identified**	**References**
Invoice entity extraction	Chargrid technique graphical convolution network FPN	Scanned invoices= 84.28% VATI = 0.881 VATI= 93.26%	To work on the document classification task	(Katti et al., 2018) (Liu et al., 2019b) (Zhang et al., 2020a)
Invoice entity recognition	Bi-directional LSTM-CNNs-CRF	POS= 97.55% and CoNLL = 91.21%	The future scope was to explore a model for multi-task learning	(Ma & Hovy, 2016)

The following table shows the accuracy comparison for each field extracted by different methods on the VATI dataset.

Table 14: Performance comparisons (F1-score) on VATI dataset (VATI, 2021)

Entities	**Chargrid**	**NER**	**GCN**	**TRIE**
Code	89.4	94.5	97.0	98.2
Number	85.3	92.4	93.7	95.4
Date	89.8	82.5	93.0	94.9
Pick-up time	82.9	60.0	86.3	84.6
Drop-off time	87.4	81.1	91.0	93.6
Price	93.0	94.5	93.6	94.9
Distance	92.7	93.6	91.4	94.4

Waiting	89.2	85.4	91.0	92.4
Amount	80.2	86.3	88.7	90.9
Average	**87.77**	**85.59**	**91.74**	**93.26**

3. Realtime Company Dataset:

The business provides automated invoice data extraction solutions to their client. They have an in-house OCR and data entry software where their clients can use to upload the invoices. For OCR and text extraction, they have relied on OmniPage OCR API based solution. Once the client submits the invoices in pdf and images, it was sent to the OCR engine. Which, after processing, returns the text and location of each text in the form of an XML file. Later, this file was read, processed, and the text was extracted.

Now, our objective was to enhance the existing system and apply machine learning and a unique rule-based engine so that more data can be extracted. According to the business, the cost of validation was becoming expensive. Although research and the existing OCR industry talk about high accuracy, they were not getting this in a real-life scenario. Therefore, there was a constant need for an enhanced rules-based engine and the latest machine learning model, which can be combined so that templates of any unknown invoices were generated in real-time and gets enhanced slowly. Thus, the accuracy increases with time. Furthermore, the system should be dynamic enough to adapt to any kind of invoices across the globe. Moreover, all relevant fields that can be extracted should be available. The above-discussed challenges and gaps are already stated in the survey and questionnaire section of this chapter.

Henceforth, to validate the accuracy, the business provided 40,000 invoices that were supposed to be tested and compared with their existing system. The dataset consists of invoices and receipt images from English speaking countries. That means, data was in the English language, and there are no predefined templates. The images can be of any format, quality, and size. These were real-time images received from businesses from the time between 2010 to 2015. The company was already using an existing system and were

getting a variable accuracy for each field extracted. An expected baseline percentage was provided to test our system and validate how good we can do. For the experiment evaluation, predefined 3300 real-time mixed invoice/receipts images were used. This was selected to cover all kinds of issues in data extraction. The dataset was split into 70% for training and 30% for validation.

A survey also validated the expected data extraction. Results show that 16.67% of users expect up to 95% accuracy, 27.78% expect around 90% accuracy, 33.33% expect around 85% accuracy, 16.67% expect 80% accuracy, and 5.56% would accept 75% accuracy. It also depends on how much data they want for their business or need. The below diagram illustrates the outcome of the feedback.

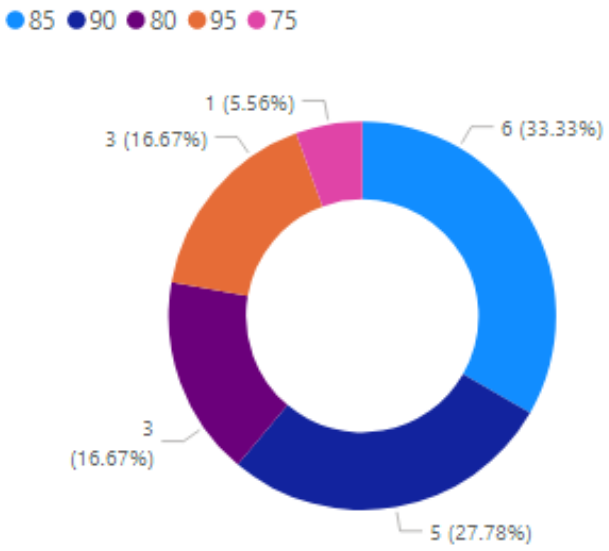


Figure 30: Survey feedback - In general, what is the level of accuracy expected in your business?
"Devised by the author".

Furthermore, the below tables represent 'Company A' baseline accuracy for the field extraction. They have given the different field names that are required to be extracted.

Table 15: Company A baseline expectation.

"Devised by the author".

Field Name	Baseline Accuracy
Invoice No	71.00
Document Type	75.00
Date	80.00
Vat No	80.00
Currency	75.00
Discount	75.00
Carriage	70.00
Tax Rate	75.00
Tax Amount	75.00
Net	75.00
Total	85.00
Tele	80.00
Fax	50.00
Website	90.00
Email	50.00
Bank Account	50.00
Due Date	50.00
Zip Code	50.00
Country	70.00
Average	**69.5**

Similarly, the line-item level fields are also required to be extracted, and they have defined the accuracy separately as follows:

Table 16: Item table and Tax table fields.

"Devised by the author".

Field Name	Baseline Accuracy
1. Item Table	
Product Code	70.00
Description	85.00
Quantity	80.00

Unit Price	75.00
Net Amount	85.00
Tax Amount	85.00
Tax Rate	85.00
Tax code	85.00
Total	90.00
Discount	90.00
Average	**83.00**
2. Tax Table	
Net Amount	60.00
Tax Amount	60.00
Tax Rate	60.00
Average	**60.00**

2.7. Conclusion

In this chapter, an introduction to the text extraction system was presented. Studies relating to existing technologies and OCR tools such as Google OCR, Azure, Amazon, SemaMediaData, Expensify App, Abbyy, and Rossum, were discussed. Then, a literature review was presented related to text extraction and, later, a survey review was performed. Finally, gaps and relevant datasets were identified for further research work.

The study of existing OCR tools contains the details of the algorithm or model used, the image format supported by the system, the accuracy achieved, and the source of methods and the tool's limitations. The existing systems use OCR and machine learning systems to detect the text. Existing OCR systems were also creating bounding boxes for text detection, but this capability must be enhanced. An enhanced rule-based algorithm was identified to get the exact text or character information of a particular box. The machine learning-based algorithms used for text detection, such as CNN, R-CNN, Faster R-CNN (Ren et al., 2016), and GANs, were studied for further enhancement wherever appropriate.

The survey review was performed on three companies to evaluate existing OCR applications and software. For that, the questionaries were created based on multiple interviews with the stakeholders and after identifying gaps in the existing research. The questionnaire includes all the details related to an existing system (e.g, what is the time required to extract the accurate text from the given invoice image?; What are text detection and extraction accuracy?; How many fields are correctly classified?; Any suggestions to improve that APP or software?) Next, the gaps identified in the literature review and survey were explored. The gaps related to data extraction, the quality of image, detection, and many other areas were identified. After identifying the research gaps and business requirements, the primary focus shifted to selecting an appropriate dataset. For the dataset, publicly available datasets in the literature review, the researcher's still used were considered. In the dataset study, the details are tabulated, such as accuracy, the number of fields extracted and the limitations of the dataset, such as the number of images less, low quality.

3. A Novel Expert System

3.1. Introduction

In this chapter, several solutions to the gaps identified in the previous chapter are proposed, implemented, and the results are discussed. In the previous chapter, a list of current challenges in invoice data extraction and, more generally, issues created by OCR were discussed. The chapter contained a detailed study of different approaches taken using the latest machine learning techniques. The latest work related to ANN, RNN, CNN, and other related research was studied for better text extraction. That included Faster-RCNN, the preferred machine learning algorithm for object detection. It was one of the main algorithms used for target detection on the image. The Faster-RCNN algorithm and OCR technology were used to build a machine learning model that recognizes and extracts text on the scanned image. In the research-based study, the GANs model gave brilliant results in small object detection, text detection, table localisation etc. Therefore, The GANs is considered as a possible candidate to detect the text from scanned invoice images.

This chapter will start with the study and evaluation of text block detection and text extraction. We will evaluate the best model based on the previous study and synthesise it with specific requirements related to invoice data extraction for text block detection. For text extraction, we will explain the processes of text extraction which were considered in designing our solution framework. We will try to understand how data is arranged on an invoice and, using this understanding, develop a relationship between nearby information such as tax and the total amount that are placed together or customer and address on the basis of invoice location. With this, we will finalize an approach to increase the accuracy of data extraction further. Again, this will involve a comparative study of different concepts with other research work related to solving text and object detection on images.

Once the OCR is performed on an image that contains information about invoices and bills, text extraction needs to be performed to identify meaningful data within it. Many

issues affect the quality of text extraction, such as variance in layout, design template, print style, font, paper quality, image quality, and language. This chapter explains the system designed from zero to extract data from an XML-based file provided by the OCR engine. Apart from text, this XML file contains various information in the form of metadata. This information will be used to extract most of the data in the best possible manner. Once this was done, the output file in the form of XML format was read, and identification of the text was made in the pre-processing stage. Later, after individual data is extracted, specific business rules were applied to get the final output in the form of key-value pair. The following subsection will explain the solution methodology applied, steps performed to correct data, filter incorrect data and finally extract possible corrected data. It will also present the performance evaluation achieved for the same. Finally, the gap and limitation shall be discussed.

3.2. Text Block Detection

In this section, we achieved the second research objective that is 2(a) "*To understand and apply various machine learning and pattern recognition techniques to increase accuracy*." The different algorithms are used to identify various critical objects or aspects in the image according to their importance and differentiate one from the other. In the case of object and text block detection, we will use a machine learning model to identify the block's position, maintain a learning mechanism, and save the knowledge in the database for future model training and testing purposes. When an image is passed through this machine learning-based algorithm network, it passes through many layers such as convolutional, non-linear, pooling and fully connected layers. In CNN architecture, a convolutional layer always comes first. The computer sees the image as a matrix of pixels, and it starts reading the matrix from the top left of the image. A smaller matrix in the region is called a filter or neuron. Convolutions are produced as the filter moves along with the input image. The filter then multiplies its values by the original pixel values,

which are then summed up with a number. The filter then moves further right by 1 unit each time to read the remaining part of the image and perform pixel multiplication. When the filter has passed across all positions, a matrix is obtained. This matrix is smaller than the input matrix.

The network consists of several convolutional networks. When an image was passed through the network, the output of the first convolutional layer becomes the input for the second convolutional layer. This process continues with each subsequent convolutional layer in the network. The next non-convolutional layer after the convolutional layer is nonlinear. Without the nonlinear property of this layer, a network would not be sufficiently intense and would not be able to model the response variable (as a class label). A pooling layer follows the nonlinear layer. This layer performs a down-sampling operation on the width and height of the image. This results in the volume of the image being reduced. The objective of this section is to recognize and extract text data by assigning multiple predefined labels to the input image. The coordinates of the bounding box on the image are extracted and stored along with the image size for future reference. The three invoices/receipts datasets are used for testing and validation. The machine learning-based approaches are used for data block identification.

In the paper (Ren et al., 2016), Faster R-CNN was introduced as an object detection algorithm that lets the network learn the region proposals and removes the requirement of a selective search algorithm. The image was fed into a convolutional network, which generated a convolutional feature map as a result. A separate network was used to predict the area proposals. Initially, the expected area proposals will be reshaped using the RoI pooling sheet. This layer then helps to classify the image within the proposed region and predict the coordinates of bounding boxes. Faster R-CNN was much faster than R-CNN and Fast- RCNN. It was used for real-time object detection. The OCR Invoice Classification model uses Faster R-CNN for label prediction and text recognition. The text was treated as an “object”, and specific labels were assigned based on its features.

When a user gives a digital invoice as input to the model, the F-RCNN algorithm was able to predict the position of text in multiple regions of interest on the image. Thus, with the help of predicted coordinates, rectangle bounding boxes are drawn on the invoice. The next stage was extracting text and numbers from these boxes. Afterwards, the researchers discussed block detection, the evaluation of the Faster R-CNN model. Firstly, the researchers implemented the R-CNN, then Fast R-CNN and finally the Faster R-CNN model. The comparison chart gives a clear idea of how the Faster R-CNN network is better than those two models. The detailed working of Faster R-CNN is given in the case of object detection/ text detection.

	Method for generating region proposals	Test Time (in secs)	Speed Up
R-CNN	Selective Search	49	1x
Fast R-CNN	Selective Search	2.32	25x
Faster R-CNN	Region Proposal Network	0.2	250x

Figure 31: Comparison between R-CNN, Fast R-CNN and Faster R-CNN (Ren et al., 2017).

Based on the figure above, we may say that the Faster R-CNN was better than the R-CNN and Fast R-CNN and how the model improves each time (Ren et al., 2017).

1. The R-CNN uses the SVM as a classifier and the selective search algorithm to search the ROI, which was slow and to complete an operation requires more time.
2. In the Fast R-CNN model, the input image was directly passed to a CNN model, which generates the feature map. The feature map has many ROI proposals from which the ROI pooling extracts the regions. ROI pooling has a fixed window size. In Fast R-CNN, the original image was passed directly to a CNN, which generates a features map. Instead of using the number of SVM, only the softmax function was used, which has better performance than the SVM. It takes 2 seconds per image to detect the object, which fails in the case of a large dataset.

3. In the Faster R-CNN, the drawback of the above was improved by using the region proposal network.

We proposed this architecture of a faster R-CNN model for the block identification, as shown in the figure below.

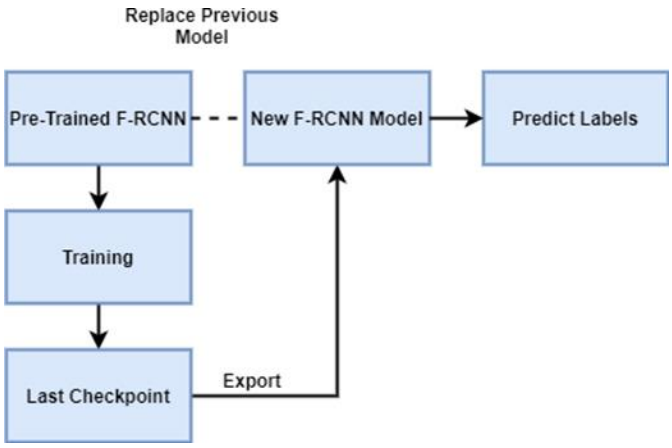


Figure 32: Proposed Training Architecture.
"Devised by the author".

The proposed training architecture requires the labelled data. So, the first task is to label both the training and testing datasets images. This is done to assign labels to multiple objects in an image with the help of bounding boxes. Using the LabelImg tool, the data on the invoice is labelled as predetermined labels such as "address", "date", and "amount". The invoice labels, along with bounding box rectangle coordinates, are saved in the database. LabelImg tool gives the output file in the two extensions or format as VOC XML or YOLO text. The user interface of the LabelImg tool is as shown in the figure.

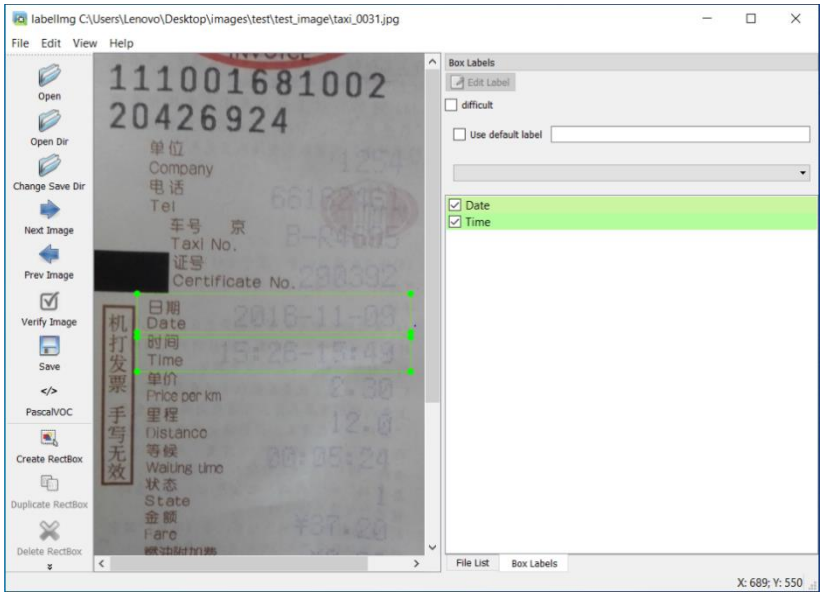


Figure 33: The user interface of the LabelImg tool.
"Devised by the author".

After setting the required parameter, the next part is to train the network. Training is a method to evaluate the performance of an algorithm on a specific problem. The training dataset is used by the algorithm to learn and make observations. This prepares a model for training. Another data set is used as test data where the output values were withheld from the algorithm. From labelled examples, a model is trained by deciding the appropriate values for all of the weights and the bias. A machine learning algorithm constructs a model by inspecting several examples in supervised learning. The objective is to find a model that minimises loss. A value indicating the magnitude of the model's prediction error on a single example is called a loss. The prediction was ideal when the loss was zero; otherwise, the loss was more significant. Model training aims to achieve a low loss. As a result, the aim is to find a collection of weights and prejudices that, on average, have low loss across all instances (ML, 2021).

Training requires a sample dataset. In this project, around 3300 invoice images are collected. 70% of invoice images are used for the training model and 30% for the testing. The solution to our problem lies in the premise of supervised machine learning. Hence,

predefined labels “address”, “date”, and “amount” are used to classify regions containing text on the image. These regions are treated as objects while classifying. The coordinates of each image and the text's location are stored in a comma-separated values (CSV) format file. The machine learning model is trained using these image coordinates. Before training, the dataset was split into two parts, training and testing. During training, a loss of below 0.05 is desired for better accuracy. When this is achieved, the training is stopped. The model is exported and is ready to predict labels. If the training does not give a satisfactory result, the model is re-trained using a corrected dataset. For every new training process, the previous version of the model is saved, and the updated version will replace the older version in the pipeline.

The GAN-based method research has been carried out based on further research and development in the data block detection models. GAN is an unsupervised learning algorithm of a neural network. The first GANs model was developed by Ian J. Goodfellow in 2014. The network has two models named discriminator and generator. A generator model is used to generate a photorealistic image based on the applied input image and loss or noise. The discriminator network continuously checks the generator's output that is generated image is real or fake by comparing it with the ground truth image or dataset image. Based on the literature review of the GAN model, it has superb results for image-to-image translation problems, super image resolution, and many other areas of research. In the case of the detection of small objects from a remote sensing image, EESRGAN gives the 95.5% average precision value for the COWC dataset. The proposed SR network with faster R-CNN has provided the best results for small objects on satellite imagery, according to experimental results. (Rabbi et al., 2020).

Model	Training Image Resolution-Test Image Resolution	COWC Dataset (Test Results) (AP at IoU = 0.5:0.95) (Single Class-15 cm)	OGST Dataset (Test Results) (AP at IoU = 0.5:0.95) (Single Class-30 cm)
EESRGAN + SSD	SR-SR	89.3%	81.8%
EESRGAN + FRCNN	SR-SR	**95.5%**	**83.2%**
ESRGAN + SSD	SR-SR	88.5%	81.1%
ESRGAN + FRCNN	SR-SR	93.6%	82%
EEGAN + SSD	SR-SR	88.1%	80.8%
EEGAN + FRCNN	SR-SR	93.1%	81.3%

Figure 34: The EESRGAN method result analysis table (Rabbi et al., 2020)

The TLGAN model (Kim et al., 2020) was used to localize the text in the document. The invoice dataset was used for the experimental analysis of Scanned Receipts OCR and Information Extraction (SROIE). TLGAN has achieved a 99.83% precision value for the SROIE dataset. The comparison of TLGAN with other methods as shown below,

Rank	Date	Method	Recall	Precision	Hmean	Ref.
-	***2020-10-19***	***TLGAN (ours)***	***99.64%***	***99.83%***	***99.91%***	
1	2020-08-10	BOE_AIoT_CTO	98.76%	98.92%	98.84%	[21, 20]
2	2019-04-22	SCUT-DLVC-Lab-Refinement	98.64%	98.53%	98.59%	N.A.
3	2019-04-22	Ping An Property & Casualty Insurance	98.60%	98.40%	98.50%	[48, 12, 49]
4	2019-04-22	H&H Lab	97.93%	97.95%	97.94%	[20, 12]
5	2020-09-27	Only PAN	96.51%	96.80%	96.66%	[21]
6	2019-04-22	GREAT-OCR Shanghai University	96.62%	96.21%	96.42%	[50, 51]
7	2019-04-23	BOE_IOT_AIBD	95.95%	95.99%	95.97%	[9]
8	2019-04-23	EM_ocr	95.85%	96.08%	95.97%	N.A.
9	2019-05-10	Clova OCR	96.04%	95.79%	95.92%	N.A.
10	2019-04-21	IFLYTEK-textDet_v3	93.77%	95.89%	94.81%	[22]

Figure 35: Experimental results of TLGAN and others for SROIE task 1, 2020-10-19. (Kim et al., 2020)

The DetectGAN model (Zhao et al., 2020) was used to detect the text for camera-captured document images. The SROIE dataset was used for experimentation. This model gives a 98.98% precision value in the case of text detection. The comparison chart is as shown in the figure,

Methods	Detection results		
	Precision (%)	Recall (%)	F-score (%)
Koo's [11]	68.53	72.81	70.61
Fastext [10]	69.69	81.89	75.30
EAST [24]	89.02	92.75	90.67
RetinaNet [21]	86.78	89.06	87.91
PSENet [25]	96.70	92.40	94.34
TextBoxes++ [20]	82.34	88.79	85.44
1st rank of contest	98.64	**98.53**	98.59
DetectGAN	**98.98**	98.51	**98.74**

Best values are given in bold

Figure 36: Comparison of the proposed method in (Zhao et al., 2020)

Based on the comparative study for text block detection using the GAN method, the DetectGAN has shown outstanding results. Moreover, the faster R-CNN model gives good results in the case of small object detection. Therefore, it was decided to use the GAN based framework, which has a generator and discriminator network. The generator model will be implemented with the help of the Faster R-CNN network, and the discriminator model will be designed with PatchGAN.

3.3. Text Extraction

Once the text block is detected, the next step is to extract text from the image. Existing OCR technology is utilized for the detection and extraction process. In addition, the novel, GAN-based solution is implemented, as discussed in the previous section. The outcome of both text blocks is then fed to the text extraction system. We propose an automated document extraction system to extract all useful & frequently used data or value from any

scanned document used in any service sector for increasing paperwork efficiency, reducing human errors and the number of human interferences required. This will improve the SCM in terms of work efficiency and accuracy related to the documentation process. For a smart operation, manually classifying the variety of images present in huge quantities in real-time can be transformed to do the same task using an analytical tool that will scan the documents and classify them accordingly. Also, it will automatically extract the text mentioned in the respective documents or invoices into the internet devices and can be manipulated, thereafter as per need to build any business strategy.

Process flow of automated invoice processing

The process of extracting information from images can be easily understood by going through the process flow in the figure below.

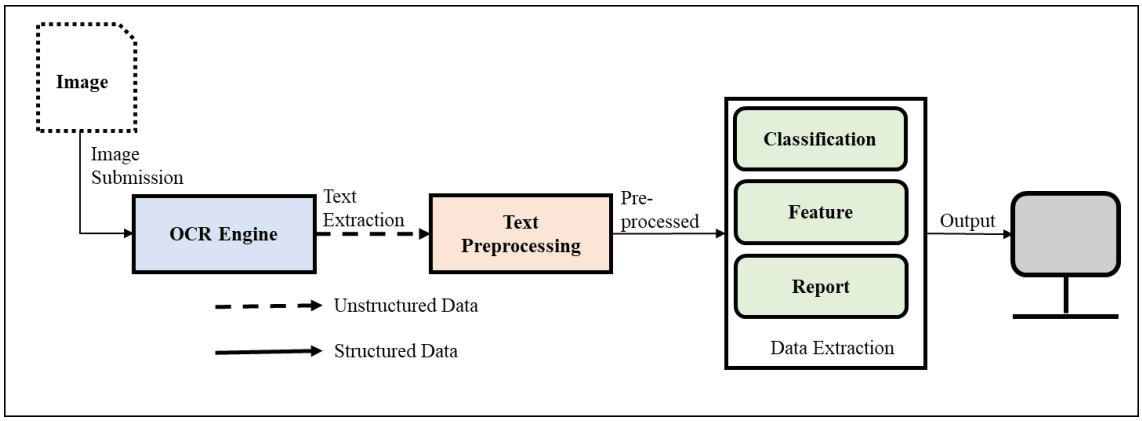


Figure 37: Image extraction process flow.
"Devised by the author".

Step 1- OCR extracted text (unstructured data) is generated by an OCR engine. OCR text is the electronic form of a printed or manually written document.

Step by step workflow in an OCR Engine:

A. The direction of text in the scanned images must be identified. This is done by aligning the lines perfectly.

B. Text dimension must be known. Example - Is it in a single column or two dimensions.

C. The position of the Baseline must be decided in every subsequent text line of each column. By this baseline allotment, the 2D problem will get reduced to 1D.

D. Generating single characters from the tokens of each line.

E. Running the program by tokens and comparing it with the unknown characters.

Step 2- Pre-processing of the unstructured data must be performed, and this generates structured data on which data extraction can be easily performed as per client or an organisation requirement.

Step 3- As per the client requirement, we can perform a classification of the data extracted, features can be generated, manipulation or any calculation can be performed, any statistical report can be generated for planning the business strategies.

Step 4- After data extraction, the required data can be displayed/viewed in any format. Example- Client or user can get the data in pdf, Email etc.

Now, after extracting the unstructured information from images, there is a need to extract information from the unstructured text extracted, illustrated in the figure above. A dictionary facilitates an additional step for unstructured data conversion into a structured format. There will be a need for a dictionary for performing any kind of information extraction, and some specific analysis also requires creating a dictionary of your own. The steps involved are discussed below:

a) Manually we cannot analyse the entire text. Hence, we use a stratified sample to build a dictionary.
b) For capturing the real essence of the text, we do data cleaning. Example- Car’s, car and cars must be counted as a single word.
c) After the text cleaning, we extract the most frequently occurred words. This leads to the formation of our dictionary.

d) Now, the entire dataset must be cleaned to make sure that the dictionary created works well on the entire dataset.
e) We can do a categorisation of each transaction statement.
f) After having tags on each transaction statement, the entire dataset can be summarised to fetch business insights and make some business strategies.

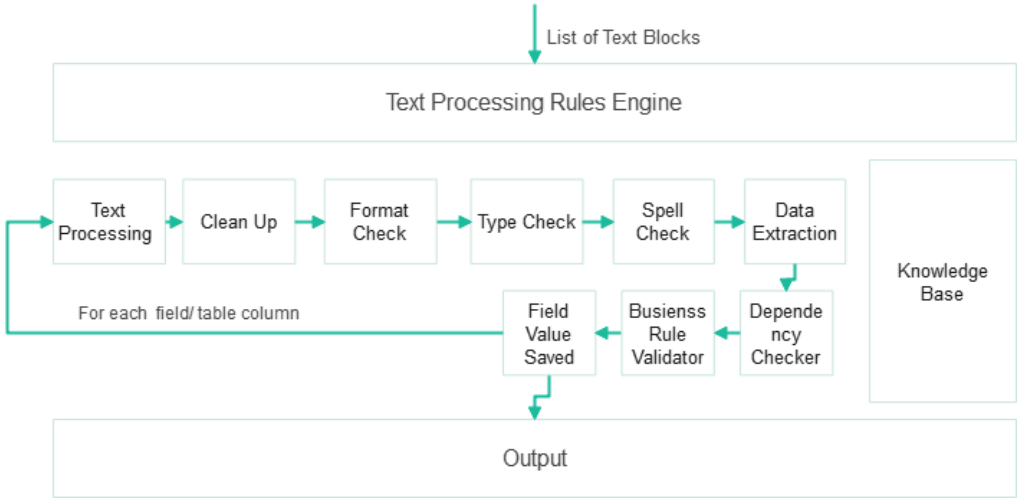


Figure 38: Text Extraction Flow.
"Devised by the author".

The above diagram explains the text extraction flow that has been designed in this thesis. It consists of text extraction pre-processing: OCR error clean-up, format checking, spell checking, and finally extracting the key-values one by one and storing the results in the system. Firstly, as explained earlier, that user gives an input image to the deep neural network, that is, to the GAN model and/or to the external OCR engine. Both the OCR models perform respective block identification and pre-processing to finalize the same structured block objects data. Based on the confidence value, blocks selection will be performed. Now next part is to perform pre-processing of extracted text data, cleaning of data. Based on the cleaning of text information, like OCR garage clean-up, format checker, type checker, finally, spell checking is performed. At last, the data is extracted for each field which gives the output in terms of text. The overall processing and

extractions are based on a unique rule-based engine. The heart of the engine is pattern matching which is thoroughly designed by collecting all the various kinds of invoice patterns from 1200 different kinds on invoices, receipts, bills, flight tickets, hotel stays, restaurants, and more. Once the data is extracted, a dependency parser checks for further validation and updates the extracted value accordingly. Finally, business rules are applied if required, and the specific field value is saved. Once all the field extraction is completed, the result is parsed and returned to the calling service.

3.4. Solution Methodology

In the solution methodology, the 2(b) *"To develop a novel method that works very well on any generic invoices."* research objective is accomplished. After a thorough analysis and detailed study, it is possible to envision and understand the proposed solution methodology. The proposed solution is based on existing machine learning technology and a unique rule-based system that is designed in such a way that maximum data can be extracted. The system requires a minor custom change for each addition of a new entity. The system does not fully depend on an external OCR engine. It uses its own machine learning-based OCR system for data block detection and text extraction. If an external OCR engine is provided, both the output is used and based on unique custom logic combined information is used for data extraction. The following diagram shows the complete invoice data extraction framework.

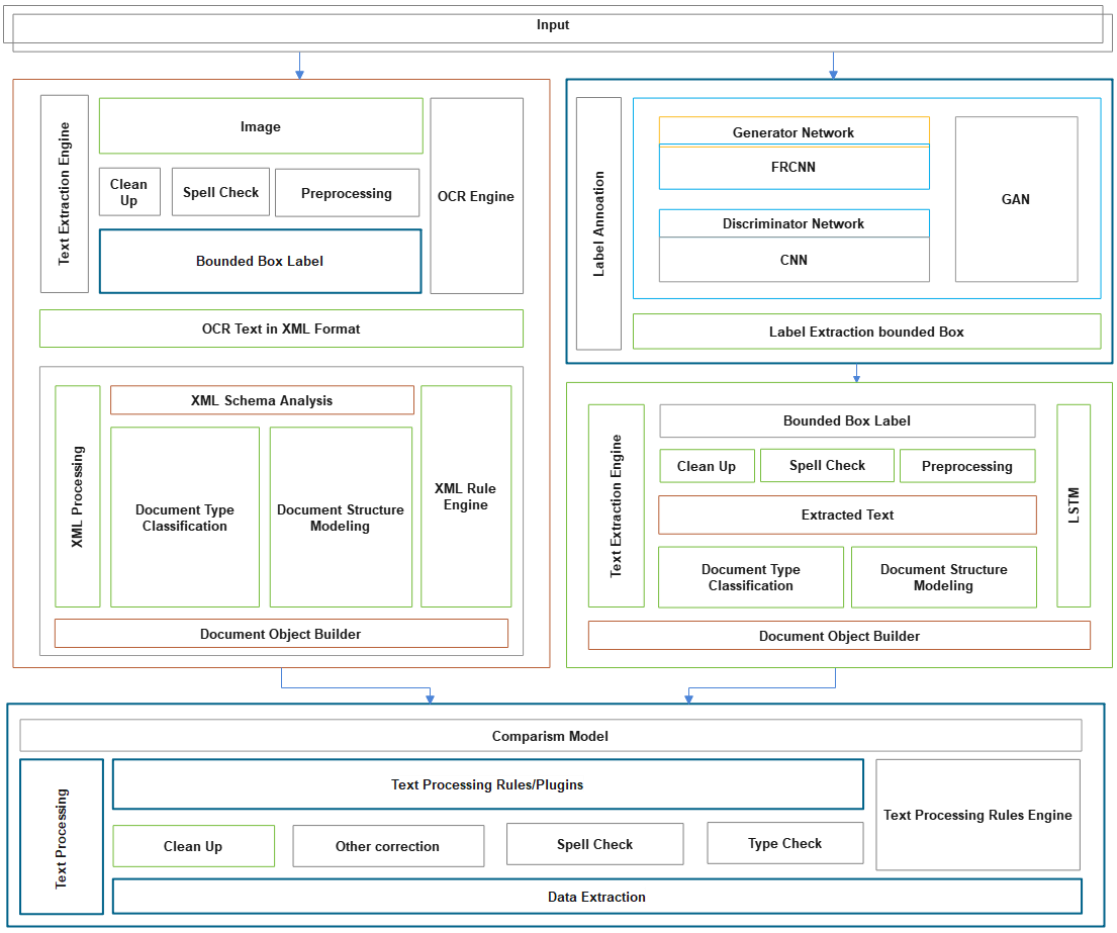


Figure 39: Invoice Data Extraction Framework.
"Devised by the author".

According to the framework, input is accepted in the form of a single document image file or a pdf file. The source file is sent to an external OCR engine, shown on the left side of the diagram. The OCR engine returns the output in the form of XML file. This OCR text in XML format is converted into a standard document object. It also performs minimal text extraction pre-processing like text clean-up and spell check so that initial document classification can be performed and document structured modelling can be defined. After this pre-processing, a structured block of text is sent for fields and table level data extraction. On the top right side, again, an OCR engine is designed to detect and extract text from blocks of text in the invoice. Here to pre-processing is performed

too similar to what was performed of external OCR engine was used. This is to ensure that the block of text is ready with proper tagging, neighbourhood relationship and other relevant properties before it is sent for actual data extraction.

Once the block of text is sent for text extraction, they are converted into lines of blocks. Using a unique formula, the blocks are arranged in a line as they exist in the invoice document. The text pre-processing does text cleaning and formatting before individual extraction can be performed. After text pre-processing, the rule engines loop in each field/entity that needs to be extracted and based on the field type, it loads respective text extraction methods. For extracting data, DataExtracor is loaded, and for extracting amount, AmountExtractor is loaded. The system moves to the next stage and performs further relationship extraction, which is based on known nearby possibilities of fields. It also performs dependency checks like net amount and total amount. Finally, the output in the form of JSON format is returned by the system.

To explain it further, let us discuss the step-by-step approach taken to perform the data extraction. The figure below represents the step-by-step process executed to extract data from OCR XML content from a technical perspective.

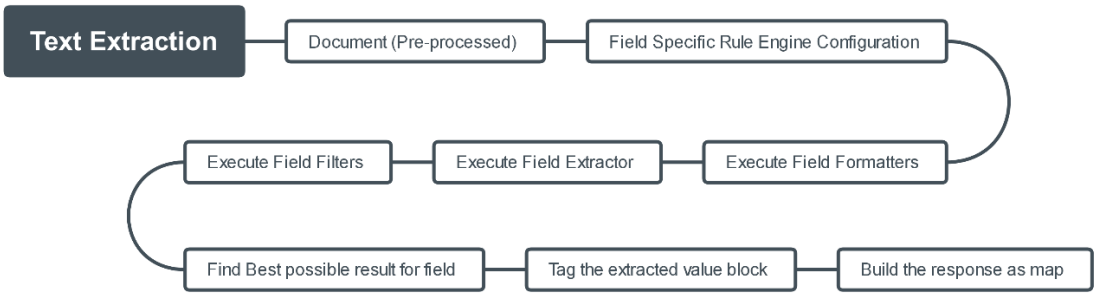


Figure 40: Step by Step Extraction Process.
"Devised by the author".

According to the above flow diagram, once the data receives a post-OCR correction to the text extraction system, it performs pre-processing setups like clean-up and format checks. Later, field-specific rules engine configuration is loaded. The engine's methods

are called field formatter, field extractor and field filter. Then best potential matches are checked against multiple-output for a single field. If finalised, the field is tagged accordingly. For example, if data is found and finalised. The blocks that contain date is tagged as a date field. Finally, a response map is built and returned as output. In the following subsection, the details of methods that have been implemented in this system have been discussed. The sections below outline a list of challenges that have been resolved, as discussed in the last chapter.

3.4.1. XML Pre-processing

The blocks of the text identified by the OCR engine is arranged in an orderly manner from left to right and top to bottom. A relationship is defined to maintain graphical relations between nearby neighbour blocks. Therefore, based on text property and the text contained in each block, this set of data blocks are arranged and maintained in a well-defined structure.

In this step, the XML file is read. All the text blocks are arranged in sequence from left to right and top to bottom to identify the relationship between blocks. The position of each block was stored in the documents object builder within the respective block. Here each block represents a single unit of textual information that was identified by the OCR engine. This block may or may not represent the combined information that was supposed to be extracted. Several text classifications have real-time applications such as information retrieval, reviews analysis and others. Managing differences between structured and unstructured data in text mining is a challenging task. Furthermore, the varied sizes of text documents and their feature extraction and accurate document classification were also complex tasks. The critical issues in XML based document classification systems were conveyed, and an improved XML based text classification model was proposed to classify the semi-structured data, i.e., XML documents, according to their subjects. A hybrid classifier was implemented using Bay's classifier and fuzzy logic to analyse and recognise the XML-based document patterns. The use of a classifier

was to organize the decisions of fuzzy-based classifier and Bays classifier in terms of weights to enhance the classification accuracy (Gurjar & Parihar, 2020).

3.4.2. Block Identification

Block identification presents the identification of objects and blocks that contains text and are required to be extracted. This framework sub-section has been designed to ensure that machine learning can be performed and dynamic template matching can be done. The system can learn based on subsequent supervised learning performed by users who validate the invoice data. In the new system, they will correct the invoices related data and mark the location in the invoice where the values were derived. Irrespective of whether the OCR is done by external service or performed using GAN based model. The block object information is stored in the database for dynamic template model generation and better data extraction. Let us understand what work is done by GAN related methods.

Due to the numerous documented successes of the GAN model, we have decided to choose a generative adversarial network to detect the text in invoice images. GAN based models are designed for image-to-image translation, image enhancement, image super-resolution, and in these tasks, they yield impressive results. For object detection, using the Faster R-CNN yields satisfactory results in the case of small object detection. So, we have finalized our model, which has a primary network as GAN in that, the generator model is designed by using the Faster R-CNN and the discriminator network with the help of the PatchGAN model.

A. Generator Network:

The working principle behind each block of the Faster R-CNN model has been explained. The model requires the labelled data. The image was fed into a convolutional network, which generated a convolutional feature map as a result. A separate network was used to predict the region proposals. Initially, the RoI pooling layer was used to predict and reshape the region proposals. This layer can then be used to classify data. (Girshick,

2015). Faster R-CNN was much faster than R-CNN and Fast- RCNN. It is used for real-time object detection. The OCR invoice classification model uses Faster R-CNN for label prediction and text recognition(Ren et al., 2016).

B. Discriminator Network

The PatchGAN (Isola et al., 2017) was the name derived from the Markovian discriminator. In low data frequencies, the existing model fails to distinguish between real and fake images. PatchGAN is a structure-at-the-scale-of-patches algorithm. The discriminator network was used to determine if each image N x N patches was real or false. The discriminator model was trained convolutionally across the picture, averaging all responses to generate the final output. When considering independence between pixels separated by more than a patch diameter, such a discriminator effectively generates the image as a Markov random field. The below diagram is part of the main framework and has been expanded for better explanation.

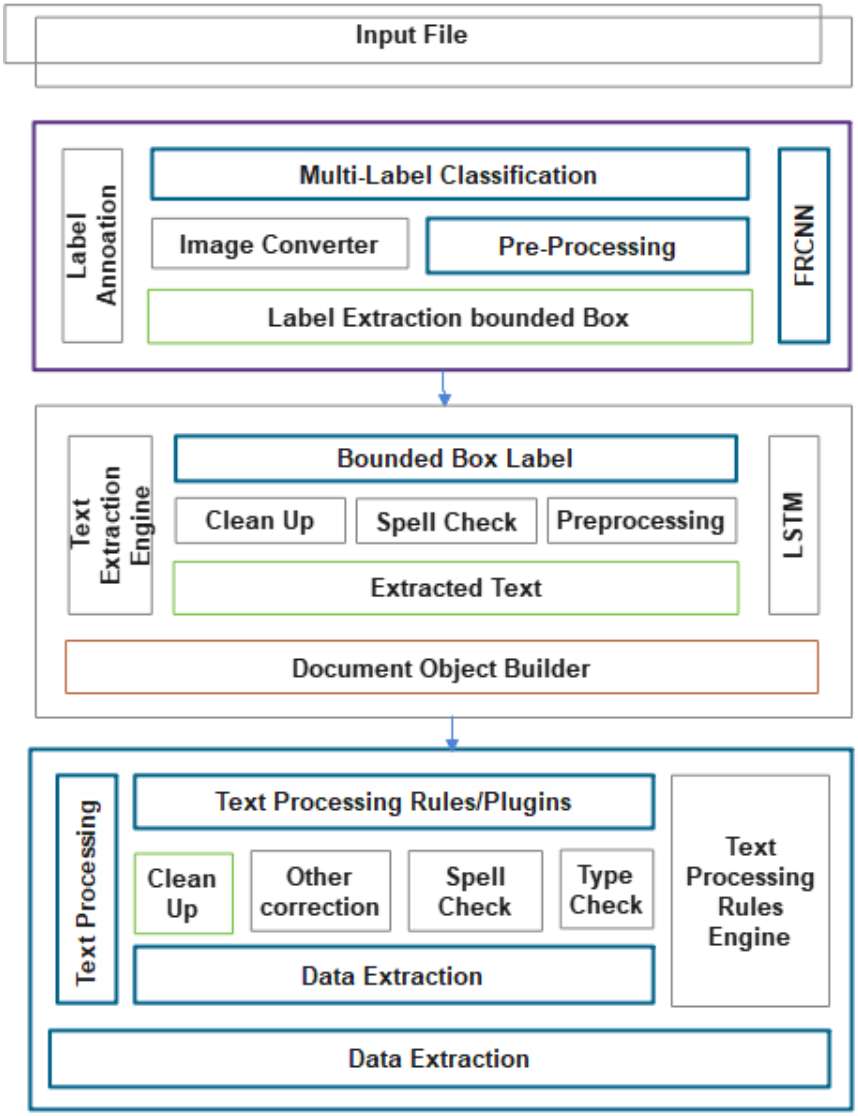


Figure 41: Final Design - Block Detection Model.

"Devised by the author".

In this solution, we have defined a new model to extract quality invoice related text. In the figure above, it is shown that the overall unstructured data processing architecture is made up of a set of modules or sub-processes. In the data block identification, the input file is passed to the FRCNN model. The FRCNN model adds annotated labelled data. Based on the annotated label, it can define the confidence score of each block that contains

the textual information. The data block is then passed through the convolution layer, and the required features are extracted. The sliding window concept is used in the region proposal network, which is applied on feature maps and coordinates of bounding box detected. With the help of the RPN model, the bounding boxes are predicted. The FRCNN model output is in terms of bounding boxes that contain textual data. So, the FRCNN model can predict the text block on the applied image with the help of saved model weights. Now next part is to extract the text from the identified block. The LSTM model is used to extract the text from bounding boxes. In the text extraction unit, we performed the pre-processing, cleaning of data, spell checking.

3.4.3. Block Position

Analysis and identification of the closest methodology to arrange the blocks in the correct sequence and then create a relationship between nearby blocks was incredibly challenging. Based on the top/bottom coordinate, value blocks were first arranged from top to bottom in a stack. Later, using a recursive function, each block was further arranged in lines of blocks if they are next to each other, i.e., left or right neighbour, by comparing the left/right block coordinates. Finally, the blocks in the straight lines were normalised by using the following function:

$$\sigma = \sqrt{\frac{1}{n}\sum_{k=0}^{n}((b-t)_i - \mu)^2}$$

$$|b,t| = \begin{cases} mid = t + \dfrac{b-t}{2} \\ b = mid + \dfrac{\sigma}{2} \\ t = mid - \dfrac{\sigma}{2} \end{cases}$$

where,

σ = standard deviation
t = top block coordinates
b = bottom block coordinates

Now, after the arrangement of blocks, a relationship map is created to connect nearby blocks as either left / right neighbour or top/bottom neighbour. This is done to extract key-value fields in two ways:

1. Right block extracted values (left-right combination)
2. Down block extracted values (top bottom)

Right block extracted values have priority. If the expected value is found on the right side, it is considered; otherwise, bottom blocks are checked using the earlier relationship map. If both categories were empty, we fetch the first value from the extracted list of values and make this the final value for a given key field.

3.4.4. Loading Key Data

We first read all the non-tabular fields from config DB and start configuring the rule engine for them. In Rule engine configuration, we are supposed to tell the following info for a field.

1. Field dataType (e.g., Decimal, String)
2. Field Extractor Type (e.g., Regex, Classifier, Table)
3. Field-specific formatted (e.g., Date Formatter, Format Amount near vicinity Formatter)
4. Field-specific Filters (e.g., Strict Vicinity Match Filter, Invoice Date Validate Filter)

Once this is derived from the database, the dynamic key property is loaded in the memory, and the document object, which consists of blocks of text, is fed for data extraction. The significance of this methodology is that it helps to add any new fields in the system, and at run-time, any new field value gets extracted.

3.4.5. Format Check

The format check is executed based on the sequence of format read from the database. Formatters are used to correct the value pattern or convert all different formats of a particular value into a single format (to make them homogeneous). The final corrected format is used finally at the time of extraction. All formatters and filters are defined as a plugin; as such, they can easily be plugged or unplugged from database configuration. Format check included regular expression-based checks like data, numeric or text. If a data value 10/05/2020 is read as i0/05/202o then the format checker will detect it as a date type and try to correct the date format.

3.4.6. OCR Error

In this, all standard error created by OCR due to various issue from image quality to OCR accuracy, is taken care. For example, the letter ‘m’ might be read as ‘iii’. In this, direct replacement of incorrect text is done with the correct text.

Different information or field that needs to be extracted varies by data type and format. For example, date fields vary in different formats across different countries and need to be corrected in line with a predefined, generic format. The numeric value needs to be corrected, e.g., ‘1,33 .5 6’ to ‘133.56’. OCR errors are non-spelling related errors and are considered based on their regular occurrence in a specific format all the time.

3.4.7. Spell Check

Levenshtein Distance is used to calculate the distance between strings (Lhoussain et al., 2015). They proposed this approach for the Arabic language spell checking. The edit distance was the smallest number of simple editing operations needed to change a misspelt word into a dictionary word. As a result, to correct a misspelt phrase, one must keep a collection of solutions in mind. The Levenshtein Distance was a metric that calculates how many edits are needed to convert string 1 to string 2 (Yulianto et al., 2018). For instance, the distance between 'pride' and 'price' was 1 (here replaced character 'd’

with 'c'), and the distance between 'invici' and 'invoice' is 2 (replaced character 'i' with 'e' and inserted 'o'). As a result, the Levenshtein Distance tests the similarity between two strings, where the source string (s) and the target string (t) as the source and target strings, respectively.

There are fewer editing operations that can be performed.

For example,

- If s is ‘invoice’ and t is ‘invoice’, then edit Distance(s,t) = 0, because no transformations are needed as strings are already identical.
- If s is ‘irvoice’ and t is ‘invoice’, then edit Distance(s,t) = 1, because one substitution (change ‘r’ to ‘n’) is sufficient to transform s into t.

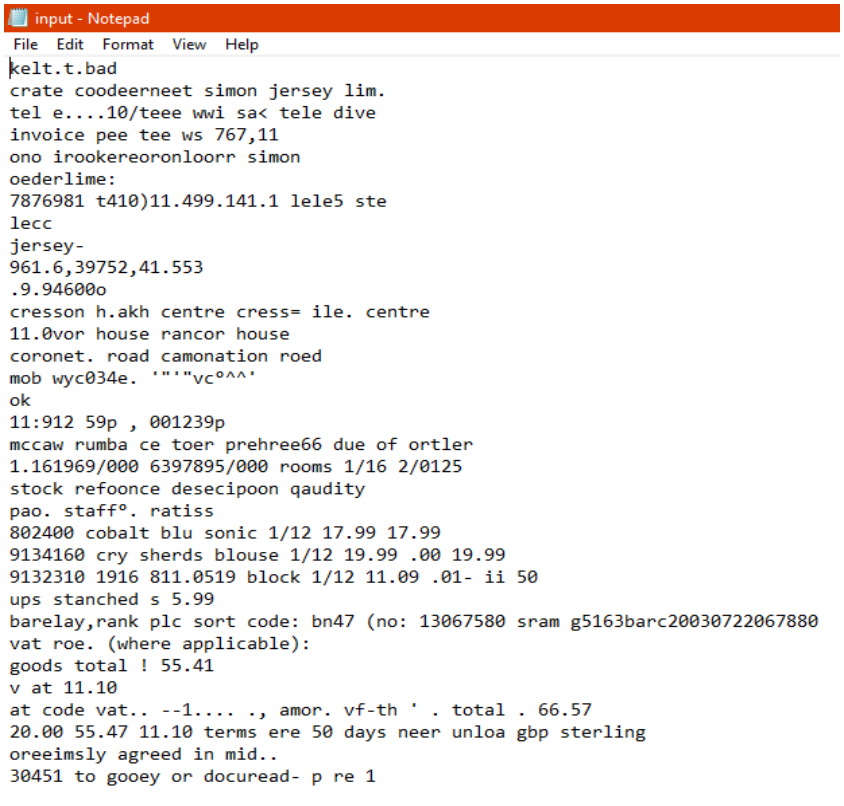


Figure 42: Sample OCR text file with errors.

"Devised by the author".

After the OCR is performed, the text returned by the OCR engines contains many errors. The figure above is the sample OCR text file with errors, on which we must do the spell check. To do the spell check, we used the Levenshtein Distance formula. It will check the spell and correct it with the help of a dictionary by updating the distance between two words. The sample OCR error-corrected file is shown in the figure. In the figure below, the word "toer" is an incorrect word corrected as "toner", similarly "refoonce" as "reference", and so on.

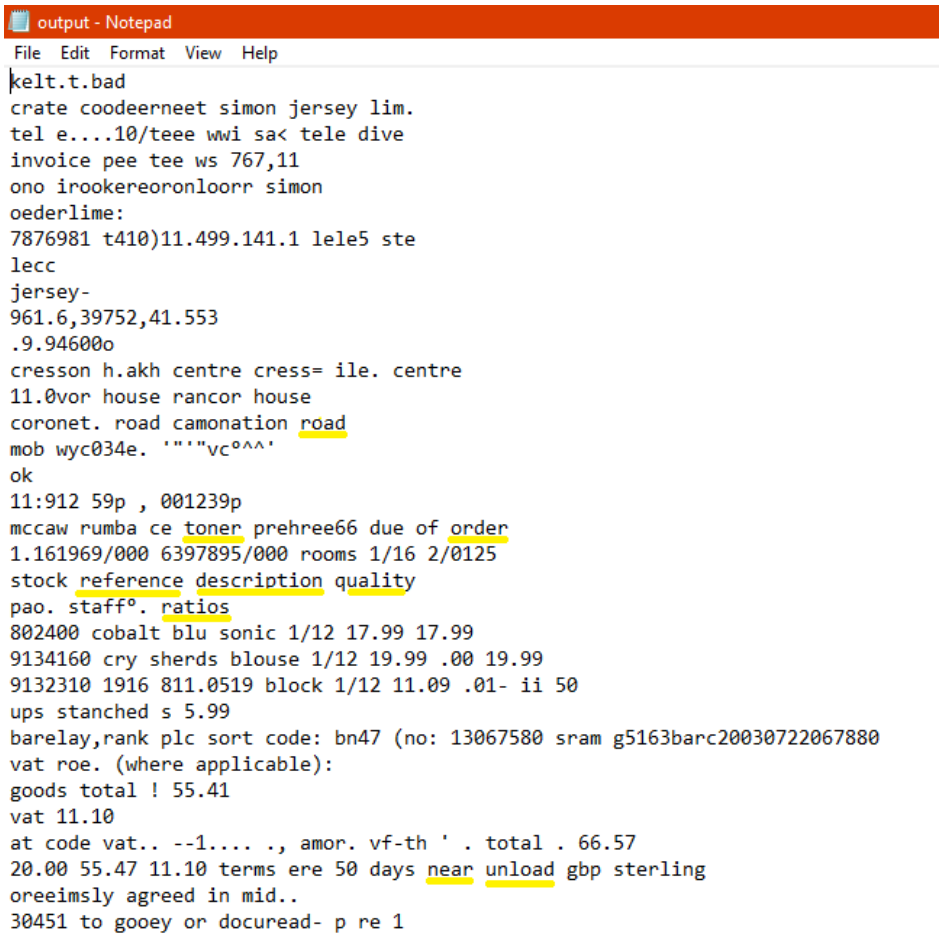


Figure 43: Sample OCR text file showing correct words.

"Devised by the author".

Therefore, we see that the character in an OCR text file is not read correctly and contains too many errors. This error needs to be corrected before any information is extracted. For this, the spell-checking process is applied. This process tries to correct the error based on a set of already defined dictionary lists. This dictionary contains a list of frequently used words and business-specific words. Using the Levenshtein distance (LD) formula for character sequence matching, continuous character separated by space is spell-checked with the predefined list of words. Based on the defined threshold, replaced words are decided. The figure below represents the CSV input file where uncorrected words and corrected word distance are calculated with Levenshtein Distance's help. Moreover, the distance between two words is defined in the third column.

	A	B	C
1	input_word	corrected_word	levenshtein_distance
2	askin	asin	1
3	bacs	back	1
4	bakewel	bakewell	2
5	#NAME?	NULL	2
6	cheaues	cheques	2
7	filo	file	1
8	vveles	NULL	1
9	gakden	garden	2
10	goocfs	goods	2
11	goole	google	1
12	gwent	went	1
13	deseripuon	NULL	2
14	indo	info	1
15	jilint	filing	2
16	korrecot	correct	2
17	teleetione	NULL	2

	A	B
1	input_word	output_word
2	askin	asking
3	bacs	back
4	bakewel	bakewell
5	#NAME?	name
6	cheaues	cheques
7	filo	file
8	vveles	vessels
9	gakden	garden
10	goocfs	goods
11	goole	google
12	gwent	went
13	deseripuon	description
14	indo	info
15	jilint	joint
16	korrecot	correct
17	teleetione	telephone
18	neer	never

Figure 44: Levenshtein Distance input-output comparison Fig A with edit distance and Fig B with the correct value. “Devised by the author”.

Above, figure B shows that the uncorrected input word is corrected with the help of Levenshtein Distance; that is, it acts like a spell checker here. And with the help of a

dictionary look, it will correct the word or simply removes the incorrect character from the string. For example, the incorrect word “teleetione” is corrected as “telephone”.

3.4.8. Key-Value Based

This includes a list of values that were identified using the name of the field that needed extraction. In this case, the key was first searched, and then the value was fetched either right of the key or bottom of the key. For example, ‘Date’, ‘Total’, ‘Discount’ etc.

3.4.9. Classification Based

This includes information extraction, which requires counting vicinity words to classify the document or field under a specific category. In this, a key parameter does not exist. For example, currency and document type and only the value exists.

3.4.10. Vicinity Words

Vicinities are the keywords used to identify the value of a field, whereas **regex** is written to validate the pattern of that value. A combination of both is used to extract the result from the text of a line. There can be multiple vicinities for a single field; therefore, weighting to each vicinity is configured based on the analysis to give priority to that vicinity while doing value finalisation and extraction. Vicinity weight is ranged between 0.0 to 1.0, Where 0.0 resembles negative vicinity.

To add more, **negative vicinity** refers to the exclusion keyword(s). It means that specific keywords will *not* be used to extract the values of that field. If the field was extracted using negative vicinity keywords, it would be discarded.

E.g., let us assume a line has text as “*sub total 123.45”.*
While extracting the values for total, it they be extracted as 123.45 for vicinity ‘total’.

But as soon as we see a vicinity as ‘sub total’, which is negative vicinity for total, it will be discarded.

3.4.11. Regular Expression

To write regex, one first needs to identify several patterns for that value. Then it will try to combine all patterns in a single regex. If not possible, multiple regexes can be written for the same field, which gets executed one by one as regex is loaded as a list of values. To combine regex with vicinity at run time, a string phase in regex something like (**0)** gets replaced by actual vicinity before regex is applied to the text.

e.g., regex for Telephone:

(?<=([.:,;](0,1))((0))[.:,;!\(\{](1,5))([oc!l\d|\-|/| \)\(\)](5,15))(?=\s)

After replacement, it will be:

(?<=([.:,;](0,1))(Telephone)[.:,;!\(\(]{1,5))([oc!l\d|\-|/| \)\(\}](5,15))(?=\s)

To understand more, according to (de Jager & Nel, 2019), human testers are responsible for manually identifying and capturing the right strings, which are often difficult to decipher due to the lack of formal definitions of fields. Regular expressions were used to identify patterns and filter out strings of fundamental structures that the regular expression described. Each candidate value for a field key that determined a regular expression was added to a pool of potential candidates for the key. Regular expressions are used to extract individual values from larger strings and identify strings that are not labelled. Regular expressions can be used to identify labels for tax values. Many of the tax values of corporations have a rigid structure as specified by the governing bodies, and regular expressions can be used to identify labels for tax values.

3.4.12. Data Extraction

This section is the core of the data extractor. Each field is supposed to be extracted by one extractor function at a time. Based on the type of key, one of the extractors is loaded in the memory, and then data extraction is performed. Currently, three distinct kinds of extractors are implemented:

1. **Regex Extractor:** To extract value like key-value pattern. In few cases, values are also extracted without vicinity (e.g., doc_date and total). As sometimes, there

is no key printed in the invoice bills. Regex is a complex pattern-based rule that looks for specific patterns in the text, such as the date and extracting the value.

2. **Classifier Extractor:** To extract value based on keyword weighting and count. Supported fields: doc_type, country, currency etc. For example, whether an invoice belongs to the country "USA", a similar list of words is looked at such as the US, USA, America etc.
3. **Table Extractor:** To extract values in of row-column pattern. The item list and tax summary tables are field types that further break down into inner columns and fields. This extractor is called separately with different column tables passed as arguments-supported fields: line item and tax summary.

3.4.13. Filtering

Filters involved majorly two tasks:

- Narrowing down results (by checking validity of results based on business assumptions) e.g.: StrictVicintyMatchFilter, InvoiceDateValidFilter and ExtractRelevantAmountFilter. Multiple values might have been extracted, and the final value is filtered and selected.
- Change the order/format of resultant value (e.g., Sort, Refining, case conversion, capitalizing etc.) e.g.: SortExtractedValueFilter, NumericSupportConversionFilter, TextCapitalizationFilter

3.4.14. Dependency Mapping

In this process, if specific fields are not extracted or found, the system tries to execute a dependency match between fields to check the possible default value based on some other field. For example., if the country is extracted, then default currency can be known or extracted. Again, if the postcode is known, then the country can set to default accordingly.

The field dependency matching is maintained in the data-based, and only for those whose final extracted is empty are considered.

3.4.15. Final Field Tagging

To protect a block from getting fetched multiple times, we need to tag them once an extracted value for a field is finalized. Furthermore, there are ambiguous names for fields such as ‘total’ is an invoice level field, and it is part of the line-item table and tax table too. Therefore, the pattern for tagging has been defined as:

[parent field name]:[child field name].

So, for ‘total’ it will look like

1. field:total (for non-tabular field extraction)
2. item:total (for tabular line-item extraction)
3. tax:total (for tabular tax item extraction)

3.4.16. Response Object

Finally, in the process of information extraction for each field, the last step is to parse extracted data to its corresponding data type (e.g., doc_date as Date, net_amount as BigDecimal, doc_id as String etc). After this, the values are appended in the response map with the field name as key and parsed value as the map's value.

3.5. Performance Evaluation

In this section, 2(c) *“To identify the role played by automatic data extraction on SCM optimisation.”* and 2(d) *“To validate the optimization in accuracy, cost, and speed at which the model works.”* research objectives are successfully achieved. After the complete execution of the accuracy test, the system is sent to the data validation team for correction and validation of extracted data. In a web portal, a human manually compares the image with the extracted data and updates the reason for incorrect extraction. In case of no appropriate reason found in the drop-down as per observation made, ‘Incorrect’

reason is selected, and later further specific reasons are added by the development team if needed.

In the performance evaluation, we have analysed the proposed method with the existing methods. For the evaluation, we used three distinct types of datasets, such as the real-time company dataset and two academic datasets, such as SROIE and VATI. The experimentation results are tabulated in the below table for different invoice/receipt datasets.

Table 17: Result: SROIE dataset, Our method.
"Devised by the author".

Method	**Precision (%)**	**Recall (%)**	**F1 score (%)**
Koo's (Koo, 2016)	68.53	72.81	70.61
Fastext (Busta et al., 2015)	69.69	81.89	75.30
EAST (Zhou et al., 2017a)	87.65	79.5	83.37
SegLink (Shi et al., 2017)	89.02	92.75	90.67
PSENet (Li et al., 2018)	96.70	92.40	94.34
W-A net (Anand & Khan, 2020)	83.60	86.10	84.83
DetectGAN (Zhao et al., 2020)	98.98	98.51	98.74
Our Method	**98.85**	**98.65**	**98.75**

We tested our method on the SROIE dataset. The results are tabulated in the table above. The different performance measures are calculated, such as precision, recall and F1 score. All the results are presented as percentages. We compared them with the previous method and can conclude that our method yields better results based on this analysis. As the F1 score value is more than that of the existing research method, with a value of 98.75%.

The mathematical formulas are shown below:

Precision = TP / (TP + FP)

Recall = TP / (TP + FN)

$$F1 = 2 * \left(\frac{Precision * Recall}{Precision + Recall}\right)$$

Where, TP = True Positive, FP = False Positive and FN = False Negative

Our method's values with precision are 98.85%, recall 98.65, and at last F1 score value as 98.75%.

Table 18: Result: VATI dataset, Our method.
"Devised by the author".

Entities	**Chargrid**	**NER**	**GCN**	**TRIE**	**Our Method**
Code	89.4	94.5	97.0	98.2	**98.6**
Number	85.3	92.4	93.7	95.4	**95.8**
Date	89.8	82.5	93.0	94.9	**95.6**
Pick-up time	82.9	60.0	86.3	84.6	**84.7**
Drop-off time	87.4	81.1	91.0	93.6	**95.4**
Price	93.0	94.5	93.6	94.9	**95.2**
Distance	92.7	93.6	91.4	94.4	92.2
Waiting	89.2	85.4	91.0	92.4	**93.1**
Amount	80.2	86.3	88.7	90.9	**93.7**
Average	**87.77**	**85.59**	**91.74**	**93.26**	**93.81**

In addition, we also tested our method on the VATI dataset. The results are tabulated in the table above. The different named entities are recognized. Our method gives better results in terms of F1 Score than the TRIE method. We only got less value for a single field which is "Distance". We have compared the previous method, and based on the analysis, we conclude that our method gives better results. Our results have outperformed the TRIE method as the average accuracy of our method is 93.81%.

Finally, for validating the result in company data, we are using a total of 3300 real-time invoice/receipts images. From the whole data, 70% are used for the training and 30% for validation. After the execution of approx. 3300 records and updating of pass/failure reason for each extracted field, following results are achieved. Here, the left column represents single item field names and to its right is the accuracy percentage of the baseline model, which is decided by company A and right to the baseline is our method results with sub-columns such as accuracy and recall precision and F1 score. All the data is tabulated below table as,

Table 19: Result: Company dataset, field-level extraction - Our method.
"Devised by the author".

Field Name	Baseline	Our Method Results			
	Accuracy	Accuracy	Recall	Precision	F1
Invoice No	71.00	87.58	85.35	95.19	90.00
Document Type	75.00	92.44	93.47	98.78	96.05
Date	80.00	91.92	92.92	98.67	95.71
Vat No	80.00	91.33	84.34	96.83	90.15
Currency	75.00	92.78	93.74	98.84	96.22
Discount	75.00	99.91	96.67	99.40	98.02
Carriage	70.00	99.94	98.97	99.82	99.39
Tax Rate	75.00	99.47	93.26	98.74	95.92
Tax Amount	75.00	89.56	83.43	96.61	89.54
Net	75.00	83.90	78.86	95.48	86.38
Total	85.00	90.14	91.26	98.34	94.67
Tele	80.00	90.96	83.12	96.54	89.33
Fax	50.00	97.98	89.57	97.99	93.59
Website	90.00	97.54	94.97	99.07	96.98
Email	50.00	99.32	95.98	99.27	97.60
Bank Account	50.00	93.18	81.27	96.09	88.06
Due Date	50.00	99.66	96.19	99.31	97.73
Zip Code	50.00	99.97	99.94	99.99	99.96
Country	70.00	99.72	99.76	99.96	99.86
Average	**69.5**	**94.59**	**91.21**	**98.15**	**94.48**

This new system has again outperformed the results with the baseline model. The baseline model of company "A" has an average accuracy of 69.5%, and our proposed method has 94.59% for the single item field names extraction. The F1 score value of our method is 94.48%. Overall, all the results are tabulated in the table above for the company dataset.

Table 20: Result: Company dataset, table level extraction - Our method.
"Devised by the author".

Field Name	Baseline	Our Method Results			
	Accuracy	Accuracy	Recall	Precision	F1
○ **Item Table**					
Product Code	70.00	88.68	92.02	94.53	93.26

Description	85.00	88.53	91.91	94.46	93.17
Quantity	80.00	88.66	92.0	94.52	93.24
Unit Price	75.00	88.60	91.97	94.50	93.22
Net Amount	85.00	88.42	91.85	94.41	93.11
Tax Amount	85.00	88.65	91.97	94.50	93.22
Tax Rate	85.00	88.70	92.01	94.53	93.25
Tax code	85.00	88.68	92.02	94.53	93.26
Total	90.00	88.58	91.95	94.48	93.2
Discount	90.00	88.70	92.01	94.53	93.25
Average	**83**.00	**88.62**	**93.69**	**93.22**	**93.22**
o **Tax Table**					
Net Amount	60.00	79.27	87.67	91.43	89.51
Tax Amount	60.00	79.43	87.75	91.49	89.58
Tax Rate	60.00	79.43	87.75	91.49	89.58
Average	**60**.00	**79.38**	**87.72**	**91.47**	**89.57**

All of the invoices and receipts have different layout structures. The layout may contain the text information in table format. Even still, each table format might represent data in various styles with different font sizes. This makes it challenging to achieve sufficient accuracy through field level extraction (Gilani et al., 2017; Liu et al., 2020). Detection of the table in the original size image and investigate the overlap of columns to get a better segmentation result (Reza et al., 2019). This was most challenging due to the complexity of table structure, dynamic columns, and printing in between columns. The proposed method has outperformed the results for the field level extraction method with the baseline model. The baseline model for company “A” has an average accuracy for the item table of 83% and a tax table accuracy of 60%. The proposed method has accuracy for the item table of 88.62% and a tax table accuracy of 79.38%. The F1 score value of my method for the item table and tax tables are 93.22% and 89.57%, respectively. The overall results are tabulated in the table above for the company dataset.

In addition, cost optimization analysis has been performed. Before the invention of automated software and tools, people manually added the information required for cost

optimization analysis. The study showed that processing 500 invoices manually takes 40 hours. Later, with the existing text extractor tools/ software, an automated system reduces the effort cost by 25 hours. Still, the optimisation cost was high according to the business, and they were expecting further optimization. Now the proposed method further reduces the time to 20 hours which means another 20% reduced the cost of optimization achieved with respect to existing tools/ software. Also, the time taken by this newly proposed system to automatically extract text is around 3-6 seconds based on quality, size, and amount of data in an invoice/receipt image.

3.6. Contribution

This chapter provides the solution in terms of various research gaps identified in chapter 2, section 2.5 related to text block detection and text extraction such as image quality issue is resolved with the help of the GAN model, block position identification using Faster CNN model, and the LSTM used to extract them from identified bounding boxes. A detailed explanation is provided in the previous sections, such as 3.2, 3.3 and in 3.4 of this chapter. The performance of the implemented model is compared with other techniques, which shows our solution methodology is better than others, as mentioned in section 3.5. In this chapter, the following research objective was fulfilled:

2(a) To understand and apply various machine learning and pattern recognition techniques to increase accuracy.

2(b) To develop a novel method that works very well on any generic invoices.

2(d) To validate the optimization in accuracy, cost, and speed at which the model works.

3.7. Need for Further Enhancement

In the overall execution of this process, there were many limitations and problems identified. Some of them have been discussed below:

1. **Data block combining problem:** We combine word blocks by nearby spaces to form a composite data block. When tagging of these data blocks with field names was done, data was successfully extracted. So, those same blocks are not read again in case of a similar match. There is a possibility of two or more field data that has been combined as a single data block. The value will not get extracted as expected because tagging of the field will cause a problem here. If two-word blocks are far away in the image, we must consider those spaces while making them a line. Due to printing issues, some end up getting combined. So, the correct value extraction of other fields was missing. The logic was unique but failed in many instances. Further work needs to be done.
2. **Data Extraction problem:** In the case of digit related field, let say the amount, we consider formatting the amount considering the possibility of space in-between. However, then sometimes another field value, such as 'vat 67892 45 453', is also considered as the amount. This requires improvement. Currently, resolving this type of issue leads to the failure of other numeric fields separated by spaces. e.g., account_no was being read total amount. Furthermore, if in-between header row and value rows, few other lines have been added, the top-bottom mapping will not be possible. Thus, predefined template matching will help for better tabular data extraction.
3. **Problem with negative vicinity:** Scenario: Negative vicinity word is not close to the actual vicinity word (or part of some other block of text) but was in the same line, it got identified, and hence value has been discarded from actual extraction. e.g.: vat goods/servs vat goods/servs vat sub total-. 16.59. Here we can easily see that ‘net amount’ could have to get extracted with subtotal-. 16.59 it gets extracted in the first place, but at the time of negative vicinity, identification gets discarded. The reason

being vat is a negative vicinity for field net amount, and it was found here in the line text.

4. **Block match priority:** Value fetched from the right block always have high priority than the value from the down block. The priority order is to finalize from multiple extracted values: highest weighted vicinity > vicinity length > index in text.
5. **Word/Spell Error:** OCR is efficient, but it typically contains some errors. These errors require the post-processing effort of proofreading the electronic form of text. This task can be tedious and dull as a large amount of text is generated in electronic form but is demanding too. When a human read anything, he uses a broad spectrum of his knowledge to make sense of linguistics. When provided with an optically digitized image, the machine is prone to errors in the absence of such knowledge. The goal of OCR is to transform a document image into character-coded text. The OCR transformation process uses image analytics heuristics, which classifies the character using the character code corresponding to the image's character. In the process of automation of text extraction, this remains a big challenge. The OCR may be efficient in converting images to text where the images were scanned correctly and had a higher resolution. However, we cannot specify the user to provide higher resolution images every time. We need the spell checker to check for non-word errors, which can be a substitution, deletion, or insertion errors. Although the OCR has a built-in spell checker, the user needs to correct all the errors identified by the spell checker manually. Hence, to avoid the task of proofreading or manual correction of one file at a time, we need to work on this section and propose an automatic correction of errors for better data accuracy.
6. Multiple labels on the image are predicted successfully with a moderate confidence score. The text on the scanned invoice is detected and extracted successfully. Coordinates of the annotation boxes are also stored with the output. It has been observed that the text extraction for the label “address” had comparatively better accuracy than other labels such as “date” and “amount”. As the date had no fixed

location and format on the invoices, it was difficult to detect and label it appropriately. The accuracy was lower due to an insufficient amount of training dataset. The accuracy of prediction will increase by using a more extensive and more varied invoice dataset. Alternate machine learning algorithms can be explored to improve the overall accuracy of the systems.

3.8. Conclusion

This chapter discussed text extraction in the form of information needed as per business requirements. It lists down information in the form of single values and tabular items that are needed to be extracted. The list includes almost everything contained by an invoice. Next, the pre-processing phase discusses XML analysis for data block arrangement, spell check, OCR error correction (which cannot be done in spell check) and formatting of certain specific fields such as date and amount. Furthermore, it discusses a list of problems and challenges which exist in the existing system and/or are faced by the current user.

Enhancements can be made to the system in terms of user-friendliness. This can be achieved with the help of additional tools such as app development or running the project in an interactive environment such as a website. The ratio of the training dataset to the testing dataset is a crucial factor in determining the model's accuracy. The model has been trained using a limited number of images. The prediction accuracy will improve if a more prominent image dataset is used for training. The scope for improvement in error correction is limited because the existing OCR system does not learn from past mistakes when extracting text. To avoid repetition of the same errors and improve the accuracy of text extraction, a machine learning model was proposed in parallel to the existing OCR. This model for text extraction will detect and extract text from scanned images and learn from any errors in the extracted output. This will lead improved error correction cycle. Thus, using machine learning will be able to extract the text correctly in the future.

Furthermore, the machine learning-based Faster R-CNN model (Ren et al., 2016) was implemented for object detection purposes. The Faster R-CNN gives the bounding box over a labelled text. For labelling a required text data or the object, LabelImg is used, which gives the output in terms of the XML file. In the case of an invoice, scanned receipts, the various fields such as “Date”, “address”, “time”, “total”, etc., are marked. The drawback of the Faster R-CNN model, the complexity of the model, is high, requires a more significant number of parameters for learning, fails in case of low resolution and super-resolution of the images. GAN's success stories in the fields of image super-resolution, image to image conversion, and image animation have shown highly long-lasting results. The GAN model has two networks such as generator and discriminator. The purpose of the generator is to generate realistic images with the help of noises such as adversarial loss, L1 loss, and MSE. The discriminator distinguishes between the two images, such as the ground truth image and the generator output image. Based on this, it gives the decision whether it is real or fake image data. The GANs are trained with the help of the min-max game equation.

Finally, we tested our method on the three datasets such as real-time company, SROIE and VATI. For the SROIE dataset, different performance measures are considered for our model. We compared it with the existing model, and the results are tabulated in terms of precision, recall and F1 score. Our results are best than that of an existing method. The value of the F1 score for the SROIE dataset is 98.75%. For the VATI dataset, the named entity extraction was performed, and the comparison values are tabulated.

Similarly, the results outperformed in terms of accuracy over the TRIE method. The average accuracy of our method was 93.81%. We conclude that our method is better than that of existing models. Furthermore, the company baseline model has an average accuracy of 69.5%, and our proposed method has 94.59% for the single item field names extraction. The proposed method has accuracy for the item table as 88.62% and tax table as 79.38%. The F1 score value of this proposed system for the item table was 93.22%,

and the tax table was 89.57%. The company with positive notes accepted the overall performance analysis conducted on a real-time company dataset.

The proposed method also reduces the cost of optimization. It takes 40 hours to manually extract the text from 500 invoices. The existing automated system takes 25 hours. The proposed method takes only 20 hours. Also, the time taken by the proposed system to extract text automatically is around 3-6 seconds based on invoice/ receipt image quality, size, and amount of data.

4. Non-Word Error Correction

4.1. Introduction

In this chapter of the thesis, we have sought to understand and analyse the machine learning approach for invoice data extraction. We have tried to identify the need to enhance certain sections of this expert system by using machine learning techniques. We have understood the approach for automated OCR recognition and data extraction from a scanned image. We analysed the existing system used in the current industry and research work. We have also identified a further lack in the research and the need to enhance certain sections of the invoice data expert system using more machine learning techniques that should be modelled and applied appropriately. One of the techniques for further enhancement of accuracy is to study the application of spell checking on words and characters which are misspelt due to OCR performance in text extraction. These issues could be in the form of non-dictionary-based errors, which even a standard dictionary algorithm cannot correct, or a dictionary-based error too, which after correction, converts the word to some other word, therefore changing the semantic meaning too. Currently, the application of the spell-checking process is successfully used to a significant extent in sentence correction for books and news archives based on scanned documents where there are syntax and semantics present in every sentence and paragraph. However, it is incredibly challenging to correct errors in invoice data extraction due to the unstructured flow of text and lack of ‘complete sentence’ (a very well fact followed in any spoken language, for example, English).

As such, there is a further need to understand how spell checking can be applied in this area of research. This chapter will describe the spell-checking process in greater detail. After critically evaluating different spell-checking methods, we will finalize a model and apply it to increase the accuracy of invoice data extraction. This will involve a comparative study of different research work done related to solving post-OCR errors in text extraction. Finally, we will use the methods on the same dataset used in earlier

chapters to analyse and evaluate the efficiency. We will present our novel application of the machine learning method. We will also try to present the importance of spellchecking in OCR, when and how that should be applied. For this, we will walk through the details of OCR error types, post OCR word correction solution and research done in this area in the form of a literature review. Finally, we will run an experiment and evaluate the efficiency of the solution. Let us start now and understand the process of spell checking in detail. We will try to enhance data extraction accuracy by understanding the incorrect word/character that is misread while performing OCR on scanned images.

In this chapter, we focus on the non-word error generated in the OCR data extraction process. When we say spell checking process, we do not imply comparing with dictionary-based words. Instead, these errors are more specifically due to incorrect character extraction done by the OCR engine. The correctness of the text is solely dependent on the resolution and quality of the image. Since we have no restrictions on the quality of the input image, the OCR text conversion results frequently contain errors. Moreover, this significantly increases the distorted images.

Furthermore, to add more, independent research is being carried out for image quality enhancement for text correction even before the OCR process is started. However, in our research, we are only focusing on post OCR error correction. Therefore, we will discuss the errors introduced in text extraction, why these errors occurred and why there is a need for spell check? What was the motivation to create a spell checker for OCR, and what are the objectives?

To understand in a simple way, when the OCR engine extracts text from a given image, all the extracted characters, numbers, or symbols are not read correctly (Bassil & Alwani, 2012). Let us say,

- character '*m*' gets recognized as characters '*rn*',
- character '*l*' as number '*1' (one)*,
- character '*S*' as number '*5*',

- character '*O*' into number '*0*', and so forth. (Bassil, Youssef, and Mohammad Alwani, 2012)

(Bassil & Alwani, 2012) found a solution to the above problem that uses human intervention to manually review and correct the OCR output text by proofreading the original image and what was extracted by OCR. This is a time-consuming process that leads to considerable human error. It is also incredibly costly because it relies on human effort and resources.

Apart from automating a section of the finance supply chain process to save and optimize operational time, another objective is to reduce further the cost incurred in the human verification of documents. Moreover, to overcome or reduce this proofreading process, we designed a spell checker that automatically checks for errors and correct them as and when similar errors are recognized. This helps in correcting the desired information and increases the accuracy due to the correction of key fields in the invoice document. For example, even if the date was mentioned correctly, the key 'Date:', before the value '12-03-2018', was read as 'Dale:'. Now when we correct this word, key-value pair extraction accuracy also increases. Thus, this further reduces the time required to validate the final output.

According to (Kumar, 2016), most spell checkers which detect and correct errors worked on word level and used a dictionary. A dictionary consists of words that were assumed to be correct for a given language. The words from the text were checked from these dictionaries. When the word checked was not found in the dictionary, it was flagged as an error word. The words were then suggested to the user(s) based on the several techniques and algorithms used in the spell checker so that the new words can be replaced in place of the error word. Spelling detection and spelling correction is the most common issue with spell checkers, as they work differently in different algorithms. The spell checker usage depended on the type of application such as machine translation like OCR, web search, and/or information retrieval from archival documents. For invoice data

extraction, we will implement a spell checker, which has a custom dictionary to search for error words and, finally, propose a solution by analysing and implementing different algorithms to correct these errors. This custom dictionary will contain words around invoice and receipts related words used worldwide in different invoice related documents.

The following section will introduce the background theory on how the spell checker was implemented by analysing the research that guided the final, proposed solution related to errors introduced in the spell-check testing. In this theoretical discussion, we will classify the errors found in OCR. Furthermore, we will discuss the solution methodologies used in our spell check to correct and detect non-word errors and different algorithms used to implement them. Later, we will share the result analysed of different methodologies used in designing a spell checker, along with the output generated from each methodology. Finally, we will conclude the work done and discuss any future work that might be carried upon.

4.2. Word Error Analysis

In the pursuit of developing a better methodology for spell checking that is specific to our system requirements, we used and analysed different algorithms and data structures in our path, such as the Levenshtein Distance algorithm, cosine similarity, and BK (Burkhard Keller) trees data structure, machine learning-based Bi-LSTM. Finally, after analysing the efficiency of these algorithms, we proposed a solution. A spell checker was proposed, which replaces the text with the encoding decoding sequences with the help of the Keras sequential model.

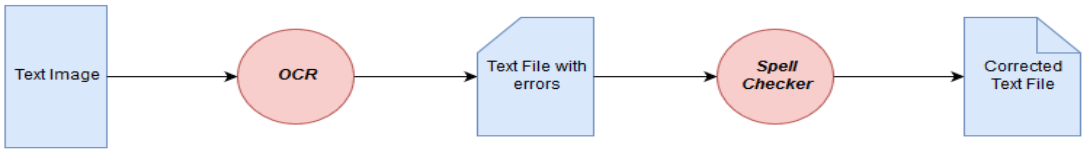


Figure 45: Basic flow diagram of OCR spell check.
"Devised by the author".

To understand the OCR engine and spell check process that needs to be performed, consider the example shown in the diagram above. The scanned invoice document in the form of an image or pdf goes through an OCR software scanning process to extract text from it. This extracted file typically contains some errors. The text file generated with errors needs to be corrected either by proofreading, in which case suggestions are provided from the OCR spell checker for each word or by automatically correcting with the most appropriate word. As already stated, proofreading can be time-consuming and error-prone due to human error. Since we have a custom dictionary for spell correction for the invoice slips, we can automate spell correction by various methods. The final output file will be a corrected text file based on automatic word replacements. We will study various methods that can be applied for word replacement.

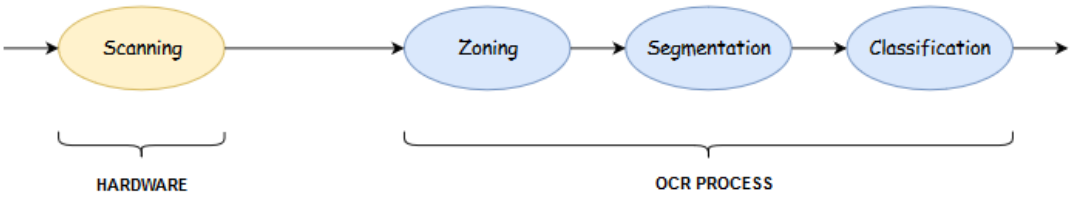


Figure 46: OCR Standard Procedure.
"Devised by the author".

Before jumping to errors solutions methodology, it is essential to understand some basic OCR conversion process to quickly lookup these errors. The standard procedure is responsible for error generation.

Table 21: The following phases are described for error generation at various steps (Taghva & Stofsky, 2001).

Phase	**Problems**
Scanning	The original document's poor paper/print quality, bad scanning equipment, and other factors may all contribute to scanning issues. As a result of such errors, errors may occur at any stage of the conversion process.
Zoning	Incorrect decolumnisation was caused due to such errors. They may manipulate the order of words in image text, resulting in an incorrect document.
Segmentation	An original document with overlapping characters, or non-standard fonts, broken characters may cause segmentation errors.
Classification	Classification errors were remarkably like segmentation errors. They typically result in single character replacements in a word, including some other effects than segmentation.

Furthermore, the various errors produced by OCR are described below. These errors were the most common errors produced by the OCR and are essential when working with OCR as they are more related to the OCR aspect rather than the Spell checker. `(Niklas, 2010) classified the various type of errors encountered in OCR into four types:

- **Segmentation Error:** Segmentation errors, as also explained earlier, is caused by an original document with overlapping characters, or nonstandard fonts, broken characters, as mentioned earlier. Different words, line, or character spacing causes white spaces to be misread, resulting in segmentation errors. For example, ‘whatis’ instead of ‘what is’ and ‘perma nent’ instead of ‘permanent’
- **Misrecognition of character:** Noise and font changes was used to avoid accurate character recognition, resulting in incorrect word recognition. For example, instead of ‘code’, use ‘codecc' or ‘’invoice@’ instead of ‘invoice.'
- **Case sensitivity:** Due to font variations, upper and lower-case characters are mixed up. E.g., ‘inDiA’ instead of ‘india’.

- **Changed Word Meaning**: These types of errors are related to the semantic of the sentence. This arises due misrecognition of characters that results in new words. The words are spelt correctly but change the contexts of the sentence (Niklas, 2010). In spell check classification, they classified it as a real word error. For example, Cake the sentence construction as, there was confusion between {piece, peace}(Kumar, 2016)

The primary focus of the spell checker for OCR generated text was to analyse different methodologies for spell checking that can be applied to the machine-encoded text generated from the OCR. OCR generates text either by pattern recognition or feature extraction but also generates error-based words during various phases of text generation like tokenisation, etc. The spell checker typically employed contains mostly two types of errors. These errors were essential to be understood to correct the errors generated by the OCR. The error generated can be broadly classified by spell checker as:

- Non-word (spelling) errors and
- Real word (semantic) errors.

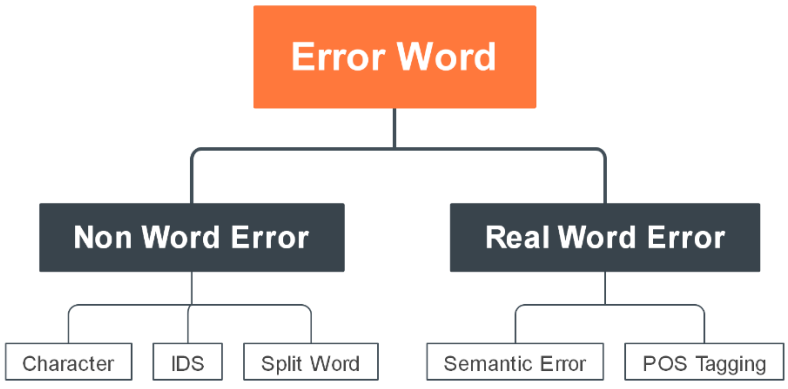


Figure 47: Word Error Types.
"Devised by the author".

A. Non-Word Error

When the correct spelling of a word is known, but the word is misspelt, the non-word or typographic errors occur. These errors do not follow any linguistic. A study conducted by (Damerau, 1964) and later confirmed in (Ren & Perrault, 1992) shows that 80% of typographic errors fall into one of the four groups below, which were known as IDS errors.

- Single letter insertion, e.g., “hinvoice” for “invoice”
- Single letter deletion, e.g., “invoic” for the “invoice”.
- Single letter substitution, e.g., “pride” for “price”.
- Transposition of two adjacent letters, e.g., “enxt" for “next”**.**

Moreover, single-error errors are those errors that are caused by any of the above editing operations. (Kumar et al., 2018). Other error includes split-word errors and merged errors.

- Split word: When space is wrongly inserted within the word.
- Merged words or run-on: When the space between two or more words was not inserted.

The split word and run-on errors were to be checked before going for IDS error correction. Usually, the character string was checked in a word list (dictionary). When there was no match, this string was not a valid word. Then the correction effort was started (Samanta & Chaudhuri, 2013).

B. Real Word Error

When the correct spelling of a word is unknown, these errors occur. When it comes to cognitive mistakes, the misspelt word has the same or similar pronunciation as the expected correct word (Naseem & Hussain, 2007). They are considering the same example for the sentencing piece of cake. Here, peace is not an error word, but it should be a piece according to linguistics. These errors are generally not identified by the spell checker using dictionary lookups.

Our spell checker is specific to and focuses only on non-word errors. Finally, we proposed a better solution to the problem. The main objective was to detect and correct the non-word errors. We used two principal methods:

a) First, check the misspellings generated from OCR with a term occurring in the dictionary called a dictionary lookup.
b) Second, to construct an appropriate Bi-LSTM approach for raw text data, which will extract expected words with the help of an encoding and decoding mechanism. In this mechanism, the mapping function is used to get the exact probability of a word.

4.3. Literature Review

In the literature review of the error correction chapter, the third research objective that is 3(a) *"To study and investigate the role of text correction and enhancement in the decision-making process of the system." is* achieved. Spell checking is a vast field, and much research has been done and is still going on these days. To improve accuracy and understand the best possible methodology for our specific task of non-word error correction. We performed a lot of research that we will be discussing here. (Taghva et al., 1994a) proposed a system in which OCR generated text quality has been improved. The information retrieval system is efficient in case of OCR error. Because here, more

knowledge and statistical information were extracted from the text. Most errors were occurred due to the space addition and removing at the time of typing or formation of the text. After the post-processing, 91% wrong spellings were found; and later, those were corrected. The future scope was to make the system automatic without user interference. Later (Perez-Cortes et al., 2000) used a stochastic error-correcting and parsing approach to identify the error and correct using a simulation model. They tried the deterministic and non-deterministic approaches to test word correction in the name field. There were roughly 27125 first names. Result achieved was around 1.74%-word error for incorrect names as compared to 32.54% in post-OCR. In this paper, only specific fields were targeted to show how the model behaves. Furthermore, additional errors appeared in invoice related documents which could not be corrected.

According to this paper, (Takeuchi & Matsumoto, 2000) used the OCR to recognize the error. The recognized errors were corrected by using the linguistic information online. For these, different methods were used, such as character trigram, stochastic morphological analysis, and word trigram models. For training, all the model's large untagged text was used. It gives a 94.3% correction rate. Furthermore, based on research (Pal et al., 2000), proposed the system for the Indian language for error correction and detection. The morphological approach was used to separate root words and suffixes. For the experiment evaluation, the Bangla document was used. Here only a single character error method was used, which gives an error correction rate of 84.22%. In future, they want to implement it for multiple characters.

In 2002, (Majumder et al., 2002) used “n-gram: a language-independent approach to IR and NLP”, the purpose of this research was to discuss the use of the n-gram approach in the field of Information Retrieval (IR) and NLP, which was made to be language independent. They explained n-grams usage in various fields and some preliminary experiments on some Indian documents. It has described the character and word n-gram concepts. It helped us to understand that we should be using the character n-gram. The drawback was that it does not describe the recent usages in the research as it was just in

the beginning phase. Furthermore, (Kruatrachue et al., 2002) implemented a genetic algorithm based on the Thai OCR error correction method. The word graph was constructed from spelling with the help of vocabulary, and then the graph searches the correct sentence using the model like language model, bi-gram, and trigram. For searching the words, different size nodes were constructed from 10 to 200.

(Kolak et al., 2003) used a generative probabilistic OCR model. This model was designed to improve error correction after OCR was performed. The objective was to enhance NLP operation with better post OCR text results. The model behaves better in the cross translation of text scenario. The model was tested on English OCT on French text, and the results were better than a commercial French OCR. The authors wanted to evaluate other language translation processes. The paper lacks any discussion on the importance of the machine learning model approach, and evaluating that will surely increase the accuracy.

(Zhuang et al., 2004), used the local information and global information, the language model along with combinational model that was conventional n-gram language model and the new LSA (Latent Semantic Analysis) language model. This technique gives better results when the candidate list contains more correct characters, so to achieve this Viterbi search process was used because it has Chinese like characters. The 60.9% error reduction was achieved by improving the candidate list. Later, (Watcharabutsarakham, 2005) used a statistical method to detect and correct the spelling without using the dictionary. NECTEC (National Electronic and Computer Technology Centre) was used to collect the data and to evaluate the system, ArnThai software was used.

Later, (Bassil & Alwani, 2012) proposed a context-based error correction algorithm for detecting and correcting OCR non-word and real-word errors in the post-processing stage. The suggested solution was based on an online spelling recommendation from Google, which consists of the database created based on terms and words sequence found on the web. They achieved an error rate of 3.1% and 3.1% on English and Arabic, respectively,

of 126 and 64 words. While OmniPage gave the error rate of 21.4% and 12.5%, respectively. In future, they plan on applying multiprocessing options. The algorithms described here are similarity keys, edit distance, rule-based techniques, probabilistic techniques, neural networks, and n-gram based techniques. These techniques were introduced in brief, which helped to gain some more techniques which were already existent. This paper describes some already available tools in various Indian languages.

Furthermore, (Gupta & Mathur, 2012) proposed research for the modern spell checker to Hindi corpus using the NLP technique. It focuses on the idea that most spell-checking systems detect and correct errors on word level rather than character level and use a dictionary concept. The spell checker checks the words in the dictionary; when not found, it was detected as an error, and for correction of those errors, searches the word in the dictionary and suggest the words that match best fit in the dictionary using the different algorithms. The character n-gram approach from this research for matching two strings based on sets to calculate the accuracy between words was a great idea (Gupta & Mathur, 2012). The purpose of this research was to do a comparative study of various techniques to correct the various type of errors detected by spell checkers on dictionary lookup. It also discussed some tools with their environment to find the difference between functionalities. It has discussed Levenshtein Distance and n-gram in NLP, with tools implementing them. Since it is only a survey, it helped us to understand more frequently used techniques. They also found that introducing the padding spaces in character n-grams can improve accuracy (Shah et al., 2012).

According to (Mohapatra et al., 2013), the character reproduced in encoded digital form produces errors which were influenced by the reproduction quality (original v/s photocopies), resolution of the scanned document or noise inherited by digitalisation process (like in boundary detection) or any mismatch between the characters on which character image classifier was trained and with the rendering of characters in the printed document. Noise may cause two characters to merge as one or recognize a character incorrectly or even may not recognize a character. It uses an automated correction

mechanism and uses techniques like n-gram analysis and confusion matrix. The correction system was based on approximate string matching. They used cases for string matching and an assumed accuracy of 70% at an OCR level. Using this approach, they gathered the knowledge required for pattern matching using n-gram analysis for our spell check design (Mohapatra et al., 2013).

The purpose of this research by (Bhaire et al., 2015) was to form a spell checker application using auto-suggestion techniques that can be integrated with larger applications. For the auto-suggestion part, the Levenshtein algorithm was used and java NetBeans for application development. According to (Kissos & Dershowitz, 2016), a machine learning algorithm was proposed to learn the number of features and correct misspelt words from the OCR. For the experimentation, the Arabic language was used. The network gave a 35% reduction in word error rate. For the training, 250 document images were used, which contained 60,000 words. The future scope was to improve the performance of candidate ranking and classification. Furthermore, the study has been carried out by (Muhammad et al., 2016), who stated that the model mainly focuses on character segmentation and alignment for single and multi-character. The evaluation shows that the multi-character recognition model was better than the single character trained on 502,167 words. The model has a 94% correction rate.

Later (İnce, 2017) has proposed the software for spell checking the n-gram based approach was used and error correction with the edit distance algorithms for the agglutinative language, which was Turkish. The evaluation resulted in software checks of 10,000 words per second. The system's performance as a spell check was 95% success rate, and spelling errors was 86% success rate. This research also explains the process of a spell checker, which must be followed while designing. Hence the design of spell check was similar to our spell checker with a slight difference in a development environment, which in our case was a python programming language. They used various cases to automate the spell checker for automatic correction. They used the first suggestion from a list of all possible suggestions for the automated task. The merit of the research gave

the idea to reduce the search time complexity. The TRIE data structure uses prefixes for searching, but instead, they used a dictionary lookup to identify the correct word and the Levenshtein Distance algorithm to find the best possible suggestion. The only drawback of the system was when the error was at the starting of the word; it may not give any suggestion.

In this paper, (Shah, 2017b) presented research on a tree data named Burkhard Keller (BK) tree structure that was used for spell checking based on the Levenshtein Distance concept. Various software's auto-correct feature was implemented through BK trees. The core concept behind this research was how the search time is reduced in fuzzy string matching using Levenshtein Distance. It was slightly a newer approach for this method problem. As basically, they were using the brute force approach previously to check the incorrect words with all the dictionary words with their respective edit distance. In this approach to find the nearest match for the incorrect word, they were equated all of the words in the dictionary to the Levenshtein Distance. It was taking a long time with the threshold value, and the complexity was O= $nl * m * n2$. Where n1 is denoted as the total number of terms, and m was the mean size of a perfect match, and n2 was the length of the incorrect word. Using the BK tree specifically reduces the time complexity to $O = nl * n2 * logN$, where n1 was the mean length of the string in their dictionary and n2 was the length of the incorrect word. That was approximately log N, where n is the number of elements. It was the new advancement for their spell check design as they have slightly reduced the search time complexity (Shah, 2017b).

According to this paper (Zhou et al., 2017b) have developed a model for spell correction for foreign language that was for English text. For that, they developed an encoder-decoder based framework. To solve sequence to sequence learning, RNN was used. It was not clear which technique to use when the input and output sequences were different lengths. The network detailed that the encoder was a multi-layer RNN model, the first layer of the encoder was a bidirectional RNN, and the decoder was a multi-layer RNN.

For the experiment analysis, the e-commerce dataset was used. They achieved 62.5% accuracy. They have not mentioned any gaps and future scope for their research. Furthermore, (Cappelatti et al., 2018) proposed a system related to the conversion of text information into the voice called a Vocalizer. The OCR was used for the detection of text from the captured image. The following part Needleman-Wunsch algorithm for conversion and modification, was implemented with the help of spellchecker PyEnchant. According to (Srigiri & Saha, 2018), the system implemented was implemented, which used spelling correction with the help of OCR, neural word embedding and Levenshtein Distance. To learn the word embedding, the Continuous Bag-of-Word (CBOW) model was used. To detect the spelling errors, context information and entity recognition was used; if the error was found, the word embedding gives an appropriate word from the candidate list. By using Levenshtein Distance, the distance was calculated between the wrong word and candidate list. Distance value less indicates the correct word or spelled correctly.

As stated in this paper, (Roy, 2019) implemented "Denoising Sequence-to-Sequence Modelling for Removing Spelling Mistakes." The drawbacks of rule-based methods were explained. The encoder-decoder architecture was developed. As the regular encoder-decoder structures were implemented with the help of RNN, but they designed it by using the stacked self-attention and pointwise fully connected layer. The decoder uses the 6 blocks: residual, multi-head attention, position-wise feedforward network, normalisation layer, SoftMax and masked multi-head attention. For the evaluation, Wikipedia articles were used. The accuracy of the proposed model was 97.93% when the number of errors was 100. Later, (Ramena et al., 2020) implemented a combinational model with the help of basic three models as CNN, bi-directional LSTM and conditional random fields (CRF). The CNN model was used to extract the morphological information from the character for the encoding. The Bi-LSTM model uses the RNN unit and extracts contextual information from the sequence in both the direction. The purpose of using the CRF model

was for the structured prediction of sequences. For the experimentation, the Wikipedia text was used. They achieved a model accuracy of 94.02%.

Furthermore, (Singh & Singh, 2020) proposed a deep learning-based model to detect and correct the error. The system has two steps, one was to identify the error, and the second was to correct the error. Spell-checkers were designed with the help of traditional methods like statistical methods and rule-based methods. The HINDIA model works on an attention-based encoder-decoder bidirectional recurrent neural network (BiRNN). For the evaluation, the model uses the 'monolingual corpus' dataset, which IIT Mumbai for training and testing developed. The testing accuracy of the model was 74%.

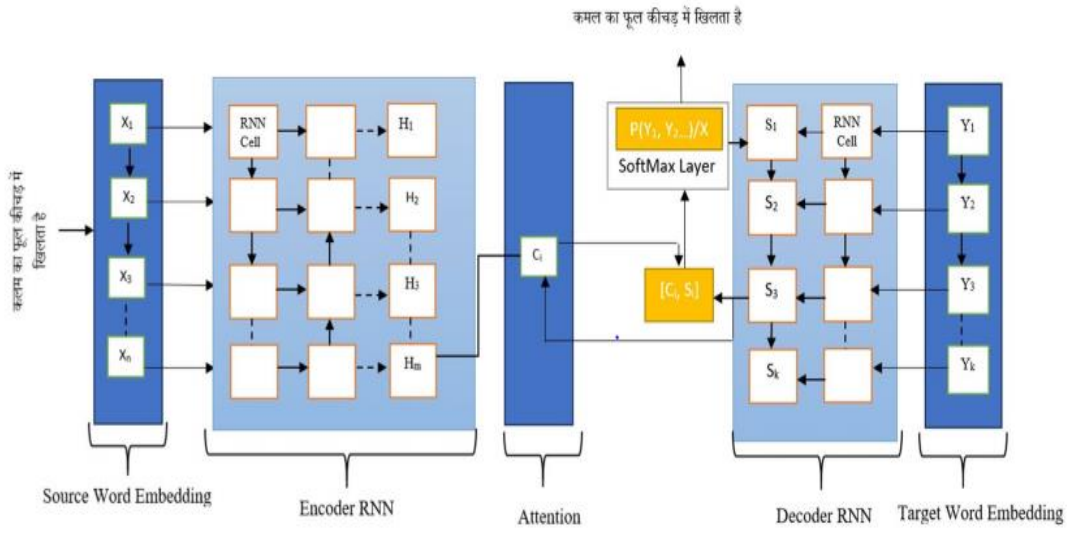


Figure. The block diagram of the BiRNN model (Singh & Singh, 2020)

The Encoder–decoder RNN (En-De RNN) takes the input and encodes it in a fixed vector length. The Encoder–decoder RNN (En-De RNN) was useful when input and output have different lengths. The SoftMax layer was used for word-to-word classification. The BiRNN model has the advantage of analysing the text's predefined state in both directions until making an error correctness decision. The model's disadvantage for the internal design requires a fixed length.

Furthermore, the research has been carried out, by (Ahmadzade & Malekzadeh, 2021), by using a deep neural network-based model to correct spellings of the Azerbaijani language. For that, encoder, decoder, and attention models were used. The encoder and decoder consist of RNN with embedding and LSTM layers. The use of the attention model generates a context vector that conveys the information from encoder and decoder architecture. The proposed system was tested on 3000 real words. The system gives 98% accuracy. Later, a newly developed system by (Amin & Ragha, 2021) used RNN was to generate a grammatically correct sentence, word and paragraph from the input applied and translate the corrected output into the Hindi language. After the text generation, NLTK was used for grammar correction. The system gives the two outputs, such as one English text and the second is translated Hindi. (Schaback & Li, 2007) have also proposed the spelling correction method by using a machine learning-based multilevel feature extraction model. To predict the correct candidate, the SVM was used. For the experimentation, English plaintext from Wikipedia with the size of 1.5 gigabytes was considered. The result evaluation of 97% recall was achieved during the correction. Future scope in the performance of misspelt was suggested to be discussed for improvement.

According to (Etoori et al., 2018), the model proposed used the Sequence-to-sequence text Correction Model for Indic Languages (SCMIL), which consists of an attention-based encoder-decoder using RNN. For the evaluation, Hindi and Telugu movie names from Wikipedia were used, which were released between 1930 and 2018. This model gave 85.4% accuracy for Hindi and 89.3% for Telugu. (Zaky & Romadhony, 2019), LSTM model encodes the input text at the character level and the POs tag as a context feature. For the evaluation, Indonesian Wikipedia articles were used. The testing accuracy was 83.76%.

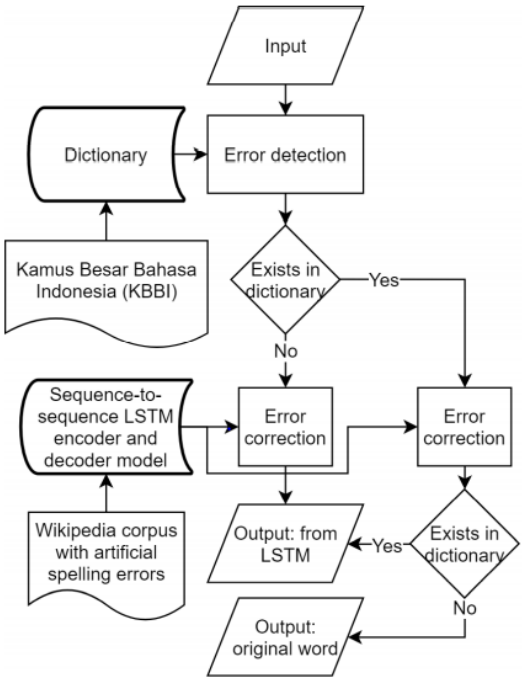


Figure 48: System architecture of LSTM model (Zaky & Romadhony, 2019)

This model was used for spell error detection and spelling error correction. There were two pre-processing steps, such as text tokenisation and POSTagging. The error detection was based on the dictionary lookup technique; here, the KBBI dictionary was used. The Sequence-to-sequence LSTM encoder and decoder model uses various features such as Character one-hot vector, Word Embedding, Part of Speech (POS) Tag.

a) Character one-hot vector: A mapping vector was used to check if the character was in uppercase or lowercase. In the Indonesian words, * indicates the end of the sentence and /t indicates start. The character vector vocabulary size was 55.
b) Word Embedding: It was used to convert the text information into a vector. The word embedding vector had size 400 that was used for training.
c) Part of Speech (POS) Tag: The vocabulary of the POS tag vector was 21. The size of POSTag was 42, that is, vocabulary multiply by 2. This was the main feature where the word is converted into a vector.

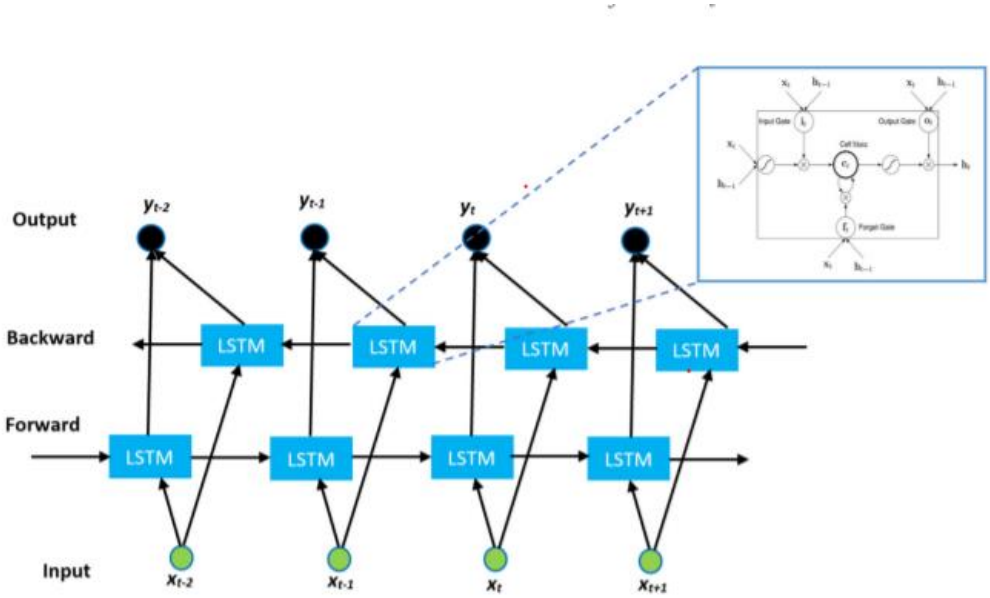


Figure 49: Overview of the Bi-LSTM model (Rahman et al., 2021).

Furthermore, (Rahman et al., 2021) proposed the "A Bidirectional LSTM Language Model for Code Evaluation and Repair". The proposed model finds the code errors. The Bi-LSTM model works for both the data earlier as well as later data. The Bi-LSTM model computes the code sequences (both forward and backward) to search for the best possible word based on the highest probability. The proposed Bi-LSTM model was found to achieve 50.88% accuracy in identifying errors during the experimentation. This model has the limitation that it was only helpful for error detection. Later, (Murugan et al., 2020) proposed a method for Tamil spell check. The spell checkers were used to check that the input has a spell or any grammatical error. They designed three models as bloom-filter, symspell, LSTM based spell checkers. The Bloom filters were dictionary-based. The sympell uses the TRIE data structure and base algorithm as edit distance. The symspell has a memory problem; due to this, it has a slow speed of operation. Finally, they implemented the LSTM based model. They have not discussed any experiment result in analysis or comparison. As the LSTM model are unidirectional, there was only present information or data was available. Based on that, it generates the new text. Consider an

example, “It is”, so it may generate “a”, “a table” like this because the future words are not available. Due to this wrong word, generation takes place. To overcome the drawback of LSTM, the Bi-LSTM model was proposed, which consists of the present as well as the future information.

The detailed summary chart of spell checker related study has been listed as below:

Table 22: Spelling Check Literature Summary.

"Devised by the author".

Area	**Model/ Method**	**Maximum accuracy**	**Gaps identified**	**References**
Plaintext	Machine learning-based multilevel feature extraction model and SVM n-gram and Levenshtein Distance algorithm	97%	To improve the classification.	(Schaback & Li, 2007) (Shah et al., 2012)
Azerbaijani Language	Deep learning-based encoder, decoder, and attention models, RNN	98%		(Ahmadzade & Malekzadeh, 2021)
Hindi language	RNN, NLTK attention-based encoder-decoder bidirectional recurrent neural network (BiRNN) Continuous Bag-of-Word (CBOW) model, Levenshtein Distance	95%	To use this system in the search engine, Grammatical errors should be minimized.	(Amin & Ragha, 2021) (Singh & Singh, 2020) (Srigiri & Saha, 2018)
Resource-Scarce Languages Indic languages such as Hindi, Telugu	Sequence-to-sequence text Correction Model for Indic Languages (SCMIL) Encode- Decoder IR and NLP Symspell	For Hindi, 85.4% and Telugu 89.3%	The automatic error correction and synthetic dataset should work for noisy images. o The correction should be improved in depth. o To improve the retrieval efficiency in the case of the Indian language.	(Etoori et al., 2018) (Mohapatra et al., 2013) (Gupta & Mathur, 2012) (Majumder et al., 2002) (Murugan et al., 2020)
Page image	OCR	91%		(Taghva et al., 1994a)
Sentences	genetic algorithm			(Kruatrachue et al., 2002)
Chinese Language	Conventional n-gram language model and the new LSA (Latent Semantic Analysis) language model.	91.9%		(Zhuang et al., 2004)

Thai documents	statistical method			(Watcharabutsarakham, 2005),)
Arabic document	A post-processing context-based error correction algorithm for detecting and correcting OCR non-word and real-word errors. The multi character recognition model Machine learning algorithm	98%	o To redesign the model, which was useful for multiple documents. o For the ceiling analysis, the candidate rank and generation should be improved.	(Bassil & Alwani, 2012) (Muhammad, (Muhammad et al., 2016) (Kissos & Dershowitz, 2016)
Windows Based Application	Levenshtein Distance algorithm, NetBeans java application			(Bhaire et al., 2015)
Turkish	the n-gram and edit distance algorithms		To expand the system for cryptology and OCR recognition.	(İnce, 2017),
Search engine	a tree data named BK tree (Burkhard Keller) structure, Levenshtein Distance concept			(Shah, 2017b)
Wikipedia	Denoising model Combinational model	97.93%		(Roy, 2019; Ramena et al., 2020)
E-commerce documents	multi-layer recurrent neural network	62.5%		(Zhou et al., 2017b)
Code	Bi-LSTM	50.88%	This model has the limitation that it was only helpful for error detection.	(Rahman et al., 2021)

Finally, most existing works only focus on spell checking tasks for the plaintext and on different types of language, but none of the researchers has specified the separate spell accuracy for the invoices/receipts text extraction. The overall accuracy for the system is high, as mentioned but, text extraction accuracy is not mentioned. For fair comparisons, we have trained the different network which is identically used for spell check. Based on the literature review, we want to comment few points,

d) Not a single researcher has proposed the spell check based model for the invoice/receipts spell checking.
e) No one has mentioned the separate accuracy of the model for key-value pairing or any pattern.
f) The results are not adequately explained.

Therefore, it was finally decided to validate different spell-checking methods and enhance the system to the next level.

4.4. Comparative Study

This chapter will only focus on non-word errors as real word error is beyond the scope of this work. The various methodologies and techniques have been discussed, like Levenshtein Distance with BK trees, cosine similarity and n-gram approach, machine learning-based LSTM, and Bi-LSTM. We have also seen that the most common was based on Levenshtein Distance and n-gram similarity. The objective was to find out what best suits our needs using a custom dictionary. Throughout this chapter, we tried to analyse different research for different methods; for efficiently removing non-word errors and applying the methods based on the merits keeping in mind the drawbacks of each.

In the previous section of this chapter, we concluded that we could use some different approaches like Levenshtein Distance with BK tree, n-gram, and machine learning-based methods such as LSTM, Bi-LSTM which can help us to correct non-word errors. The spell-checking process was well understood from a design methodology perspective. The research also explained the detailed process of a spell checker that should be considered while designing the system. This section will look at those methodologies in detail, which will help us uncover which one to use for our problem specification. We will see at

incremental steps how we have improved the correction with each methodology. The improvement was analysed by a CSV file generated at every phase that contained incorrect words and their respective corrections.

We will do our study by applying Levenshtein Distance (Levenshtein, 1966), n-gram's string similarity algorithms and the Bi-LSTM model. The Bi-LSTM model is trained with the help of Keras sequential model in terms of encoding and decoding the sequence. The suggestive process is done by calculating the similarity score using the Dice coefficient and word distances. Finally, further implementation of machine learning techniques was also added. This helped to increase the accuracy of data extraction significantly. The various methodologies used are described below. We used the methodologies in incremental order describing the advantage and drawbacks of each approach.

4.4.1. Levenshtein Distance with BK tree

Levenshtein Distance is fast, but it has certain drawbacks as it could not automatically replace the best-fit words according to a calculated threshold value. Although it can replace words, they are not the best first suggestion. This is the major problem as the replacements are not correct for some words even though they are in their suggestion list. It is also unable to give suggestions for words when they have some extra characters or symbols at the end or start of words. So, we go for another method for spell checking.

(Shah, 2017a) proposed the BK tree structure for spell checking and string matching. A BK tree, also known as a Burkhard-Keller tree, was a metric space tree knowledge structure. In BK-tree, a tree was used to select all of the components of a fixed set that were similar to the inquiry part. The most straightforward method of locating every single nearest component is to compare the query component to every other component of the settled collection. They used the Levenshtein Distance for equating the two strings. They have specified the metric space relationship equation between a and b as,

1. When the separation between a and b equal to zero, then a = b, it was represented as,
   d(a, b) = 0 then a=b
2. The distance between a and b is the same as the distance between b and a, represented as given below,
   d(a, b) = d(b, a)
3. The distance between a and b algebraic sum with the distance between b and c, which will be equals to the distance between a and c, given below
   d(a, b) + d(b, c) >= d(a, c)

The above three conditions are known as the Triangle Inequality. They have given the two operations of the BK-tree method as search and create. The detailed explanation of operations as,

- **Create Operation**

Consider a dictionary as {“FALL”, “CALL”, “SAIL”}

The elements in this above dictionary will be shown at the nodes of the BK-Tree, and there will be the same number of elements as the number of terms in this dictionary. In the above dictionary, there were three nodes. The edges were represented with the help of edit distance(Levenshtein Distance d). The first element of the dictionary indicates the root node. The distance d was calculated as,

LevenshteinDistance d(FALL, CALL) = 1

LevenshteinDistance d(FALL, SAIL) = 2

The value of Levenshtein Distance d between FALL and CALL was 1 and Levenshtein Distance d was 2 for FALL and SAIL.

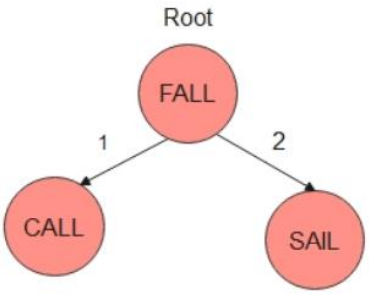


Figure 50: The Levenshtein Distance between the root node and two child nodes.
"Devised by the author".

The figure above represents the edit distance between the root and child node in terms of the Levenshtein distance formula.

- **Search operation**

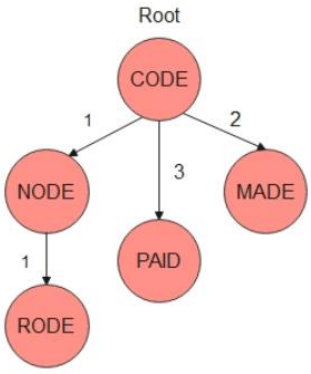


Figure 51: Sample example of string-matching using BK tree.
"Devised by the author".

The figure above represents the dictionary for searching for the correct word. To find the word start from the root node and search the correct word from the child node by moving from left and right till the end to get a minimum edit distance. To get the corrected word, first, define the tolerance T. BK tree was constructed using edit distance formula and misspelt words can be identified by searching over children with edit distances ranging from [d-T] to [d+T].

Consider an example, misspelt word as “ODE” and tolerance T is 1.

Step 1: Start searching from root node, d(“ODE” -> “CODE”) = 1 range [0,2], when the d<=T then it is included in the correct word dictionary. So, here the “CODE” is one of the matches for the given misspelt word.

Step 2: Now consider the left child node d(“ODE” -> “NODE”) = 1 range [0,2], it is also included in correct word dictionary.

Step 3: Now at the last node of tree, d(“ODE” -> “RODE”) = 1 range [0,2], this is also included in dictionary.

So, for the misspelt word “ODE”, got three correct words as {“CODE”, “NODE”, “RODE”}

Search Algorithm

a) Select the first node as the root node.
b) Define the tolerance T value, which is nothing but the highest edit distance between misspelt word and correct word dictionary.
c) Searching of correct over children ranging from [d-T, d+T], where T is tolerance and d is edit distance.
d) When d<=T, which indicates the correct word for the misspelt word
e) Search till the end of the tree, repeat the process from the second step to the fourth step.

Implementation

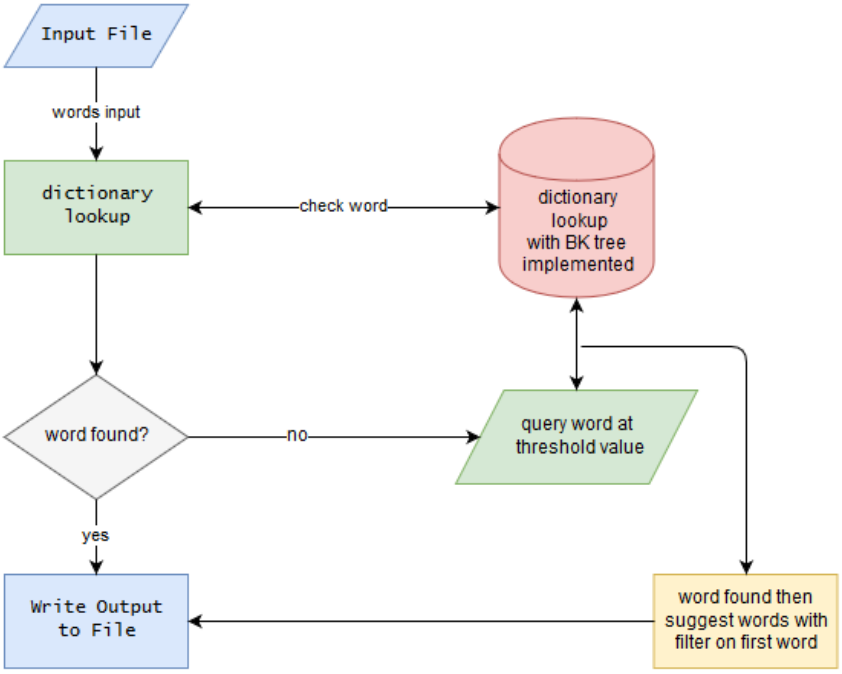


Figure 52: Flow Diagram of spell check using BK tree.
"Devised by the author".

To automate the correction of files with the best possible selection from the dictionary word, we have the concept of the BK tree, which specifically reduces the search time and bring efficiency. The working of the spell-checking approach used in the initial phase is mentioned in the figure above. We have OCR text files with some sort of non-word errors that need to be corrected. The input files are read from a directory word by word using word tokenisation, and the respective words are input to the dictionary lookup, where the validity of the word is checked. If the word is found in the dictionary, then it is valid and not replaced, while the word not found is termed as non-word errors. The dictionary is stored using in memory using the BK tree. For the error word, we have calculated the threshold value to ensure that the size of the input word be may not exceed the threshold limit. The threshold limit is the maximum size of the input word. To calculate the

threshold, we used the formula $\Phi = \frac{a \times \gamma}{2}$, where Φ denotes the threshold value, a is the length of the word, and $\gamma$is constant assumed to be 0.5. The threshold limit is set to be for a word length of 30. The calculated threshold value with the error word becomes the query input to a tree for searching and auto-correction; the query results in a list of candidates. From the list, a first-choice word is selected and replaced for correction for the error word.

Advantages

The tree is irregular and has an N array (but generally well-balanced). According to tests, searching with a distance of 1 query covers no more than 5-8% of the tree, and searching with two errors queries covers no more than 17-25% (KodeKnight, 2011). Hence it is an improvement over checking every node. When the radius is kept to 1 or 2, the search space is always reduced to less than 10% of the original (Lacchia, 2017).

Using the BK tree specifically reduce the search time complexity to O (n1*n2*log N), where n1 is the mean length of the string in our dictionary and n2 is the length of the incorrect word. That is approximately log N, where n is the number of elements.

Drawbacks

The main drawback of the approach was that it could recognize and correct only to edit the distance of 2. On analysis, we found that the correction for some of the errors, the technique was not fit as it required some sort of pattern matchings for auto-correction, i.e., it was unable to autocorrect words which had some extra patterns surrounding word like name#. Though it gave a suggestion, it was not the first suggestion.

4.4.2. N-grams Method

N-grams are combinations of adjacent words or letters of length n in the source text. It is the most widely used algorithm because its implementation is comparatively simple with quite good performance. The algorithm is based on the principle:

"If the word A matches with the word B considering several errors, then they will most likely have at least one common substring of length N".

(Sundby, 2009) presented one method for spell correction by using the n-gram approach. The proposed architecture of spell correction was divided into three categories: spell correction, initialization, and indexing. The data was loaded into memory during the initialization process. The dictionary and analysis data used for prioritising were included in this data. The spelling checker performs up to 100-500 dictionary and classification data lookups for each misspelt word in the worst-case scenario. Now next part was to do the indexing. When all of the data has been loaded into memory, indexes for both the dictionary and analysis data were created. There were two-character indexes (26x26 indexes) in the indexes, each with an integer value. The spell correction step used various algorithms such as deletion, insertion, substitution, and reversal.

Insertion Algorithm

Insertion was a character-replacement algorithm that corrects misspelt words. They used insertion to place each letter of the alphabet, which in this case is A-Z, in each character location of the incorrectly spelt or typed word.

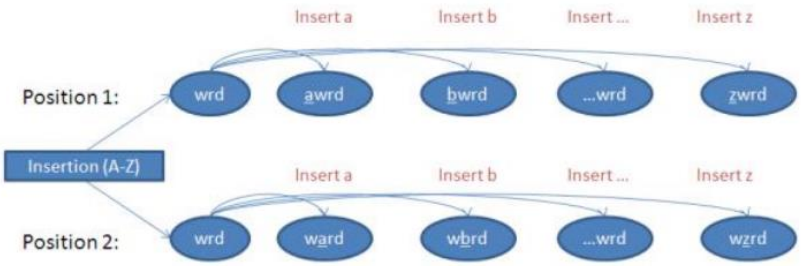


Figure 53: Insertion algorithm n-grams (Sundby, 2009)

In example above, they took the misspelt word “wrd”, and in the first position have inserted every letter of the alphabet from a to z. Similarly, in the second position, they

have inserted the character. If and only if the prepared word was included in the dictionary, it will be looked up and inserted into the word suggestion list for each character insertion.

Deletion Algorithm

The deletion algorithm was responsible for removing letters from a misspelt word. They deleted one character at a time using deletion, and in each deletion stage, they have to look up for preparing word in the dictionary to see when it exists. In the below diagram, "worrd" was corrected as "word" when they have deleted the fourth character from that word.

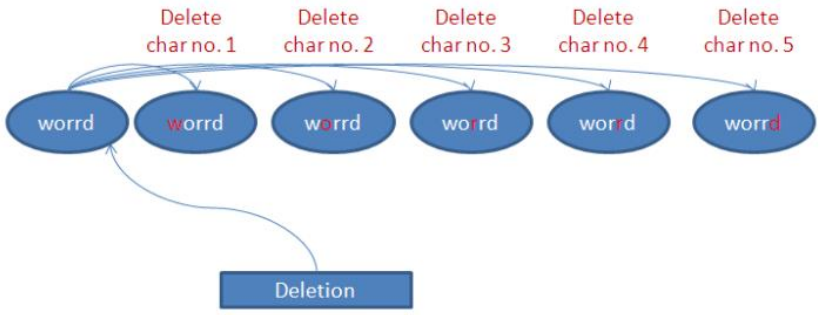


Figure 54: Deletion algorithm to delete extra characters from misspelt words (Sundby, 2009)

In this paper, they have discussed the Unigram, Bigram, trigram approaches. When the frequency of the bigrams or trigrams in the scope of the sentence was less than two, they used unigrams for prioritisation. With the help of the bigram model, information was used to prioritise word suggestions in the context of the sentence in which the word correction occurred. The trigram was an advanced bigram model where two bigram models were used to strengthen the prioritisation task. They achieved an accuracy of 95%.

(El Atawy & Abd ElGhany, 2018) proposed the automatic spell correction model based on the n-gram approach. They implemented this model for the English language. Based on lexical tools and n-gram statistics, the proposed model selects the most appropriate correction suggestion from a list of possible correction suggestions. The workflow of this

model is to accept the incorrect word and process it. The proposed method performs an n-gram for an incorrect word by comparing it to each word in the dictionary and returns the word in the recommendation list with a similarity coefficient of 1. The machine then chooses a word from the list of suggestions and replaces it in the input text. They achieved an accuracy of 85% over 2800 words with the help of the bigram method.

Algorithm

- Read the input file and word tokenize and implement the dictionary on words in the form of {key: value}
    - where the key is the input word
    - value denotes the character bigram set
- Check the key using dictionary lookup with stored character bigram with their keys
    - If input word key found
        - Write the word to the output file
    - Else use similarity between sets with a value of input word with the values of the stored bigram dictionary.
        - Return key with maximum similarity score and write it to the output file

Implementation

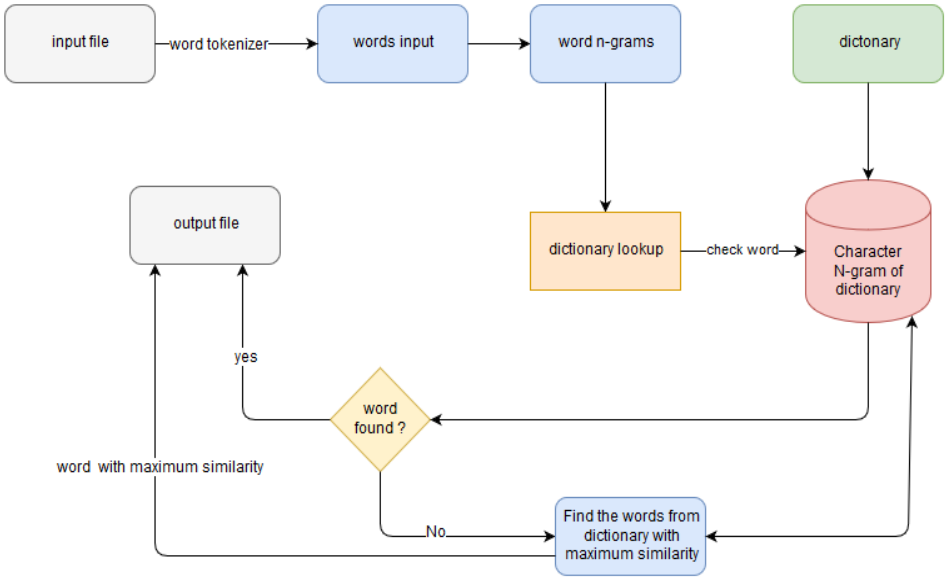


Figure 55: Flow Diagram of character Bigrams.

"Devised by the author".

In the sections above, we have briefly discussed algorithmic implementation. Now, we will be discussing the details of algorithmic implementation details through an example. To automate the correction of files with the best possible selection from the dictionary word, we have the concept of similarity score that is calculated using dice coefficient using set intersection logic.

The spell-checking approach used in the initial phase is illustrated by the figure above. A dictionary of words is stored as a dictionary of character bigrams where the keys are the words, and their values are the respective character bigrams. The OCR text files are read word by word using a word tokenizer. The words from the text file are converted into character bigrams in the same manner as the dictionary. The Key from the file is checked against the key to the bigram dictionary. If the key is found, then it is written to the output file; else, the value of the word, i.e., character bigram of the word, is checked with values of the bigram dictionary to find the similarity between the sets and print to the file the

word with maximum similarity. It will be clear with the error word not found in the key, as discussed in the example below. Example: Consider 2 strings, “catomer” and “customer”. If n is set to 2 (bigrams are being extracted), then the similarity of the two strings is calculated as follows:

Step 1: Initially, the dictionary is saved into {key: values} With keys as words and values as their bigrams with padding spaces.

E.g., {customer: {('s', 't'), ('u', 's'), ('e', 'r'), ('t', 'o'), ('m', 'e'), ('o', 'm'), (' ', 'c'), ('c', 'u'), ('r', ' ')} has 9 bigrams.

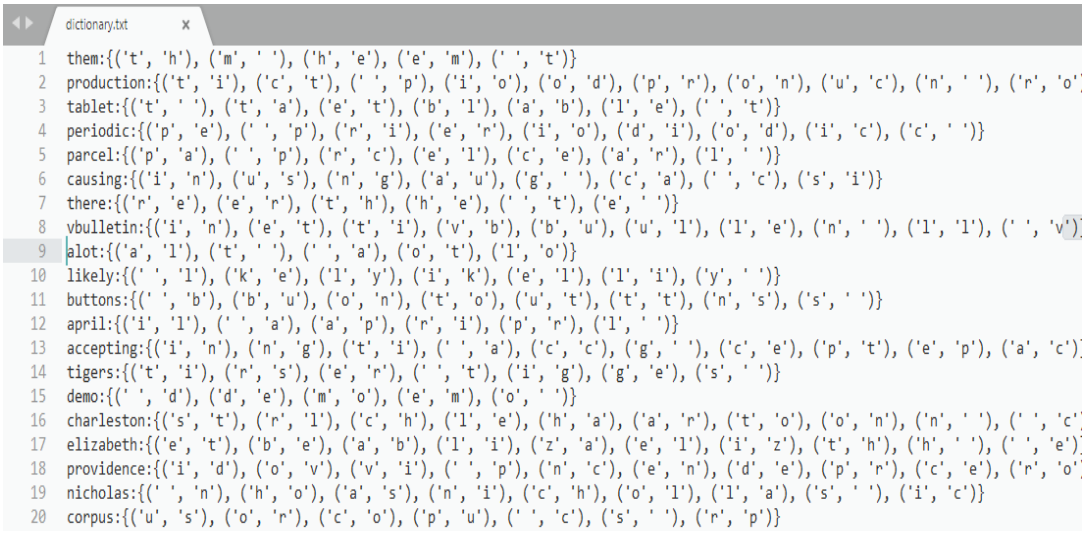


Figure 56: n-gram example.
“Devised by the author”.

Then input string is split into bigrams put into dictionary.

{catomer: {(' ', 'c'), ('m', 'e'), ('e', 'r'), ('r', ' '), ('o', 'm'), ('a', 't'), ('c', 'a'), ('t', 'o')}} has 8 bigrams.

Step 2: Find the unique bigrams that are shared with both the terms (Singh et al., 2020).

There are 6 such bigrams: {(' ', 'c'), ('m', 'e'), ('o', 'm'), ('r', ' '), ('e', 'r'), ('t', 'o')}

The similarity measure is calculated using similarity coefficient with the following formula:

Similarity coefficient = 2*C/A+B

A - unique n-grams in term 1.
B - unique n-grams in term 2.
C - unique n-grams appearing in term 1 and term 2.

Result: The example above would produce the result (2*6) / (9+8) = 0.75. The higher the similarity measure is, the more relevant is the word for correction.

The approach is implemented using set intersection logic. It calculates the similarity between the input bigram set with the sets of bigrams in the dictionary and return the similarity score as mentioned. Using this approach, we have tackled some more errors that were not recognized using Levenshtein Distance.

Advantages

This technique had an advantage over words that had a Levenshtein Distance greater than 3, which was a problem for the previous approach. It was also able to correct words by replacing them as common substring rather than edits matched them; that is, it was able to correct words that had some extra characters or symbols surrounding the word. This approach is language independent as it is easier to make n-grams for pattern matching by changing corpus or dictionary and then using Levenshtein Distance.

Drawbacks

It is slower than Levenshtein Distance as the dictionary needs to be converted to n-gram and stored in that form. i.e., scaling affects the performance of n-gram and comparison between sets takes time. It also takes higher memory for storing n-grams of the dictionary. Character n-grams are unable to correct merged word errors like ‘invoicenumber’.

4.4.3. Bi-LSTM Model

Based on the study of all existing approaches related to the spell check in case of invoice entity extraction, the model should extract the correct information, and for that, spell check plays a vital role. The most current of all approaches use the deep learning-based Bi-LSTM for the named entity extraction. Still, no research has been carried out specifically for invoice spell checking, so we have decided to implement the Bi-LSTM based model.

In (Liu et al., 2019b), the Bi-LSTM-CRF model has been designed for entity extraction. They have compared the performance of the system with two Bi-LSTM-CRF baselines. Baseline I apply Bi-LSTM-CRF to each text segment, where each text segment was an individual sentence. Baseline II applied the tagging model to the concatenated document. The results of their proposed approach were evaluated on the VATI dataset. The F1 score of this method was 87.3%. Again, it was not the value of the single spell check model; they have listed the overall model performance. It was a challenging task for the researcher to make a fair comparison.

Model	VATI	IPR
Baseline I	0.745	0.747
Baseline II	0.854	0.820
BiLSTM-CRF + GCN	0.873	0.836

Table 1: F_1 score. Performance comparisons.

Entities	Baseline I	Baseline II	Our model
Invoice #	0.952	0.961	0.975
Date	0.962	0.963	0.963
Price	0.527	0.910	0.943
Tax	0.584	0.902	0.924
Buyer	0.402	0.797	0.833
Seller	0.681	0.731	0.782

Table 2: F_1 score. Performance comparisons for individual entities from VATI dataset.

Figure 57: The performance comparison of the Bi-LSTM CRF (Liu et al., 2019b)

The method proposed by (Rahman et al., 2021) has a language model using a Bi-LSTM neural network. The limitation of the LSTM model is that it considers only previous context information. Due to the lack of this feedback mechanism, future context data is available. So, it was generating the wrong choice. The comparative results when the models were applied to the GCD dataset. The Bi-LSTM model performed better than the LSTM and RNN models (CoM: 52.4%; σ: 4.55; precision: 98%; recall: 95.5%; F-score: 96.7%) where the CoM represents the Correctness of Model which was 52.4% only.

Model	EIA	CoM	σ	Precision (P)	Recall (R)	F-Score
BiLSTM	66	52.4%	4.55	98%	95.5%	96.7%
LSTM [18]	45	33.2%	7.67	87%	89%	87.0%
RNN [18]	33	25%	7.76	80%	81%	80.0%

Figure 58: The performance evaluation of the Bi-LSTM model, which is compared with the previous model such as LSTM and RNN for the GCD (Greatest Common Divisor) data Source: (Rahman et al., 2021)

Model	EIA	CoM	σ	Precision (P)	Recall (R)	F-Score
BiLSTM	62	49.35%	3.12	97%	97%	97.0%
LSTM [18]	41	30.6%	6.09	90%	88%	88.0%
RNN [18]	29	22%	7.15	82%	79%	80.0%

Figure 59: The performance evaluation of the Bi-LSTM model, which is compared with the previous model such as LSTM and RNN for the IS (Insertion Sort) data Source: (Rahman et al., 2021)

The comparative results when the models were applied to the IS dataset. The Bi-LSTM model performed better than the LSTM and RNN models (CoM: 49.35%; σ: 3.12; precision: 97%; recall: 97%; F-score: 97.0%) where the CoM represents the Correctness of Model which was 49.35% only. So, we have concluded that the overall correctness of the model for the spell check is 52.4% as per this study.

4.5. Solution Methodology

In this chapter, we will analyse the result of the methodology used in the phases above. We use this analysis to determine the best methodology to use for our OCR generated text. Since Levenshtein was the most used algorithm, it was applied in the novel solution that we have developed and explained in chapter 3. We analyse the results of performance testing and improved performance by using BK trees. After analysing the results, it was concluded that the error correction approach was not accurate enough. We realised that we needed to employ better strategies for identifying similarity and pattern matching. However, it was also essential to understand the different theoretical aspects required to arrive at these methodologies. These aspects were analysed when the OCR produced the text output with errors, and a solution was needed to correct these errors. We ran a custom dictionary lookup to find all the error words and replicated them on a CSV file to analyse the errors in detail. We tried to find a solution using the theory of these errors. The process of spell check is as follows, The figure below is a sample invoice, which is used for the OCR text generation.

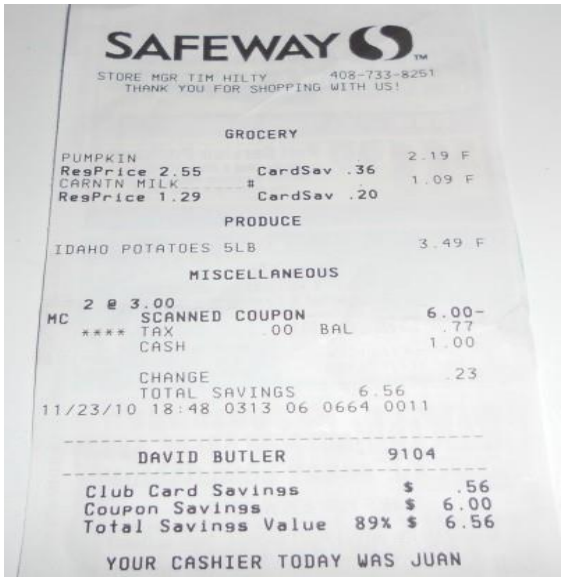


Figure 60: Sample OCR input for spell check.
"Devised by the author".

From the input sample image of the invoice, we must extract the text for further processing. The figure below is a sample OCR text generated from the image above.

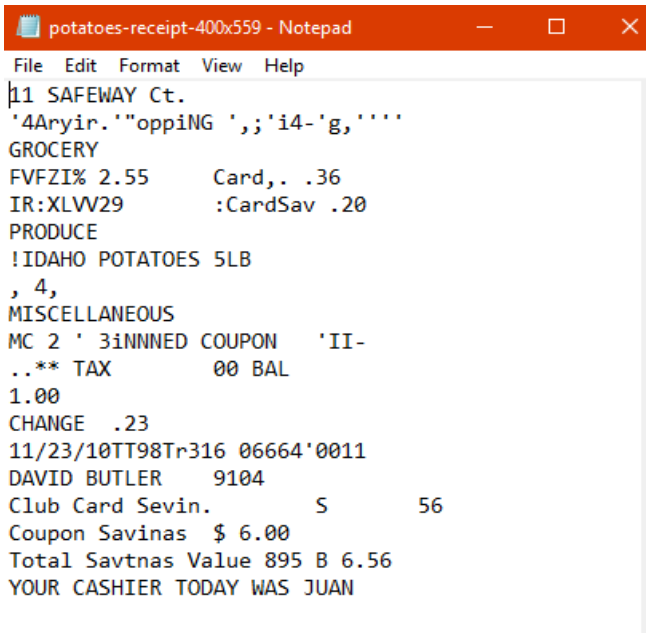


Figure 61: Sample OCR Text.
"Devised by the author".

Now, the next step is to check the spelling of the extracted text. This means spell-checking using a different methodology on the text generated by an OCR. We will also analyse traits of the corrections that occurred at both phases of the generated CSV. This will help us determine which approach offers the greatest value and, ultimately, will help determine which approach was better.

After executing all the methods on sample data, we generated CSV files that included results from three methods to analyse which approach was better. We analysed that Bi-LSTM was giving the best result as it was able to correct errors that were not possible by Levenshtein Distance and bi-gram approach. The errors which were corrected by n-grams were mostly of edit distance 3 and where the threshold was calculated according to the word was not correct. Sometimes, method 1 failed (i.e., returned NULL) due to extra characters. Additionally, there were some important observations when considering the

character bi-gram approach. Both methods were unable to correct merged word errors and real-word errors. These results were important to understand the approaches and to conclude what was needed from future improvements (e.g., it is better to merge both approaches for speed and accuracy of the spell checker). We analysed the hit ratio in both methods. The hit ratio was calculated using $\Delta = \frac{C}{E}$, where C denotes corrected Word and E is the no total number of error words in our sample dataset and Δ denotes the calculated hit ratio.

Table 23: Spell check hit ratio.
"Devised by the author".

Approach	Error Word	Corrected Word	Hit Ratio	Percentage Hit
Levenshtein with BK tree	1500	1234	0.822	82
Character N gram	1500	1342	0.894	89
Bi-LSTM	1500	**1375**	**0.916**	**91**

The analysis has been completed over the three methods as Levenshtein Distance with BK tree, character N-gram, and Bi-LSTM. For the experimentation, 1500 error words are considered for all the approaches. The Levenshtein Distance with BK tree method corrected 1234 words out of 1500, so the percentage hit of this method is 82%. The character N-gram method shows the 1342 words correction, the percentage hit of this method is 89%. Based on the literature review, further analysis has been performed for the spell checking, and the Bi-LSTM model is implemented, which gives better results than that of previous methods. The Bi-LSTM model has a percentage hit of 91%.

4.6. Performance Evaluation

In the section, further analysis of the proposed method has been tested and validated with the existing method, that is the 3(c) *"To validate the optimization in accuracy, cost, and speed at which the model works."* research objective is achieved here. After finalising the

spell check model, the proposed framework was updated, and the test was performed. The experimentation is carried out for the field level spell check extraction as well as the line-item level. Again, for the evaluation, three different types of datasets, such as real-time company datasets and two academic datasets, SROIE, VATI were used. As we have discussed, the text extraction model and the various performance measure values in the previous chapter. Now next part was to check the spell-check model performance along with the complete extraction system. The experimentation results have been tabulated in the followings list of tables for the different datasets.

Table 24: Result: SROIE dataset, Our method plus spell check.
"Devised by the author".

Method	**Precision (%)**	**Recall (%)**	**F1 score (%)**
Koo's (Koo, 2016)	68.53	72.81	70.61
Fastext (Busta et al., 2015)	69.69	81.89	75.30
EAST (Zhou et al., 2017a)	87.65	79.5	83.37
SegLink (Shi et al., 2017)	89.02	92.75	90.67
PSENet (Li et al., 2018)	96.70	92.40	94.34
W-A net (Anand & Khan, 2020)	83.60	86.10	84.83
DetectGAN (Zhao et al., 2020)	98.98	98.51	98.74
Our Method	98.85	98.65	98.75
Our Method + Bi-LSTM	**98.87**	**98.76**	**98.81**

Based on the above SROIE dataset, our final model performed best out of all the existing models, with an F1 score value of 98.81%. All the results are tabulated in the table above. Due to the enhancement of the spell checker component, the overall performance of the system is improved. The values achieved in "Our method (Text extraction GAN) + Spell check (Bi-LSTM))" is with precision 98.87%, a recall value of 98.76%, and the F1 Score is 98.81%.

Table 25: Result: VATI dataset, Our method plus spell check.
"Devised by the author".

Entities	Chargrid	NER	GCN	TRIE	Our Method	Our Method + BiLSTM
Code	89.4	94.5	97.0	98.2	98.6	98.6
Number	85.3	92.4	93.7	95.4	95.8	95.8
Date	89.8	82.5	93.0	94.9	95.6	95.6
Pick-up time	82.9	60.0	86.3	84.6	84.7	**84.8**
Drop-off time	87.4	81.1	91.0	93.6	95.4	**95.8**
Price	93.0	94.5	93.6	94.9	95.2	95.2
Distance	92.7	93.6	91.4	94.4	92.2	92.1
Waiting	89.2	85.4	91.0	92.4	93.1	**93.6**
Amount	80.2	86.3	88.7	90.9	93.7	**94.8**
Average	**87.77**	**85.59**	**91.74**	**93.26**	**93.81**	**94.03**

Furthermore, the next step was to check the spell check model's performance and entity extraction performance. Based on this, we say that our model performed slightly better out of all the existing models, and the overall average accuracy value achieved was 94.03%. Only for field price and distance and accuracy was low but gave good results for "Pick-up time", "Drop off time", "waiting" and "amount".

Finally, an experimental evaluation of the proposed model was carried on the company dataset. Here, the left column represents single item field names and to its right is the accuracy percentage of the baseline model, and right to that is the final solution method results with sub-columns such as accuracy, recall precision and F1 score. All the data is tabulated below table as,

Table 26: Result: Company dataset, field-level extraction - Our method + Bi-LSTM.
"Devised by the author".

Field Name	Baseline	Our Method Results			
	Accuracy	Accuracy	Recall	Precision	F1
Invoice No	71	90.46	89.74	95.42	92.49
Document Type	75	92.44	93.47	98.78	96.05
Date	80	92.27	93.27	98.68	95.90

Vat No	80	91.84	85.43	96.87	90.89
Currency	75	92.78	93.74	98.84	96.22
Discount	75	99.91	96.67	99.40	98.02
Carriage	70	99.94	98.97	99.82	99.39
Tax Rate	75	99.47	93.26	98.74	95.92
Tax Amount	75	90.71	85.58	96.70	90.80
Net	75	85.82	81.82	95.64	88.19
Total	85	91.15	92.31	98.36	95.24
Tele	80	91.73	84.81	96.61	90.33
Fax	50	98.10	90.30	98.00	93.99
Website	90	97.56	95.02	99.08	97.01
Email	50	99.32	95.98	99.27	97.60
Bank Account	50	93.41	82.02	96.13	88.52
Due Date	50	99.66	96.23	99.31	97.75
Zip Code	50	99.97	99.94	99.99	99.96
Country	70	99.72	99.76	99.96	99.86
Average	**69.5**	**95.07**	**92.02**	**98.19**	**94.95**

We began the experimentation, and after that, the analysis of the results of the updated proposed system with the baseline model, the following results have been derived. The baseline model has an average accuracy of 69.5%, and the proposed system has 95.95% for the single item field names extraction after the spell checking has been performed. This means that with the help of the Bi-LSTM spell checker, the average accuracy improved by 0.48%. The final F1 score value of our system is 94.95%. We conclude that using the selected spell-checking method, the overall system performance has improved further in the case of field-level extraction.

The results for the table level extraction method have been presented in the below table. The baseline model has an average accuracy for the item table as 83% and tax table as 60%. The proposed system with error correction using the Bi-LSTM method shows accuracy for the item table as 90.7% and tax table as 80.28%. This updated system's final F1 score value is 94.12%, and the tax table is 90.06%. Therefore, there is an additional increase in accuracy in the case of item and tax tables extraction.

Table 27: Result: Company dataset, table level extraction - Our method + Bi-LSTM.
"Devised by the author".

Field Name	Baseline	Our Method Results			
	Accuracy	Accuracy	Recall	Precision	F1
a. Item Table					
Product Code	70	91.4	95.21	94.71	94.96
Description	85	91.97	95.96	94.68	95.32
Quantity	80	89.68	93.2	94.59	93.89
Unit Price	75	89.63	93.17	94.56	93.86
Net Amount	85	89.81	93.34	94.58	93.96
Tax Amount	85	89.81	93.34	94.58	93.96
Tax Rate	85	89.71	93.21	94.59	93.89
Tax code	85	89.02	92.41	94.55	93.47
Total	90	89.61	93.16	94.55	93.85
Discount	90	90.06	93.61	94.61	94.11
Average	**83**	**90.07**	**93.76**	**94.6**	**94.12**
b. **Tax Table**					
Net Amount	60	80.15	88.54	91.51	90.0
Tax Amount	60	80.66	88.98	91.59	90.27
Tax Rate	60	80.04	88.36	91.54	89.92
Average	**60**	**80.28**	**88.63**	**91.55**	**90.06**

The cost optimization analysis was performed with the existing software and manual data extraction techniques. For the same 500 invoices, the manual entry took roughly 40 hours, and the existing system took 25 hours. Our methods were able to reduce the required time to 20 hours. After performing with the enhanced system, the proposed system further reduced the required time to 19 hours. Furthermore, the proposed system took between 6 and 8 seconds to extract corrected text automatically depending on image quality, size, and the amount of data. The enhanced spell check method led to a small increase in time.

4.7. Contribution

This chapter contributes to the non-word error correction, where the comparative study of different methods has been carried out as explained in section 4.4 to achieve the best output. The solution methodology is used to reduce the error rate, as mentioned in the research gaps in chapter 2, section 2.5. The proposed method that BILSTM gives good results as compared to other methods shows in section 4.6. In this chapter, the following research objectives have been accomplished:

3(a) To study and investigate the role of text correction and enhancement in the decision-making process of the system.

3(b) To identify and apply the best methodology with respect to invoice data extraction.

3(c) To validate the optimization in accuracy, cost, and speed at which the model works.

4.8. Need for Further Enhancement

Our future work will be to extend the spell checker to detect merged word errors and real (context-sensitive) word errors <Chan, 2016 #1997>, which also deals with a better approach to candidate generation where we can use the character confusion model. The character level confusion model was used to learn how a chosen OCR device corrupts characters and the sequence of characters. A character-level confusion model was trained on a specific device or system (Marovic et al., 2010). To detect merged word errors, we will be using word splitting. The basis of our approach to word splitting is a combinatorial search algorithm. It searches the word token, splitting it between each character and keeping the N best results of such splits. The probabilities of the resulting splits are calculated using a statistical language model. For the effective search, we will be using the TRIE data structure. As a real-word error detection tool, both a classic word trigram approach and a mixed part of the speech trigram approach was used. Confusion sets made

up of phonemes, word lengths, word permutation, and some self-defined confusion sets were used in the suggestion process (Tsz Ching Sam, 2016).

In the future, we will focus on improving the performance of our existing Bi-LSTM model in different languages instead of focusing on only English text correction. Still, locale support is a challenging task for invoices that are shared across geographical boundaries. The existence of multiple currencies and addresses becomes a challenge in identifying a locale that needs to default for a given invoice. So, in future, we will resolve the locale challenge.

4.9. Conclusion

From this chapter, we can conclude that the Bi-LSTM approach was better at correcting words than Levenshtein Distance and character bigram. This leads us to an understanding of our results and of the various errors corrected by Bi-LSTM. We can also conclude that it would be better to merge the two systems to increase both the speed and accuracy of our spell checker in the future. We showed a sample screenshot of various parts of our text extraction to correct. First, how we are extracting the text from the images. We have shown sampled correction mechanism to demonstrate the working of our spell check. The screenshots cover the image to the text conversion process, the OCR generated text and the corrected output file with highlighted correction. Throughout the analysis part, we analysed and gathered the approach required to correct merged word, real-word errors and the workflow mechanism required to integrate Levenshtein Distance and n-gram, Bi-LSTM for OCR error correction that we will be discussing in future scope.

We analysed how the OCR was generating text for the image text where the image quality or resolution was not good. Due to which the resultant output text files contained errors that were to be corrected. The OCR error types were discussed, which were briefly classified for the spell checker errors. The spell checker was broadly classified into non-

word (spelling) errors and real word (semantic) errors. The focus of our spell check was to correct non-word errors as real-word errors were not within the scope of this thesis. We neglected the real-word errors as real words errors are mostly checked for handwritten documents images or user typed document images, which was not our case. This helped us to use all the different methodologies to correct the non-word errors. All the methodologies were applied to find the most accurate approach which best fit our case when we had a custom dictionary. Various methodologies have been researched for correction of the non-word errors, and the most famous Levenshtein Distance, Bi-LSTM and n-gram has been selected for correction at different phases.

The first method consists of correcting non-word errors with the Levenshtein Distance. For this approach, we calculated a threshold for what constitutes an error word and searched in the dictionary to correct the errors by IDS (Insertion, Deletion, Substitution) methods. It is used with BK trees to reduce the time complexity of Levenshtein Distance as BK tree search time was similar to DFS traversal. The first word found is replaced by the error word. We got the 82% of correction accuracy for the Levenshtein Distance with the BK tree approach.

The second method consists of correcting the error words by creating a character bigram of the error word, which has been checked against a set of character bigrams of dictionary words. The intersection of the set of character bigram of error words and the dictionary words was used to calculate the maximum similarity coefficient using Dice Coefficient. Then the words with maximum similarity have been replaced by the error word. We got the 89% of correction accuracy for the character bigram approach.

We can conclude that approaches like Levenshtein Distance, BK trees, and n-grams are all viable (despite their drawbacks) for finding the most accurate use case. We chose the best methodology for our case, keeping dictionary size small so that search time can be reduced. However, again, based on further research work, we analysed the different

approaches such as LSTM and BILSTM methods which showed better results than these two methods.

Finally, the third method of this chapter uses machine learning research related to existing spell-checking methods. The primary purpose of designing this model is to overcome the limitations of existing rule-based methods, such as it only gives output based on matching of words; it fails in case of different word length and findings of error in case of a long sentence. So, we implemented a model which is resolving all the above challenges. Based on the study, we conclude that by using the Bi-LSTM, data is checked from sequence to sequence, which shows better results. We got the 91% of correction accuracy for the Bi-LSTM based approach. We concluded that Bi-LSTM is a better approach for spell correction and automatic replacement as it can match substrings for the better-filtered result, which is not the case with Levenshtein Distance and bigram. The Bi-LSTM have better off when the Levenshtein Distance and character bigram between error word and correct words were more significant than 3. It is better with results that had extra characters at starting of error words. The approach is language independent as it is easier to make n-grams for pattern matching changing corpus or dictionary and then to use Levenshtein Distance.

The overall results of the proposed method (Our method (Text detection and extraction GAN) + Spell check (Bi-LSTM)) in terms of average accuracy and the different performance measures are evaluated. The results are tested on three datasets, namely real-world company invoices, SROIE, and VATI. We got better results for the datasets than that of existing methods. The overall average accuracy for the VATI dataset is 94.03% in terms of named entity extraction. Similarly, for the SROIE dataset, the different performance measures are calculated, such as precision, recall, F1 score, and the value of F1 score is 98.81, which is the highest than that existing model which we had compared. The baseline model has an average accuracy of 69.5%, and the proposed method has 95.95% for the single item field names extraction after the spell checking, which means that with the help of the Bi-LSTM spell checker, the average accuracy improved by

0.48%. The proposed method, which uses the data extraction model with error correction Bi-LSTM method, shows accuracy for the item table as 90.7% and tax table as 80.28%. The F1 value of this proposed method is 94.12% for the item table and 90.06% for the tax table. Now proposed method that is with spell check method, which further reduced the time to 19 hours. Also, the time taken by the proposed system to extract corrected text automatically is around 6-8 seconds based on invoice/ receipt image quality, size, and amount of data.

5. Conclusion

Invoice data extraction has gained considerable momentum in research and industrial contexts. Every business wants to automate the accounting or expenses management process. This helps them to optimize the financial supply chain process for the improvement of the business. Even an individual is interested in maintaining day to day expenses using some automated tool. Due to the size of transactions that are taking place and the number of invoices that are getting generated, it is becoming a significant challenge to process, maintain, and organize expenses. Issues exist related to the input images in terms of a variety of formats, different languages, printing issues, different image sizes and qualities. Even if the text is extracted, the nature of data that is to be converted from unstructured to structured format does not meet the expectations. Therefore, the accuracy of data extracted and the time taken to extract and validate the correctness of data plays a significant role in cost optimization. This thesis has done an exhaustive literature and survey review to identify the challenges. Later, it proposed a novel invoice data extraction system that is based on machine/deep learning and a unique rule-based engine. The study was performed using the existing dataset such as SROIE (SROIE, 2020), VATI (VATI, 2021) and data received from the company.

The system was enhanced with advanced spell checker implementation. This was based on the limitations found in the proposed system. The literature review related to the spell check was done. Different solution methodology like Levenshtein Distance (Levenshtein, 1966) and n-gram's string similarity and Bi-LSTM model was tested and compared. We concluded that the Bi-LSTM model is a better approach for spell correction and automatic replacement as it can match substrings for the better-filtered result, which is not the case with Levenshtein Distance and character bigram. The Bi-LSTM are better off when the Levenshtein Distance and character bigrams between error word and correct words are greater than 3. It has better results for the extra characters at starting of error words. The approach is language independent as it is easier to make n-grams for pattern matching

changing corpus or dictionary and then to use Levenshtein Distance. The main purpose of designing this model was to overcome the limitations of existing rule-based methods, such as it only gives output based on matching of words.

The overall results of the final framework (proposed rule-based method (Text extraction GAN) + Spell check (Bi-LSTM)) in terms of average accuracy and the different performance measures were evaluated. The average accuracy for the VATI dataset was 94.03% in terms of named entity extraction. Similarly, for the SROIE dataset, the different performance measures were calculated, such as precision, recall, F1 score, and the value of F1 score was 98.81, which is the highest among any existing model which we included in the comparison. The baseline model has an average accuracy of 69.5%, and our proposed method has 95.95% for the single item field named extraction after the spell checking, which means that with the help of the Bi-LSTM spell checker, the average accuracy improved by 0.48%. We have also done the overall analysis on real-time company datasets. The proposed method, which uses the data extraction model with error correction Bi-LSTM method, shows accuracy for the item table as 90.7% and tax table as 80.28%. The F1 score value of this proposed method for the item table is 94.12%, and the tax table is 90.06%. (Chapter 4, 3(c) RO achieved here)

Ultimately, this thesis has made substantial theoretical and practical contributions. From the results, we can conclude that the novel system's performance is the best among systems currently being researched and used in the industry. The system was validated by an interested company and accepted. Additionally, the system reduced the average processing time of 500 invoices from 25 hours to 19. It also reduced the processing time to automatically extract corrected text to around 6-8 seconds depending on invoice image quality, size, and amount of data. This is a significant reduction in time that can drastically change SCM and FSCM workflow if widely adopted. Error reduction, cost optimisation, and spending are areas where this research has the potential for practical impact. In terms of impact on future research, this study has contributed by setting a new benchmark for data extraction efficiency and accuracy.

5.1. Benefit in Spend Analysis

One of the outcomes of an automated invoice system is a better classification of spend and supplier selection. One of the research objectives was: 4(a) "To evaluate the significance of supplier selection and spend classification." During the interview and survey process performed on a company who were interested in spending data analysis and cost optimization (a mechanism by which automated classification could help various industries reduce their costs was realized), the significance of invoice data automation was studied and evaluated. This study helped understand the use of expenses management systems in different industries, especially in the context to spend classification and supplier selection. The details of the study have been provided in appendix section III. The study shows how the cost of procurement of items from different suppliers can be further reduced by doing an in-depth spend analysis. This is done by finding the exact category of expenses done for each transaction. Later, based on this, the selection of suppliers can be further optimized. The reports include sample data provided by the company and evaluated the cost optimizations process.

5.2. Application in Other Domains

In this section, the 4(b) *"In other application areas."* and 4(c) *"Finally, to develop a self-adaptive system for any kind of invoices across the industry"* research objectives are accomplished. Now, this is already validated that automation in any industry and sector helps optimize a problem and reduce cost too, if not immediately but indeed in the longer run. It might require an initial investment, but in due course, it is considered a good investment due to an increase in revenue and sales. Furthermore, due to ongoing competition among various players in the market, everyone wants to make use of machine learning and AI-related solutions and techniques. Most of the time, the decision-makers

are not aware of how a whole or part of the system can be optimized? Sometimes they hear from others about advancements in technologies, therefore deciding to do something similar in their business or are approached by the consultancy service provider to provide such a solution. Whichever way, understanding the problem and providing a solution is not a simple process. Many a time, a readymade solution does not help; further, customisation is required. After several surveys and talks with many industry partners and leaders, we manage to discuss, identify, and optimize a section of the problem by using our work done in this research. The target area mainly was related to invoice and expense classification.

In simple terms, extracting text from invoices for automation and future requirement is easy as we have already created a system. Understanding where this can be applied is more complicated. The following section tries to identify and apply text extraction in different contexts to validate its importance. This is similar to the case study, which presents the existing system, the proposed system, and a discussion of how the proposed system helped them enhance the business's productivity. The proposed model is helpful in the case of key-value pairs as well as pattern matching. Based on this, it has the following applications:

1. **Banking**

A check can be scanned, and the printed amount can automatically be transferred directly to the destination account with no manual entry. This technique is perfect for printed checks but still works well with other bank documents, such as handwritten checks (Rastogi et al., 2020). To quickly build a client event, the banking industry is increasingly using invoice data extraction systems to store client-related paperwork such as onboarding content. It reduces the onboarding time considerably, thus helping to improve the user experience. Additionally, banks often use the invoice data extraction method to extract details from checks such as account number, level, and check number to speed processing. The text extraction tool can extract data from checks to collect account

information, handwritten dollar amounts, signatures, and credit transactions, containing numerous documents. The text extraction tool is used to capture the payslips of credits and debits (Rastogi et al., 2020).

For insurance, asset processing can be automated by the text extraction tool and supporting technologies. In digitalized receipts, table extraction (Kim et al., 2020) is useful by automatically detecting the tables in a document, getting text in each cell, column headings for research, data entry, data collection, etc. As some stakeholders commented in the survey feedback as,

- *"Business bank receipt, contract and agreement, tax filing documents, medical"*,
- *"Making documents searchable" and*
- *"Many avenues including statements etc."*
- *"The invoice text extraction helpful in the Trade Contracts, Master Agreements, Legal Documents for trade and dealers."*

Based on this, we conclude that invoice data extraction may be helpful in mentioned fields, such as, in banking to search the documents, statements, and filling of data in various documents. Furthermore, based on the feedback analysis of the invoice extraction, the user commented:

> *"There should be a tool, if a new billing is done by the card, (I mean the app is on in the mobile and I am doing a bank transaction from my card notification saved in the wallet of the phone) it should automatically grab the data and notify me about the recent transaction and ask me to add the bill or not."*.

Furthermore, the finance industry requires feedback for better analytics of expenses, for example, "*Parking Receipts"*. It is helpful to collect the information related to a person who is using what type of vehicle and keep the record for future use. As per the other user review, they said that *"More customer reaches the invoice text extraction"*, *which* means

it is helpful at the time of online marketing, we can quickly contact thousands of people with one click only, and it is possible using the digitisation only. As per the feedback carried out, one stakeholder has commented that

> "*In financial markets, there is a big shock to the world of trade where LIBOR is phasing out, and new IBOR will be applied to existing contracts so, if the software is able to gauge the usage of LIBOR in the contracts like Pricing, Collateral Payment, Valuation etc. If that happens, this could be a huge win. Currently, all that is done manually, and it takes huge human effort.*"

It is, thus, helpful in the field of banking, finance, business or rather everywhere we want to store income-related information.

> "*Bill Scanning and Expense Management classification. We can count expenses as well as income. Definitely, we don't need to invest valuable time to extract data"*. As this is commented by another stakeholder in survey form.

2. Healthcare

In terms of real-life applications, the model can be fine-tuned to work for similar automatic text extraction applications. For example, in the medical field, extracting printed text that is medicine names from a doctor's prescription is a tedious task. The model can be developed for reducing time-consuming tasks for the pharmacy to check the availability of medicines by automatically comparing the prescribed medicines within hand stock. The model can be developed in other areas, such as the automated reading of orders in warehouses for enhanced retail management experience and many more similar scenarios.

Employees of a healthcare organisation deal with many forms (insurance, healthcare-related forms) for each patient. We can extract information from such forms and put it

into a database. By promptly recording the data of every patient, the healthcare providers can focus on improving the service. OCR can scan reports that accommodated X-rays (Liu et al., 2019a), previous diseases information, treatments or diagnostics records, tests, hospital records, insurance payments which will be helpful to made searchable for the convenience of doctors.

Based on the survey feedback that *"Biomedical, business applications, NLP" and "In healthcare to store the patient's information and medical reports."* so the invoice data extraction might be helpful to in biomedical as well as in business to save their records.

One more feedback item related to the healthcare domain was,

> *"In our healthcare business, we get many Bills every month through the mail for Electricity, Water, Taxes and Invoice / PO for vendors. If the system is able to automate the process of going through each and every bill or invoice and extract the information with a min 90-95% accuracy, then it will save a lot of time and money. Apart from this, we also get reports sent by patients on mail where we need to manually go through the values of each test done and suggest medicines based on the test value. Many times, we also need to manually go through the previous health records of patients. The chances are that we can miss some important information. If we can automate the process and extract all the important information with accuracy and present it before the doctors in the required format, that will really save time and money."*

3. Manufacturing

Furthermore, smart operations are about developing a technology by which the Internet of Things (IoT) can be able to analyse Big Data in real-time work. From a manufacturing perspective, it can further lead to developing a new approach for improving operation excellence (Zhong et al., 2016). Manufacturers like to transform the existing operation into a smart operation for developing business strategies after detecting advanced

information through Big Data analytics. This leads to improved flexibility in physical processes, but this also leads to an increase in training related to the workforce and the use of new internet technologies and Big Data (Davis et al., 2012; Moyne & Iskandar, 2017; Thoben et al., 2017). In the manufacturing and service sector, "the Big Data analytics is about using a centralised data model for combining structured data like inventory and financial transactions in a business system. This is done with a structured operational system data like process parameters, alarms, and quality events, with internal and external unstructured data like supplier, customer, internet and data through machines to uncover new insights by implementing an advanced analytical tool".

There is a considerable need for new business strategies which improve the existing operations. Smart operations are about improving the operational activity in any manufacturing sector by analysing Big Data in real-time. This will lead to improvement in operation (Opresnik & Taisch, 2015). Such smart operation can also be understood by invoice data classification on different types of invoices related to fuel, machinery, and others. This invoice data classification is a smart operation as it reduces the workforce involvement in classifying such invoices and digitalises the system (Bokrantz et al., 2017). Even spend analysis can be done at a much deeper level. This smart operation is beneficial when a vast number of invoices are involved in the form of Big Data. With developing the analytical tool, predictive maintenance techniques should be involved to predict the information required in Big Data analytics maintenance for improving the maintenance procedure men (Moyne et al., 2016). At present, a better understanding of how global supply chain leaders can co-create financial and non-financial sustainability through data-driven and adaptive leadership (Law & Gunasekaran, 2012; Akhtar et al., 2016).

4. Other Industry

To reduce manual workload in a variety of industries, the digitisation process is used in tandem with OCR and other data extraction tools. Data extraction is useful to store the

information from business documents and passports (Lee, 1998) in the required format. The industry is always coming up with some new tools. Microsoft created an app that takes a handwritten mathematical equation and produces a solution with a detailed description of how it works. In addition, when a client or end-user uploads KYC (Know Your Customer) documents, OCR is used to collect information from them and store it for future use and reference (Pawar et al., 2020). Even to store the ancient data that may be storybooks, letters, or any document converted into the digital text format (Xamena et al., 2019). Information extraction is used by many large initiatives, such as the Gutenberg project, the Million Book Project (Moens, 2006) and Google Books, to search and digitise books and store them as archives.

Furthermore, as per the survey review, the existing text extraction software is helpful in businesses/industries. This software can be further enhanced so that end-users can get a better result from manual receipts.

> *"As the rural clients mostly submit manual receipts. And sometimes, it is challenging to understand the data for the software. They would also like to see if this software can be enhanced for scanning handwritten receipts of local Indian languages. So, this the real-time application to extract the handwritten data from the receipts specifically for the Indian languages."*

Sorting the data manually from the invoice and classifying expenses based on the items present in the invoice is a challenging task. *"To help our clients extract more data from their invoices/purchase orders for classification purposes*.", stated by user feedback. So, with the help of digitisation, we can easily classify the required data. This is applicable in shopping malls, store departments of every organisation and industry.

5.3. Contribution

The thesis's contribution is divided into two parts: Contribution to enhanced invoice data extraction system and correct selection of spell check model for invoice data extraction. The main contribution of the thesis is:

1. This thesis provides a comprehensive literature review related to OCR, text extraction on written language and scripts, real scene images, text from videos, non-invoice documents, invoice related documents in chapter 2 and fulfilled the 1(a) to 1(f) research objectives. The literature discusses the key problems, features and challenges related to invoice data extraction. It discusses different machine and deep learning techniques. Moreover, finally, a detailed review of spell-checking methodologies and model selection in context to invoice data extraction was performed in chapter 4 and accomplished research objective 3(a) *"To study and investigate the role of text correction and enhancement in the decision-making process of the system."*.
2. The applicability of the proposed solution was investigated. We implemented a novel rule-based algorithm in chapter 3 of section 3.4, with the help of machine learning techniques which satisfies the research objectives 2(a) *"To understand and apply various machine learning and pattern recognition techniques to increase accuracy."* and 2(b) *"To develop a novel method that works very well on any generic invoices."* The uniqueness of data extraction and block identification from XML is a state-of-the-art solution provided in this thesis. The XML pre-processing includes XML structuring analysis, document modelling, nearby blocks mapping and position to text mapping for best template matching representation. This ensures that the block is well arranged, and the data extraction process can be performed easily.
3. We have achieved the 2(b) *"To develop a novel method that works very well on any generic invoices."* research objective in chapter 3 of section 3.4, by

considering the implementation of its OCR engine for text detection, a Generative Adversarial Network (GANs) based model is used. The GANs based models are designed for image-to-image translation, image enhancement and which shows very significant results. The UNET model gives good results in the case of scene image text detection. However, they were inspired by object detection using the Faster R-CNN, which gives good results in the case of small object detection. So, we implemented the generator model with the help of the Faster R-CNN model, and the discriminator model uses the PatchGAN. The thesis did thorough research in selecting the best methodologies specific to invoice data extraction. It was further added with unique method selection for text cleaning, spelling checking and document classification before text blocks are finally prepared for information extraction.

4. We have contributed in chapter 4, the section 4.4 analysis between the bi-gram, Levenshtein Distance and machine learning based on the Bi-LSTM model to get a better selection for spell correction method as well as satisfied the 3(b) *"To identify and apply the best methodology with respect to invoice data extraction."* research objective. We concluded that character bigram was a better approach for spell correction and automatic replacement as it matched substrings for the better-filtered result, which was not the case with Levenshtein Distance. The character bigrams were better off when the Levenshtein Distance between error and correct words was more significant than 3. It was also better with results that had extra characters at starting of error words. The approach was language-independent as it is easier to make n-grams for pattern matching changing corpus or dictionary and then to use Levenshtein Distance. The Levenshtein Distance was fast but could not automatically replace the best-fit words according to a calculated threshold value. Although it was able to replace words, they were not the best first suggestion. This was the major problem as the replacements were not correct for some words even though they were in their suggestion list. It was also unable to

suggest words when they had some extra characters or symbols at the end or start of words. Though it is a problem in n-gram also, it is better than Levenshtein Distance. Then, we initiated the spell-checking using the Bi-LSTM model. The main purpose of designing this model was to overcome the limitations of existing rule-based methods, such as it only gives output based on matching of words; the RNN is used to solve sequence to sequence learning problem is that it is not clear what strategy to apply when the input and output sequence does not share the same length, it fails in case of different word length as well as findings of error in case of a long sentence. So, we implemented a model which is resolving all the above challenges.

5. We have made the survey feedback of different stakeholders, those using the existing market tools/ software for their business, professional studies, and many other applications to extract the text. We have represented the stakeholder's response in various chapters like in chapter 2 of section 2.5 to get better results which maybe not be included in earlier research papers and achieved 1(f) "*To further perform an industry survey to identify the importance of automated data extraction*" research objective. The limitations and future scope to improve the existing tools are listed based on the survey. The detailed comments section of the survey feedback form information is given in Appendix II.
6. The contribution in the form of enhancement of accuracy is worth noting. As the results are discussed in chapter 3 of section 3.5 and chapter 4 of section 4.6 and fulfilled the research objectives 2(d) *"To validate the optimization in accuracy, cost, and speed at which the model works."* and 3(c) successfully. The newly developed system outperformed all the data sets used in this thesis. Even it helped to optimized human cost in the supply chain management. The start of the art product created in this thesis can be developed further to make it industry-ready. Therefore, massive potential in terms of business and revenue. Few stakeholders have already shown interest in the application, which is discussed in the previous

section of this chapter that is in 5.1 and accomplished 4(b) *"In other application areas."* and 4(c) *"Finally, to develop a self-adaptive system for any kind of invoices across the industry."* research objectives.

5.4. Limitations

The newly designed machine learning-based invoice text detection and extraction system is appreciably different from the existing approaches studied in the literature review. The implemented model has been successfully tested on the data sets such as SROIE, VATI and real-time company datasets. The model showed outstanding results in the case of both text detection and extraction. The required information is extracted from the invoices successfully. Even though the proposed machine learning-based invoice text detection and extraction system has shown a significant contribution in the case of the real-time application, still the model has certain limitations that could not be completed due to time constraints. These limitations are explained as follows:

1. We tested our algorithm on the English invoices, where it demonstrates outstanding results. The model's success is limited when it comes to non-English receipts with handwritten text (Geetha et al., 2020) due to the data block combining problem, as discussed in chapter 2 of section 2.5 (11). In the data block, words are combined by nearby spaces to form a data block. Tagging of these data blocks with field names is done. Due to a lack of knowledge and proper configuration, it fails for some field detection. There is also a possibility of two or more field data that has been combined as a single data block. For this, the value does not get extracted as expected because tagging of the field will cause a problem here (Rahal et al., 2016b). If two-word blocks are far away in the image, then spaces must be considered while making them as a line. Even due to printing issues (Jun et al., 2019), some blocks end up getting

combined. Still, we are facing challenges in the case of handwritten text and documents with a printing problem.

2. The performance of the proposed system is less due to the complex network architecture. We implemented a GAN-based model with more learning parameters and, like the Faster R-CNN network, has a high computational cost. So, there is a need to work on performance for faster execution.
3. Sometimes the invoice images are captured with high-resolution cameras (Zhu et al., 2019), scanned with low resolutions (Rabbi et al., 2020), and many other aspects related to the quality of the image. At the time of pre-processing, the quality of the original image may get degraded at the time of illumination conversion. The GAN model should be explicitly optimised to invoice images for quality improvement. Further, there is an improvement required to resolve the problem of image quality.
4. Research would need to be demonstrated to explore non-invoice data extraction. The scope of the research was limited to only for English invoices to prove the concept, even though the solution design is dynamic enough to extend it to financial documents and non-financial. The solution would thus be applied to non-invoice images or other documents to get real results. (Chapter 2 section 2.5 (5) Variance in Business Requirement)
5. The implemented model has a limitation if the invoice image which has two or more receipts in a single image (Rahal et al., 2016a). It considers it as a single invoice, thus detects and extract information according to that. Furthermore, the research needs to be extended towards splitting the images if it contains more than one receipt or invoice data in a single image, which will help extract necessary information instead of losing it (Zhang et al., 2009).
6. The following table denotes what all field was successfully extracted. The pattern which was supported and the pattern which is still limited by the system. Further works need to be done on columns that define the limitations. Machine learning is a need that will help automatically define the invoice template; thus, data extraction

will be much accurate. The table-level extraction encounters the problem due to the overlapping of table columns which creates a barrier to find boundaries of the columns; the content presented differs in size and font from invoice to invoice, or simply the structure of the table is different for every invoice (Reza et al., 2019). The model sometimes fails in the case of the table without borders (Gilani et al., 2017). So, we will work on the table level extraction in future. The application of machine learning techniques can enhance data pre-processing, field mapping, and dependency mapping. Due to the stipulated time of research, we have been unable to solve every challenge faced by researchers. Still there is a scope to work on all those issues.

Table 28: Overall Limitations table.

"Devised by the author".

Sl.	Field Name	Supported	Limitation
1	Currency	1. All the values of the Currency list provided by the test team should be found. 2. If currency value was not OCRed properly, then the locale-based currency will be returned as the default	The currency that contains special symbols (other than alphanumeric) cannot be read right now due to regex limitation. PEN-S/. Some currency symbols cannot be read, e.g., for countries like AED, IQD etc.
2	Total, Net, Tax Amount, Tax Rate, Discount, Carriage	• Amount following vicinity words. (amount resides next/right side to vicinity) • Found below of vicinity words. (amount resides down/next line to vicinity) • If no amount is found by vicinity words, then get the max amount of the whole receipt/invoice as the total amount. *Assumptions-* *Get the first record from the list of sorted amounts in descending based on the index in the text.*	The amount can be found as Total by few other similar keywords of defined vicinities. e.g., in the image, it was mentioned as-- Grand total: 1234.00 to pay after a discount of 123.0 then receipt/invoice total will be returned as 1234.00
3	Invoice Number	To extract invoice number found as next/below of vicinity. 1. Only numeric values. 2. Numeric with or without special symbols like space,' /' ,' _'. 3. Values started with string values (of length2-3 chars) e.g., IN-345667 *Assumptions-* *the finalised value should contain at least 2 numeric characters, discarding if not.*	Most of the invoice issue is due to the requirement of INV-******* (or few other preceding words followed by invoice number) formats. Also, since the key is either "Invoice" or "Invoice Number", so, in regex, some character needs to be ignored before next digit-based starts. if say 'Number' is not read correctly even if present, then it creates an issue.

			If not found as next right to the vicinity, then we are looking for downside values of that vicinity, and due to the alphanumeric format of invoice number, it is hard to tell if the value extracted in this manner is not invoice id.
4	Invoice Date, Due date	"ddth mmm, yyyy yyyy-mm-dd dd-mm-yyyy dd-mm-yy mm-dd-yy dd-mmm-yyyy mm-dd-yyyy" *Assumption-* *1. Finalize max date as a result out of the extracted list (either by vicinity or without vicinity).* *2. The finalised date should not be more than 5 years old.*	If OCR does not read the invoice date. Too bad to read. Then some other date might be returned.
5	Bank Account	value for Bank account has been supported by a range of 5 to 15 characters long. It can be numeric only or numeric with special symbols like space,' /' ,' _' e.g., bank Ac 23320553 Acc No NH161969/000 A/C No VIBA01 *Assumptions-*	The bank account will not read if the vicinity (e.g., Account, bank acc) is not read correctly.

		the finalised value should contain at least 2 numeric characters, discarding if not.	
6	Telephone, Fax	Value for Telephone and Fax has been supported by a range of 5 to 15 characters long. It can be numeric only or numeric with special symbols like space,' /' ,' _' e.g., 01782 336351 01437-766528 44(0)20 7232 3010 *Assumptions-* *the finalised value should contain at least 2 numeric characters, discarding if not.*	
7	Vat Number	Value for the Vat Number has been supported for the range of 4 to 15 alphanumeric characters or special symbols like '-' '/' '|' '_' e.g., GB151629570 190379585 IE 634 696 7G GB697-8707-56 *Assumptions-* *the finalised value should contain at least 2 numeric characters, discarding if not.*	
8	Tax Rate	Value is supported for the range of 0 to 3 integer digits with fractional values. e.g., 20.00 0.00 13.00	Rate is not supported if the value does not have a fraction part. E.g., vat@20
9	Email	All valid email format has been supported.	

		e.g., sales@spraytanpro.co.uk abc@company.com	
10	Website	All valid Website format has been supported. e.g., www.shearwell.co.uk http://www.netmums.com	
11	Country	Five countries have been supported till now UK, US, NZ, GB, IE. More country matching words required from the invoices for better accuracy. *Assumption-* *If country not found by matching words currently trying a combination of currency and zip code format to guess it correctly. (currently, implemented this for the US only).*	
12	Zip	3 formats of zip code have been supported till now. Which are For the UK-- S70 1YA EG4A 4TR For the US-- CA 12345	

5.5. Future Research Work

The proposed model should be explored for the following points.

1. We tested our algorithm on the English invoices, where it shows outstanding results. Instead of focusing on a single language, the robust algorithm should be open to any kind of language (Dave et al., 2020).
2. The challenging task is performance due to the complex network execution takes more time. So, there is a need to work on performance for faster execution. The GAN model should be explicitly optimised to invoice images for quality improvement.
3. The table-level extraction encounters the problem due to the overlapping of table columns which creates a barrier to find boundaries of the columns, and the content presented differs in size and font from invoice to invoice or simply the structure of the table is different for every invoice so that we will work on the table level extraction in future.
4. Research would be demonstrated to explore the non-invoice data extraction. Although the approach design is dynamic enough to apply to financial and non-financial documents, the scope of the study was limited to only English invoices to prove the idea. To get the real results, the solution must be extended to non-invoice photos or documents (de Jager & Nel, 2019).
5. Furthermore, the research will be extending towards the splitting of the images if it contains more than one receipt or invoice data in a single image, which will help to extract necessary information instead of losing it.
6. Furthermore, we will do a study on future forecasting and automation of the model, which will be helpful for the business. The businesses include the banking sector, healthcare, and industries. Additionally, we will perform surveys to extend the model to other sectors.

Appendix I: ADE System Manual

This thesis involves the implementation and creation of a new end to end system. The following sub-sections will describe the software and its usage.

1. Portal Configuration

The system consists of a web-based portal that requires login credentials. After the portal is loaded, there are multiple menu items on top of the page, as shown below. Some may not be visible due to different access levels.

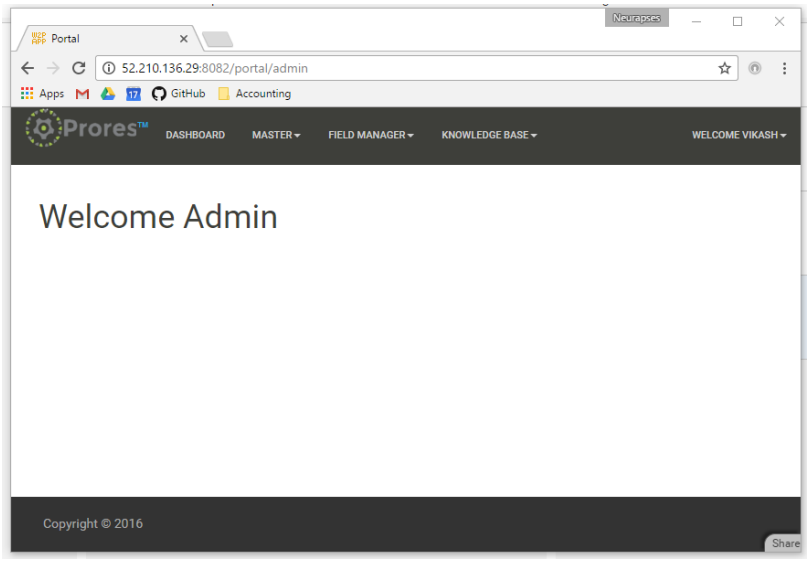


Figure 62: Main admin portal page.

"Devised by the author".

Each view/screen which is part of the menu has the option to edit/create records. With on click of a link in the menu, the following similar screen will come. Let us say, on click of *Click Master > Configuration*, it will show the following screen.

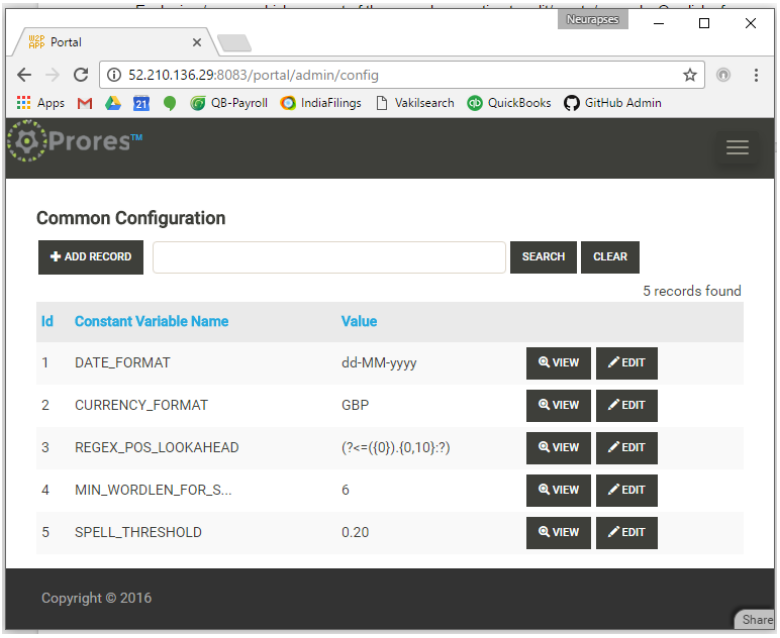


Figure 63: Sample screen.

"Devised by the author".

In this, each following function can be performed.

A. **Add Record**: To add a new record

B. **View**: To view line items in detail

C. **Edit**: to edit individual records.

D. **Search**: If the list is extensive, the record can be filtered using the search button.

Overall, the master menu contains one-time configuration settings. Other menu items are explained as below:

1. **Field Manager:** These are internal and almost one-time configurations. Changes are made once only unless new fields are planned to be added. This screen contains:
    a. Field, Item and Tax table links: It maintains a list of fields and items that need to be extracted.
    b. Plugin: It maintains a list of plugins that are already implemented for data extraction.
    c. Field to Plugin Mapping: Now, once the plugin's implementation is done and entry is added to the plugin manager, the field to plugin mapping is done here.
2. **Knowledge Base:** This menu contains entries related to data extraction.

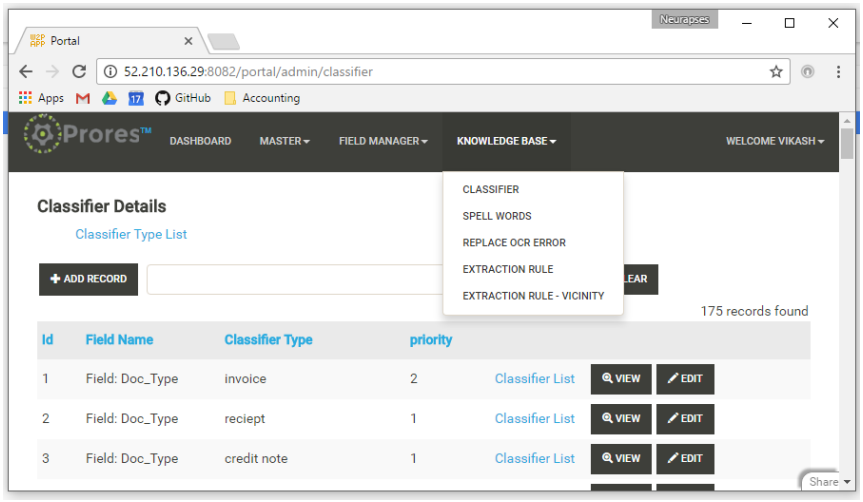


Figure 64: Knowledge base menu.
"Devised by the author".

Details for each submenu item:

a. **Classifier:** Fields that are based on classification, for example, document type, are maintained here. If we need to classify all words based on the weight that falls into the invoice type category, first select what needs to be returned and return all the words that can be found in bills.
b. **Spell Words**: Add essential words that are misread by OCR.

c. **Replace OCR Error**: Not all mistakes by OCR can be corrected by spell check. Moreover, if OCR misreads any consistent words (or group of continuous words), they are added here. Example ‘m’ is read as ‘iii’
d. **Extraction Rule:** Add/Update regular expression related to fields that are extracted based on regular extraction.

2. Accuracy Test Validation Process

After normal execution of data extraction, we need to update the system with the correct reason for failure, or in other words, validate the accuracy result. To achieve this, the following task must be performed.

Step 1: In portal application, open two windows/tabs for faster entry, for each of the following:

a. In the portal, go to menu ACCURACY TEST > OCR IMAGE CHECK as shown below:

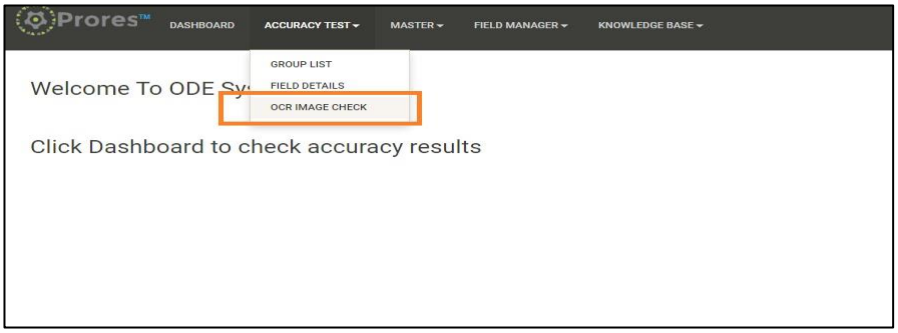


Figure 65: OCR Image check screen.
“Devised by the author”.

b. In Portal, go to menu ACCURACY TEST > FIELD DETAILS as shown below:

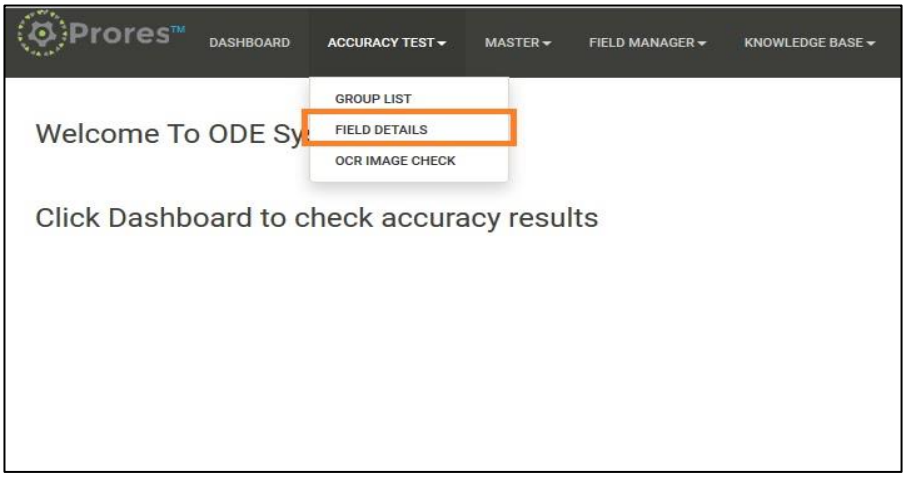


Figure 66: Accuracy Test field details screen.

"Devised by the author".

With on click of the 'FIELD DETAILS' link, the below page will be open:

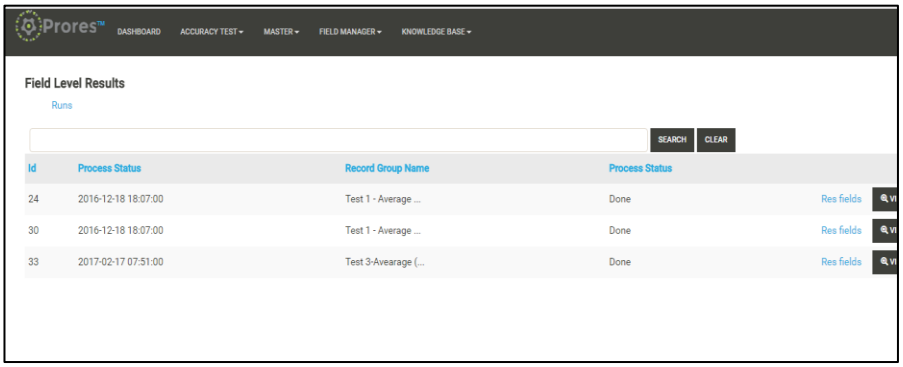


Figure 67: Field details screen.

"Devised by the author".

Step 2: On the **2nd page,** click on the **Res fields** link of the latest executed Id *(currently, i.e., 33).*

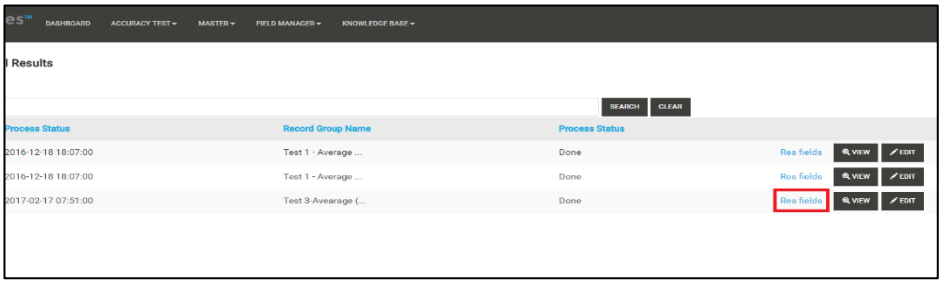


Figure 68: Result fields screen.

"Devised by the author".

Here, records of comparison of extracted vs actual data in the database for all XML will be shown. Select value under column '**XML File**' for a row.

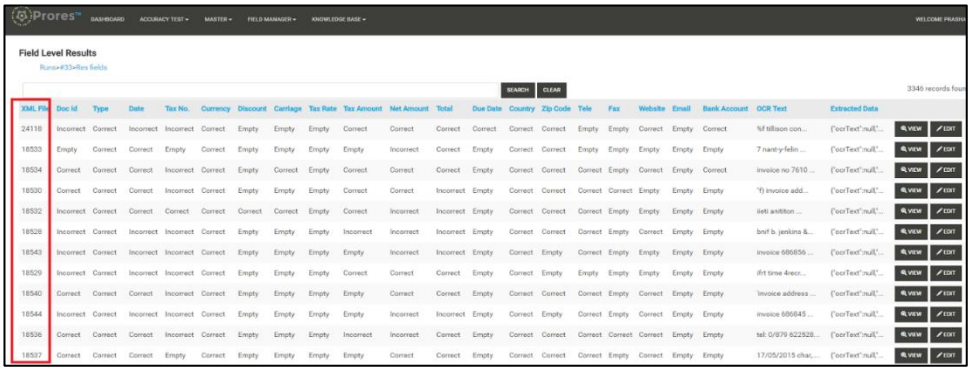


Figure 69: Execute XML list screen.

"Devised by the author".

Step 3: In Portal, go to the menu ACCURACY TEST > OCR IMAGE CHECK as shown below. Enter the XML file number and click submit:

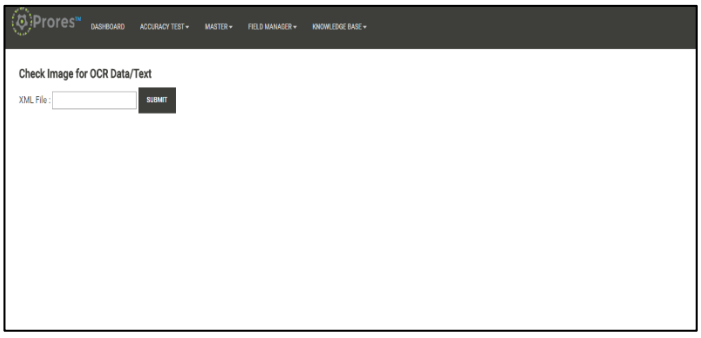


Figure 70: OCR Image check screen.

"Devised by the author".

Output page will be shown here in 2 parts as shown below:

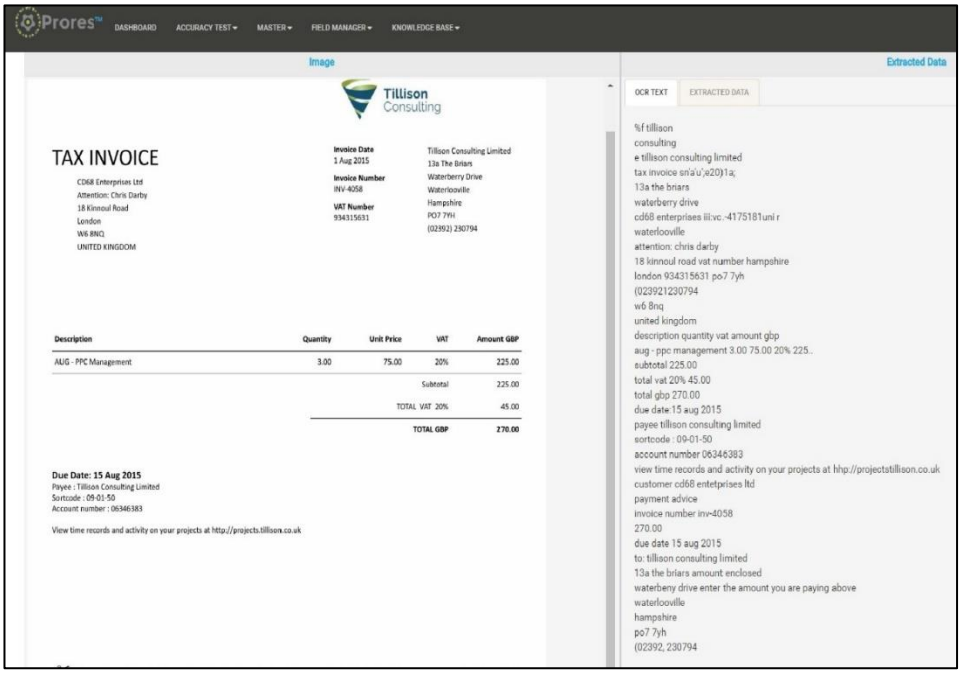


Figure 71: Output screen of the XML result.

"Devised by the author".

The left pane will contain the original image, as shown below.

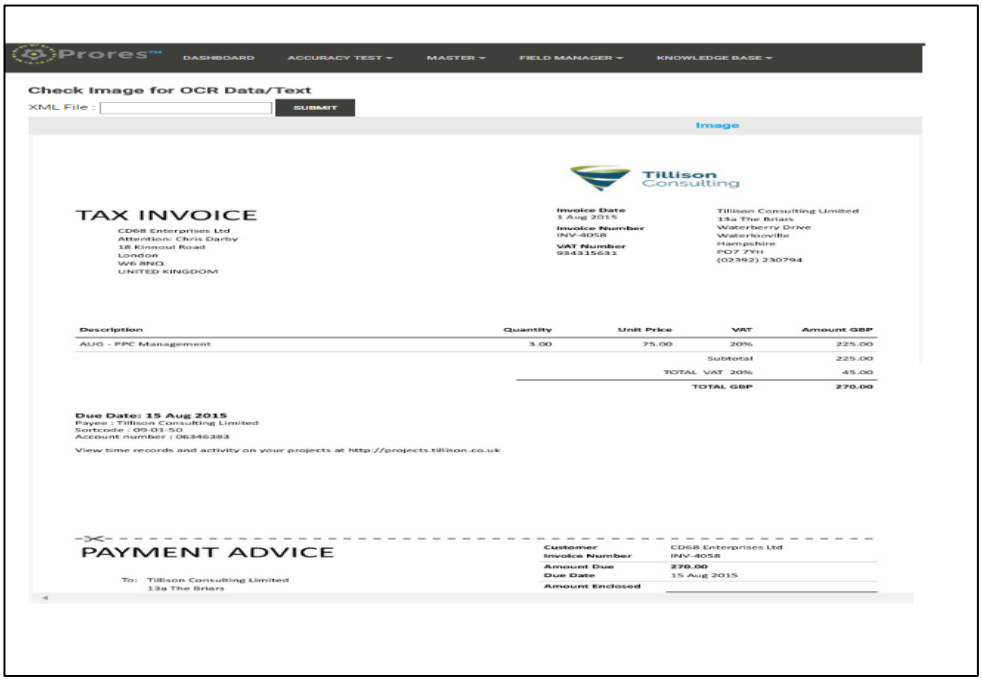


Figure 72: Left pane image view screen.

"Devised by the author".

and **right pane** will have two tabs, one for **Extracted Data** and another for **OCR text.**

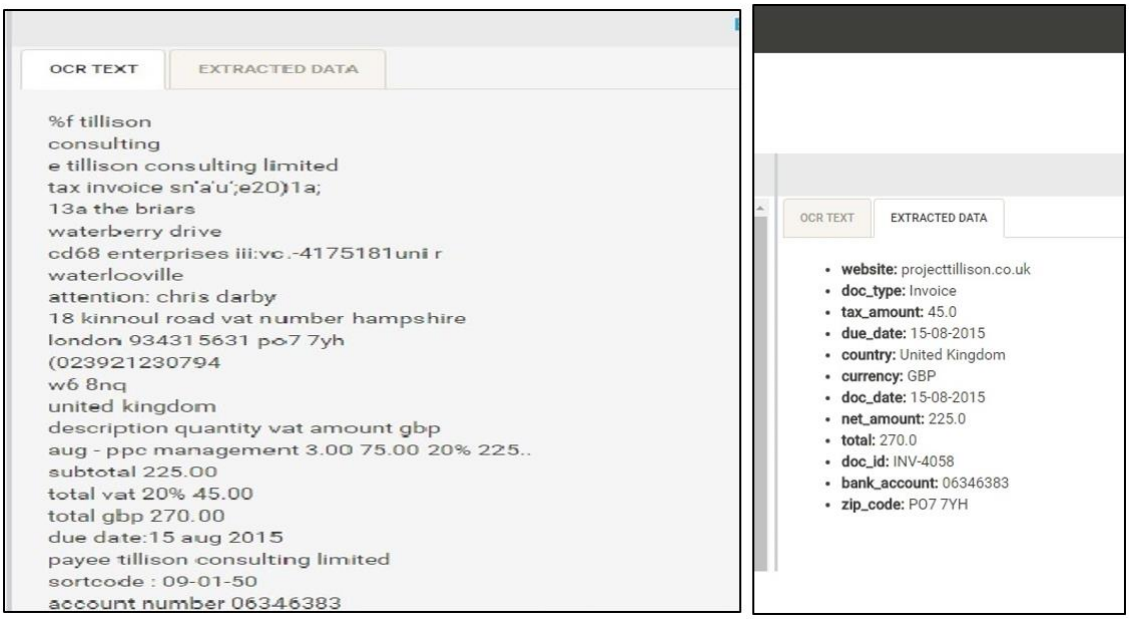


Figure 73: Right pane result screen.

"Devised by the author".

Step 4: To fill field-level data on the 2nd page, we must look at the image, OCR text and extracted data on the 1st page.

Step 5: To fill data on the 2nd page, go to the Menu > ACCURACY TEST > FIELD DETAILS > RES FIELDS (with 33 run id) screen as shown below:

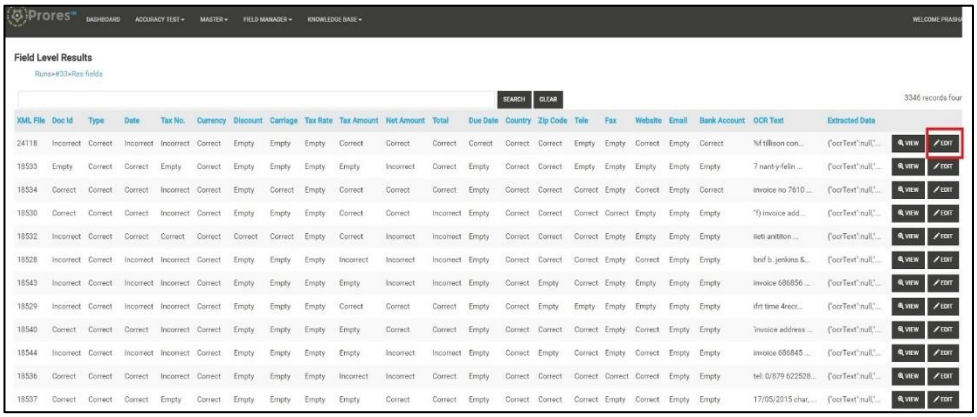


Figure 74: XML files result in the edit option screen.

"Devised by the author".

A drop-down list will be there against each field for multiple reasons.

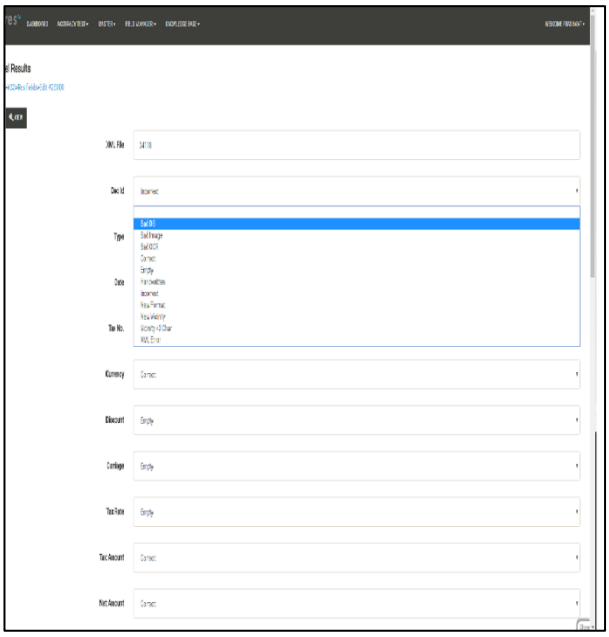
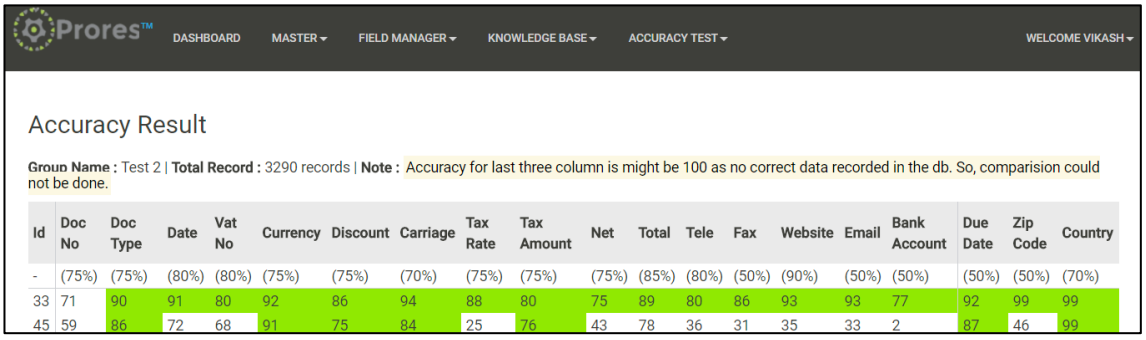


Figure 75: Failure/Pass reason update screen for each record.

"Devised by the author".

Select the most appropriate reason observed in **Step 4**. In case of no appropriate reason found in the drop-down as per observation made in step 4, for each field, select default reason 'Incorrect'. Later, the developer team can verify and add a new reason, if needed. Once this is done against all the images executed within a test run, the dashboard page will show the final accuracy percentage for each extracted field.



Id	Doc No	Doc Type	Date	Vat No	Currency	Discount	Carriage	Tax Rate	Tax Amount	Net	Total	Tele	Fax	Website	Email	Bank Account	Due Date	Zip Code	Country
-	(75%)	(75%)	(80%)	(80%)	(75%)	(75%)	(70%)	(75%)	(75%)	(75%)	(85%)	(80%)	(50%)	(90%)	(50%)	(50%)	(50%)	(50%)	(70%)
33	71	90	91	80	92	86	94	88	80	75	89	80	86	93	93	77	92	99	99
45	59	86	72	68	91	75	84	25	76	43	78	36	31	35	33	2	87	46	99

Figure 76: Accuracy result dashboard.

"Devised by the author".

3. Data Entry Validation Process

Furthermore, a separate enhanced application was created to validate and correct extracted data from the invoice. The image below shows the screen for manual uploading of invoices in the system.

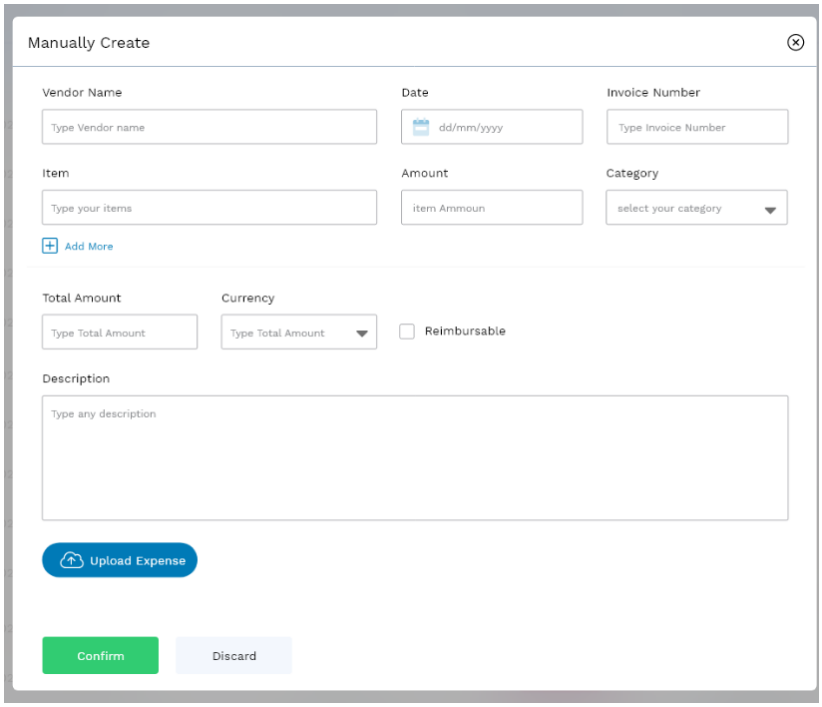


Figure 77: Upload screen for an invoice.

"Devised by the author".

Once the invoice is uploaded, the documents are sent for processing. The following figure shows the main page of all the invoices uploaded.

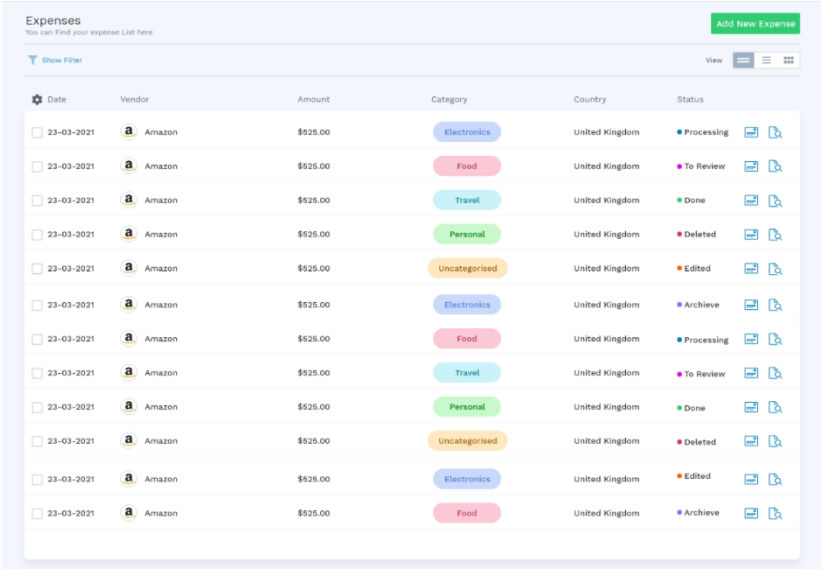


Figure 78: List of Invoice uploaded.

"Devised by the author".

After processing, the document can be opened in a separate screen, and data can be validated.

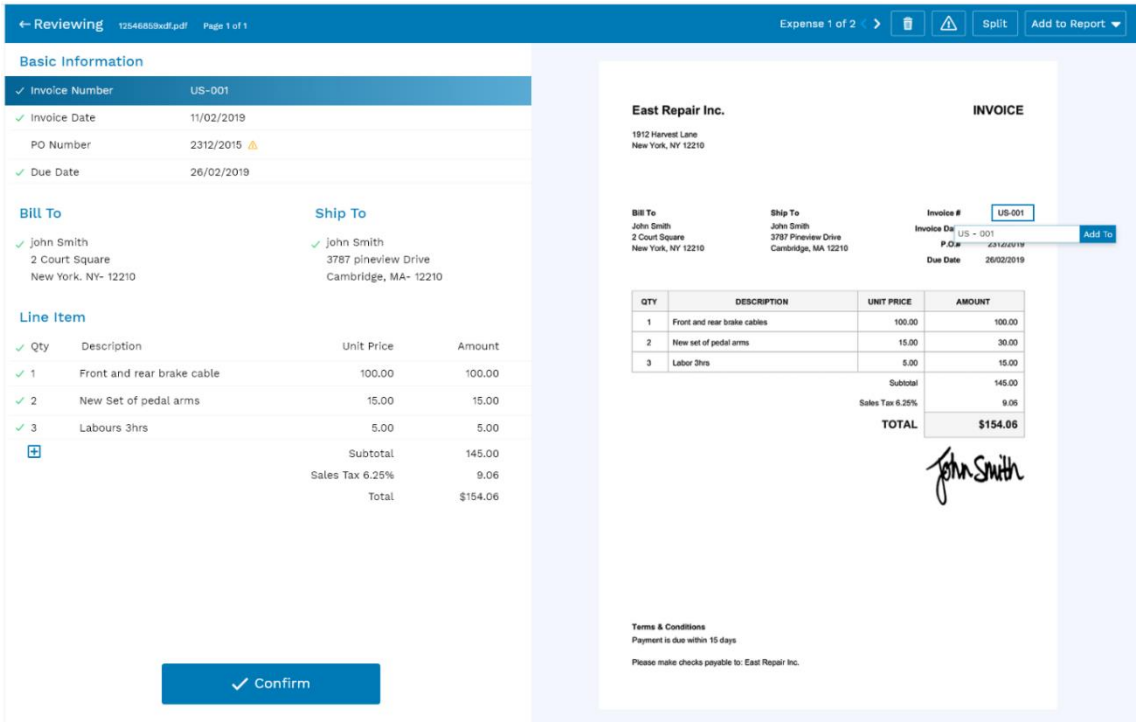


Figure 79: Invoice document editing screen.

"Devised by the author".

Data entry can be done on the left side of the screen by looking at the right side of the screen, where uploaded invoices are shown. The application also allows the drag and drop of text from the right side to the desired field. This is done for quick data entry.

Appendix II: Survey Feedback

Below is the list of descriptive feedback based on a survey done with companies using automated data extraction or are into supplier spend classification and accept the gaps in this area. Apart from these two questionnaires, the rest were objective and have been identified and explained in the gap identification section in chapter 1.

1. Where do you think can the software on text extraction be helpful or can be applied in your existing business?

Table 29: Survey Feedback: Existing business application area.
"Devised by the author".

1.	*Bill Scanning and Expense Management classification. We can count expenses as well as income. Definitely, we don't need to invest valuable time to extract data.*
2.	*Biomedical, business applications, NLP*
3.	*business bank receipt, contract and agreement, tax filing documents, medical*
4.	*In healthcare, to store the patient's information and medical reports.*
5.	*In our healthcare business, we get many Bills every month through the mail for Electricity, Water, Taxes, and Invoice / PO for vendors. If the system is able to automate the process of going through each and every bill or invoice and extract the information with a min 90-95% accuracy, then it will save a lot of time and money. Apart from this, we also get reports sent by patients on mail where we need to manually go through the values of each test done and suggest medicines based on the test value. Many times we also need to manually go through the previous health records of patients. The chances are that we can miss some important information if we can automate the process and extract all the important information with accuracy and present it before the doctors in the required format that will really save time and money.*
6.	*Making documents searchable*
7.	*Many avenues, including statements etc.*
8.	*More customer reach.*
9.	*Parking Receipts*
10.	*This software can be further enhanced so that I can get a better result from manual receipts. My rural clients mostly submit manual receipts. And sometimes, it is challenging to understand the data for the software. I would also like to see if this software can be enhanced for scanning handwritten receipts of local Indian languages.*

11.	*To help our clients extract more data from their invoices/purchase orders for classification purposes.*
12.	*Trade Contracts, Master Agreements, Legal Documents for trade and dealers*

2. Would you like to share some suggestions, feedback, or expectation from such type of data extraction software/tools?

Table 30: Survey Feedback: Suggestions and Expectations.

"Devised by the author".

1.	*I think there should be an option to send money and an option to request money. There should be a list of all transactions, Report etc. One user can easily find all relevant data in one place. There should be an option to fill in data manually and automated. If data extraction matches 100%, it will be good.*
2.	*using CGAN and GNN model, we can improve the accuracy of text extraction...*
3.	*accuracy does not seem to be as good as promised by the company. On enquiry, they will justify and say we will get back to you.*
4.	*Software should be applicable to real-time use without any constraints such as quality of image, time of computations, and the extraction accuracy should be above 95%.*
5.	*Such software can have huge application in the field of medical science however there is no such tool which can extract the information with 90% accuracy also if some tools can be developed which can extract information from the handwritten text that will be super useful in our business.*
6.	*read and process document correctly.*
7.	*There should be a from the tool, if a new billing is done by the card, (I mean the app is on in the mobile and I am doing a bank transaction from my card notification saved in the wallet of the phone) it should automatically grab the data and notify me about the recent transaction and ask me to add the bill or not.*
8.	*As of now, its extracts data from electronic receipts efficiently. But if the receipt is handwritten, then it's unable to interpret. I would like to see in future that can interpret handwritten receipts also efficiently. My rural clients mostly submit manual receipts. And sometimes, it is challenging to understand the data for the software. I would also like to see if this software can be enhanced for scanning handwritten receipts of local Indian languages.*
9.	*In financial markets, there is a big shock to the world of trade where LIBOR is phasing out, and new IBOR will be applied to existing contracts so, if the software is able to gauge the usage of LIBOR in the contracts like Pricing, Collateral Payment, Valuation etc. If that happens, this could be a huge win. Currently, all that is done manually, and it takes huge human effort.*

Appendix III: Supplier and Spend Classification

Spend Classification in Procurement

Supplier Spend classification is the process to show the spending pattern of a company on its suppliers. It is a hierarchical classification of an organization's spend that ranges from 1 to 5 levels based on the complexity of the spend. They are usually classified as:

1. Level 1 (Group)
2. Level 2 (Family)
3. Level 3 (Category)
4. Level 4 (Commodity)

This classification is universally accepted and is known as Spend Classification. There is an international body called UNSPSC (United Nations Standard Products and Services Code), which defines the category for standard international trade deals and services. Furthermore, many organisations define their categories that are derived from UNSPSC to meet the business requirement specification.

Spend classification can be understood by the following example. Suppose a company spend the UNSPSC code is 39101619. "39" refers to expenses on ‘Electrical system and lighting and components’. This is known as level 1 or L1. Adding "10" to get "3910" indicates that the business operates in ‘Lamps, light bulbs and components’. Ten here is known as level 2 or L2. The following two digits distinguish the specific industry sector, so a code of "391016" indicates that the business is concerned with ‘Lamps and lightbulbs’. Here, 16 is known as level 3 or L3. Finally, the last two digits give us the specific business of the suppliers. So, adding “19” to the code gives us “39101619”, which means that the business is involved explicitly in ‘Compact Fluorescent CFL Lamps’. 19 is known as level 4 or L4. A hierarchical diagram can be shown as follows:

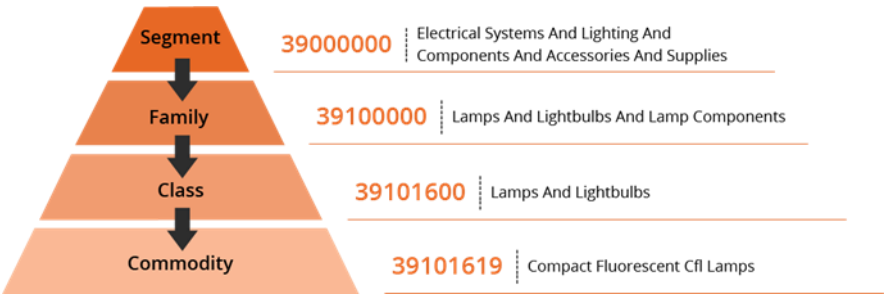


Figure 80: UNSPSC classification.
"Devised by the author".

Spend classification is essential because it provides a uniform, enterprise-wide view of spend and helps instigate meaningful organisational changes. A vendor/supplier spend review aims to determine how much of the money is spent with existing vendors. Using historical data, we can build a comprehensive spend profile for each vendor. It is a tool for optimizing inventory and delivery performance. Not just this, it also allows for multi-faceted research for data-driven decisions and helps to imagine spend insights in a variety of ways, such as by group, geography, and so on (Sievo, 2018). Spend analytics helps in cost reduction initiatives by using accurate and actionable data to help companies make the payment structure work in their best interests.

Supplier Classification

Another part of our work is Supplier classification. It has two parts- supplier spends classification and supplier normalisation. Since there are misspelt or incomplete company names in the data, it becomes difficult for us to gather other details related to them. So, the very first process is to name the organisations correctly. After the normalisation is done, we can get all the company's details from the master database, with which we can do further analysis.

On the other hand, the supplier spends classification is a process under which an organisation evaluates, structure, and classify their suppliers at regular time intervals. The classification is dependent on how buyers view them. Suppliers can be classified as

possible, normal, chosen high-value partner, main supplier, vital supplier, and so on (Stretch, 2015).

Benefits of Supplier classification:

1. Controlling cost.
2. Ensuring excellent service deliverability.
3. Reducing potential risks related to suppliers.
4. Evaluating supplier performance based on organisational standards.

The supplier and spend classification are two sides of the same coin. It can be explained by a concept called SPEND CUBE. If a company does not want to aggregate, analyse, and manage their expenditures across all locations, business units, and divisions, they need to do a spends cube analysis. Since it is projected in a multidimensional cube, the spend cube is a unique way of looking at spend data (Sievo, 2018). The three axes represent Category, Cost Centre, and Supplier.

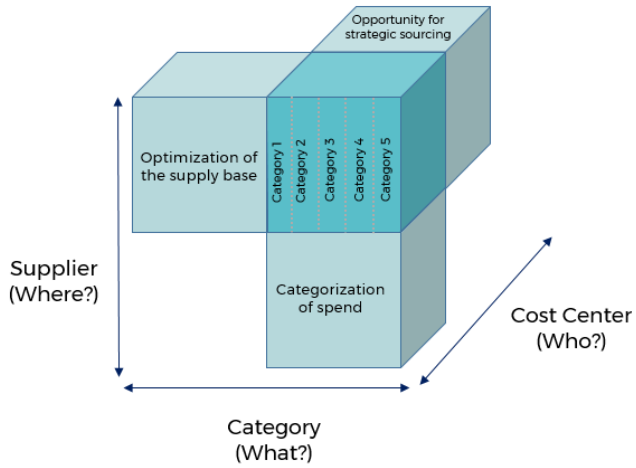


Figure 81: Spend Cube.

"Devised by the author".

Each of the cube's axes contributes important data. The group identifies the essential products and services that we are purchasing. The cost centre identifies which department the products are being purchased for. It is also possible that it's for the end-users. The supplier informs us of which vendors sell such products and services to us (Sievo, 2018).

Manually classifying the suppliers can be a hectic process, especially for large companies with hundreds and thousands of suppliers. So, to overcome this inconvenience, this automated expense and supplier classification was used to classify the client's expenses using Machine Learning. After the client's expenses are classified, we do data visualisation and analysis to know the data better.

Few questions that can be answered from this project are-

1. Which supplier am I spending on the most?
2. Which L1 and L2 am I spending on the most?
3. What has been the spending trend over the years?
4. Which geographical location did we buy it from?
5. Are those expenses relevant?
6. How often do we buy it?
7. What are the reasons for the changes in suppliers' prices?
8. How do I optimize the spending?
9. Are there any alternate suppliers?
10. Where are my saving opportunities?
11. Are we getting what we have been promised?
12. Am I able to achieve lower prices through bulk purchasing and increase efficiency?
13. Is my revenue increasing with the increasing expenses?

We automated spend analysis to accomplish these critical things:

a) Access in-depth analytical reports.

b) Classify and interrogate spend data to support spend management, supplier management and savings management.

c) Maximize savings opportunities by finding alternate suppliers.

d) Track the amount of savings.

e) Log into the portal anytime, anywhere, and from any device.

5. **Expense on each supplier**- We can show how much a client is spending on each supplier with the below plot. The names of the suppliers are plotted on the x-axis, and their expenses are plotted above their respective names.

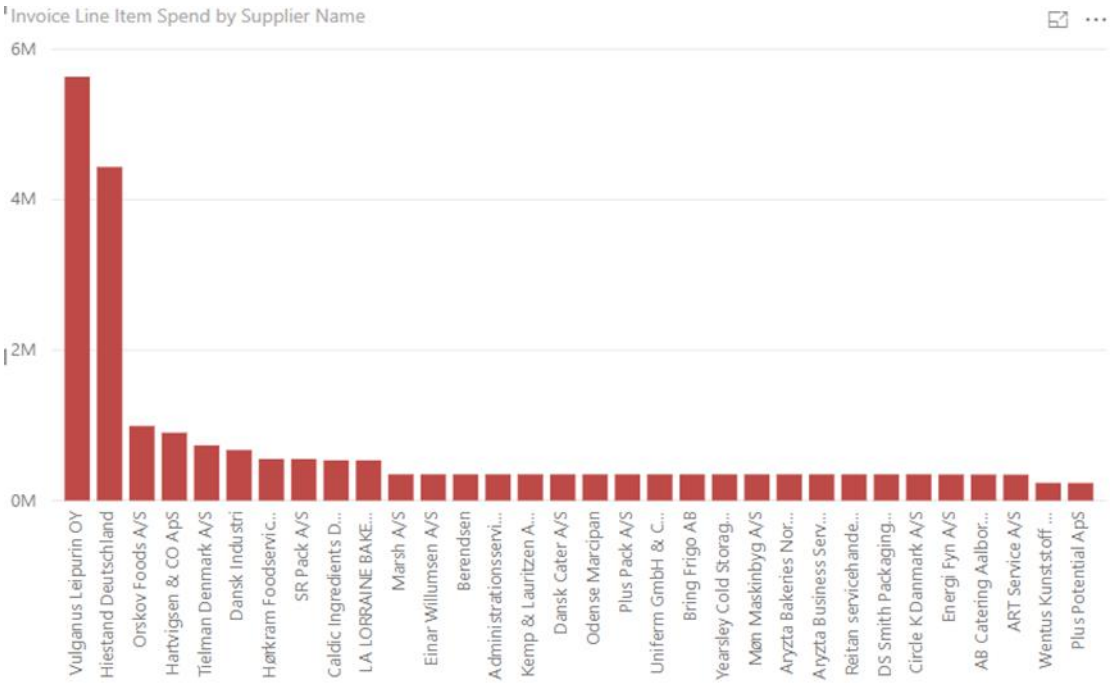


Figure 82: Suppliers with their expenses.
"Devised by the author".

We see that Vulganus Leipurin OY and Hiestand Deutschland are the two suppliers on whom the client is spending the most. These two companies alone account for a total of 41% of the client's expenses, while the other 97 companies combined account for 59% of the spend. Further analysis demonstrates the positive correlation between the supplier with maximum spend and their respective L1 and L2 (i.e., if the client is doing maximum

spend on company ABC, what are the chances that company ABC's L1 and L2 will be the L1 and L2 with maximum spend).

6. **L1 and L2 spend amounts:** The below pie chart shows the spending on different L1. We can see that the maximum expense is being done on FM, and the second number comes.

Vulganus Leipurin OY's L1 is FM, and Hiestand Deutschland's L1 is MRO. Hence, there is a positive correlation between the supplier spend amounts and their L1. So, the supplier with maximum spend is accountable for the highest L1 spend also.

After L1, we see if the same case holds for L2 also.

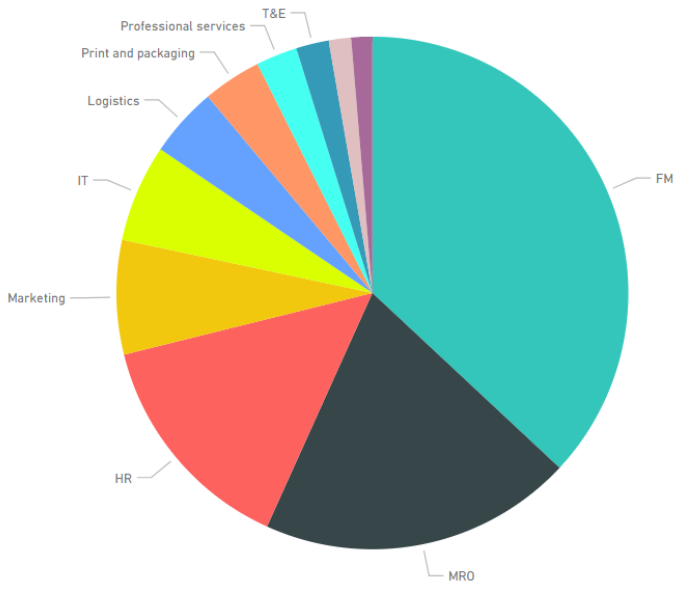


Figure 83: Pie chart showing spending on different l1.
"Devised by the author".

Vulganus Leipurin OY's L2 is Maintenance, and Hiestand Deutschland's L2 is raw materials. The below map shows that Maintenance takes up most of the L2 spending, and the second comes raw materials. Hence, there is a positive correlation between the

supplier spend amounts and their L2 also. So, the supplier with more spend accounts for more L2 spending also.

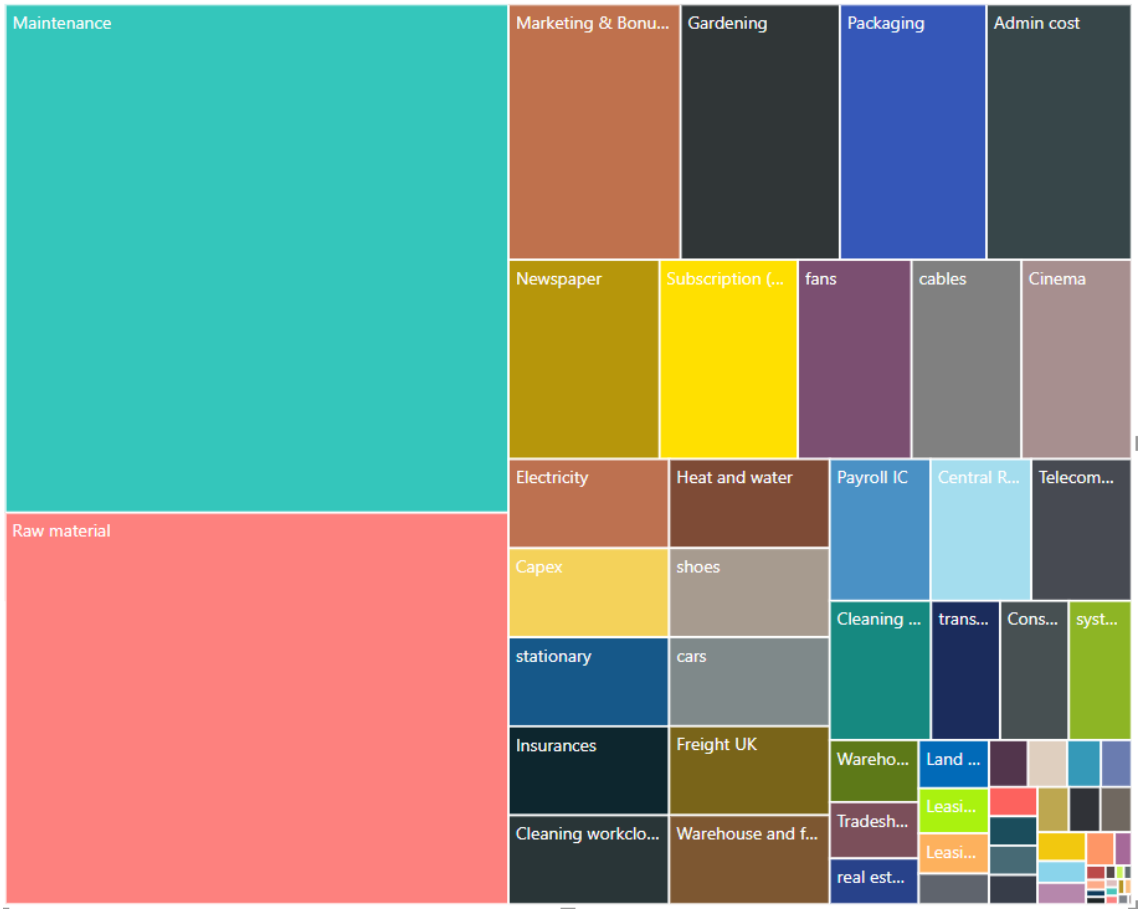


Figure 84: Spending on different L2.
"Devised by the author".

The above two graphs can be consolidated into one so that we can see the different L1 and L2 proportions together like the pie chart below.

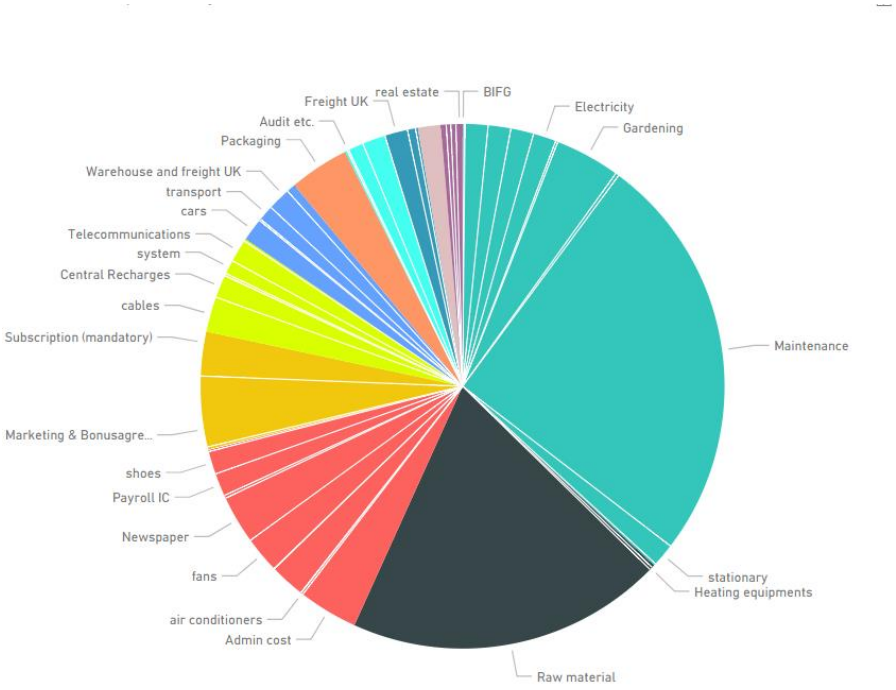


Figure 85: Grouped L1 and L2.
"Devised by the author".

7. **Scope of savings:** Sometimes, a client can be unaware of alternate suppliers who will supply them the product at lower prices and better or the same quality. Therefore, they end up paying more to their current suppliers. If the client had purchased from alternate suppliers, they could have saved a great deal of money. One such example can be shown here: instead of purchasing from Vulganus Leipurin OY, if the client had purchased from Bring Frigo AB, they could have saved $8,08,416.17 from one company alone!

After calculating all of the client's potential savings if they had purchased from alternate suppliers, we find that the client could have saved a total of $1,26,34,097.14 for 2018. The graph below gives us a visualisation of this analysis. It shows the total spend done along with the amount of savings they could do from alternate suppliers. For example, we can see that for some suppliers, there is a scope for savings, like the first and second

one, and for some, there is not, like for the sixth one. This may give the clients an idea of the suppliers that can be replaced with other suppliers and those who are already good enough, helping them save money and getting proper goods and services.

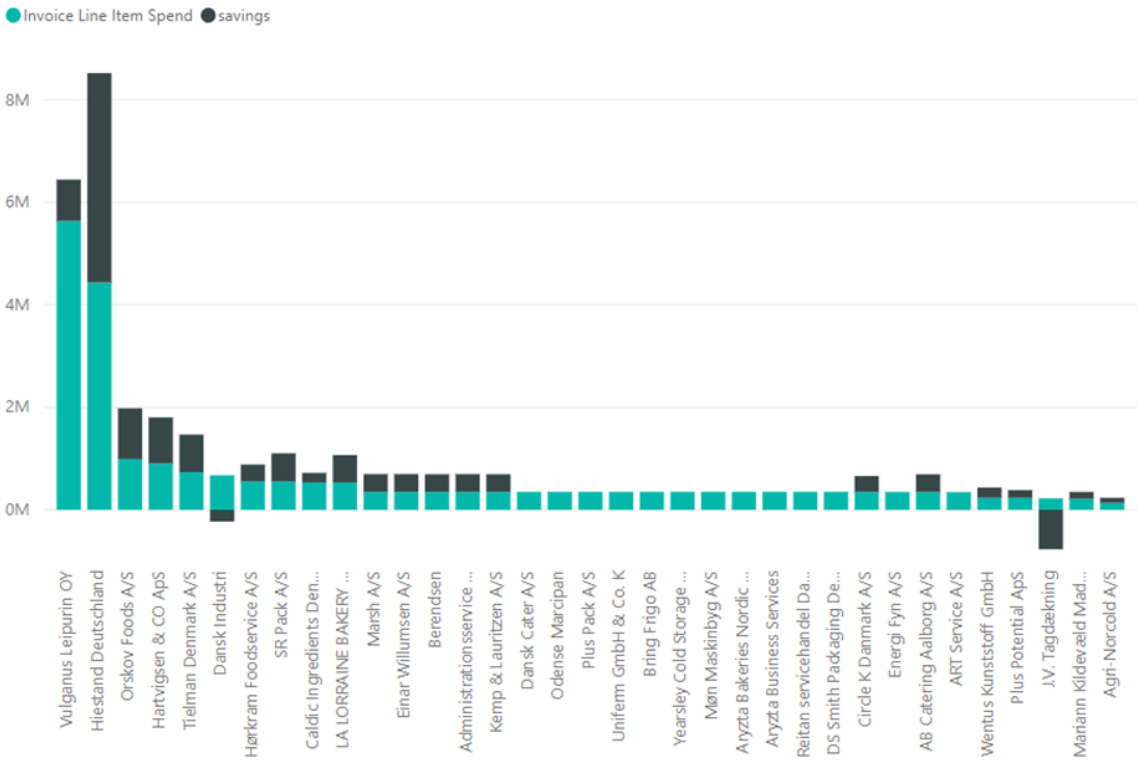


Figure 86: Savings if the client had purchased from other suppliers.
"Devised by the author".

We can show a graph with the suppliers who should be replaced to help the client reduce expenses. The client can then give a second thought to purchase from that supplier or not. An example is shown below. We can see that Hiestand Deutschland, the second-most expensive supplier, can be replaced to save the most expenses. Second comes Vulganus Leipurin OY, which is the most expensive one.

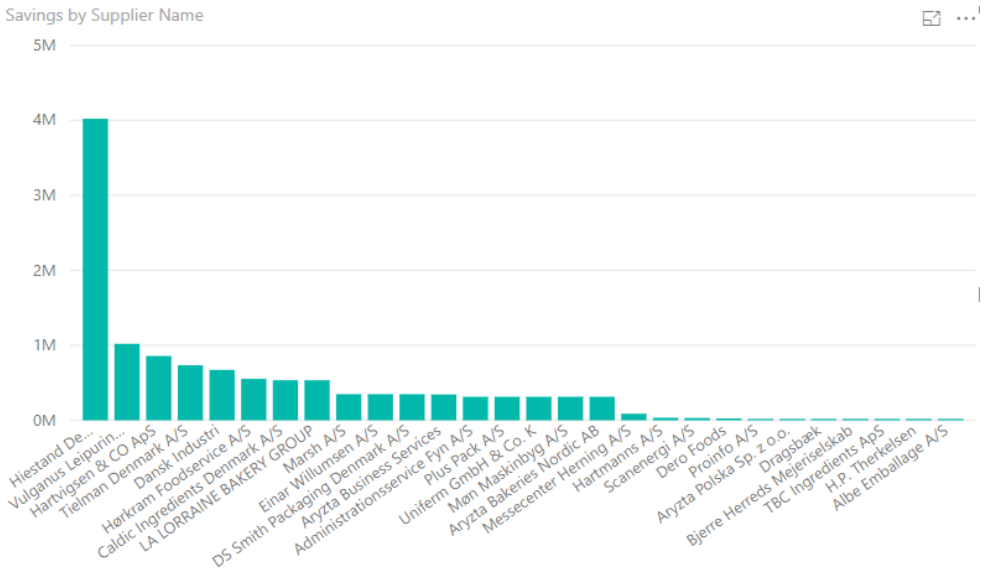


Figure 87: Maximum savings possible from suppliers.
"Devised by the author".

8. **Spend comparison on each supplier over a period of time-** A type of analysis we can show is the trend line of the spending on most expensive suppliers over the given years. This can help us in seeing the spend trends over those years, whether there has been an increase in the expenses or a fall in the expenses done on them and what expenses can we expect to do on them in coming years. This can be shown below:

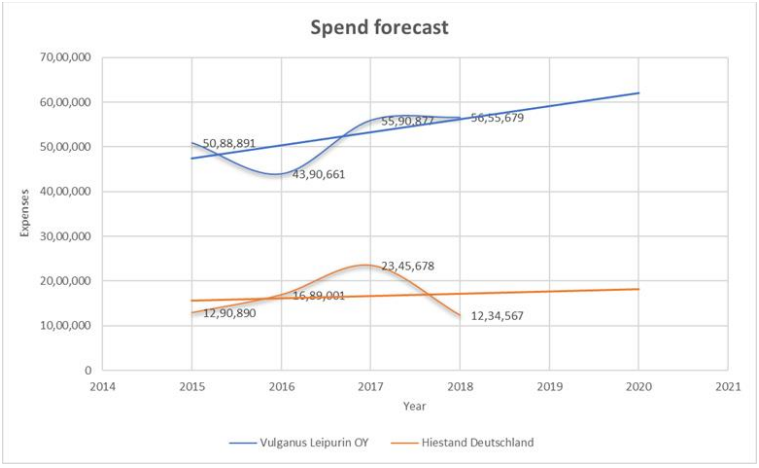


Figure 88: Trendline of top two expensive suppliers.

"Devised by the author".

From the above graph, suppose we take only Vulganus Leipurin OY for understanding the purpose. We can see that there has considerable variation in spending on this supplier over the years, but the trend line shows a steep increase (i.e., the client is expected to increase their spending on this supplier in the coming years). For Hiestand Deutschland, there is a slow upward trend. This analysis can help us in forecasting future expenses on the suppliers. For example, from the graph, we can see that for the year 2019, and Hiestand Deutschland is expected to bring us a bill of $18,00,000 and Vulganus Leipurin OY, a bill of $59,00,000. So, this is the amount the client is estimated to spend on Maintenance and raw materials for the coming year.

We can show another type of visualisation as a comparison of total spending on suppliers over the given years. This graph will help us briefly see the client's expenses on their suppliers for a given number of years. It shows us no trendline or helps us in future forecasting, but we can find out the suppliers on whom the expenses have increased or decreased from the last year so that the client can control their costs.

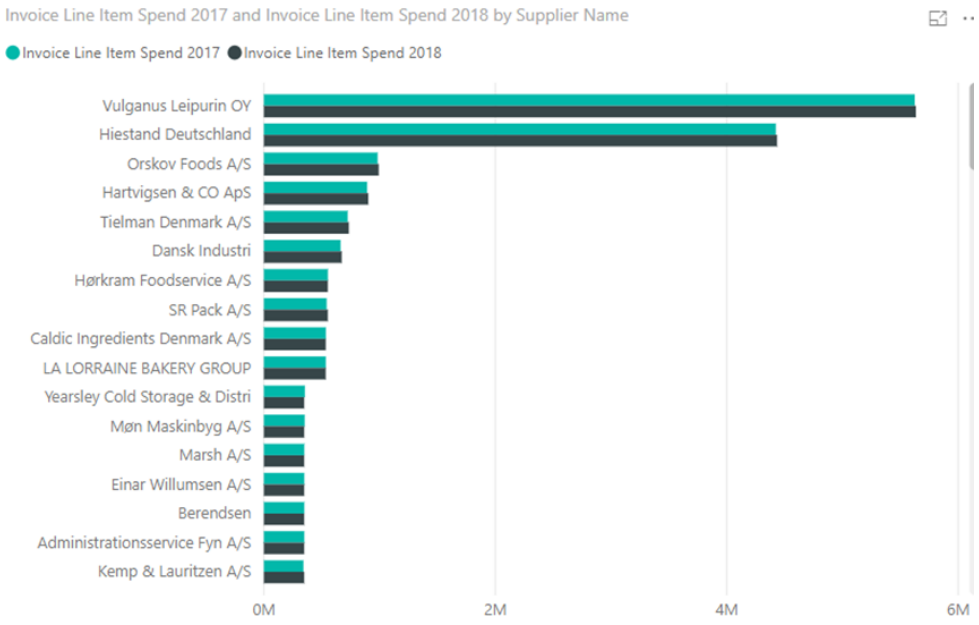


Figure 89: Two-year comparison of supplier spends.

"Devised by the author".

9. **Scope for reducing the supplier list-** In business, it is risky to consider only one supplier. It is preferable to have an alternative option while deciding between single and multiple supplier sourcing. One of the responsibilities of spend analysis is to find out if purchases are being made from preferred suppliers.

Our data shows that the client is purchasing the same product/ service from multiple suppliers. For example, Maintenance. The client has seven suppliers for this service:

Transaction ID	Creditornumber	Supplier Name	Invoice Line Item Spend 2018	GL Description
64	63621527	Pronor IndustriTeknik A/S	42.3	Maintenance
53	66189968	Byens VVS & Blik ApS	345.33	Maintenance
74	63133600	Sanistål A/S	540.3	Maintenance
28	66133750	Viggo Hansen A/S	34701.33	Maintenance
89	62625432	J.V. Tagdækning	222540.3	Maintenance
44	43668888	Kemp & Lauritzen A/S	348715.33	Maintenance
96	3583873750	Vulganus Leipurin OY	5634256.47	Maintenance

Figure 90: Supplier expense sample.

"Devised by the author".

If they had opted to purchase it from a single supplier, they would have:

- lessened their administrative cost.
- received a higher discount and hence lower prices.
- developed a better client-supplier relationship.
- managed suppliers more easily.
- maximized volume leverage to attain attractive pricing.
- reduced cost and optimised SCM.
- worked collaboratively to cut costs and increase the efficiency of purchasing and administrative staff by getting rid of troublesome vendors.
- improved inventory control by gaining negotiating power.
- gained suppliers who were more invested in the business.

What are the possible reasons for the client to purchase it from multiple buyers?

- Buying from only one supplier can be risky. What do you do if they let you down or even go out of business? You need to have at least one supplier to fall back on.
- They would not encounter the scope and breadth of creativity that happens naturally when they have a broad supply base if they choose to use their larger, more conventional suppliers.
- To win over your company, suppliers will try to offer you the best deal possible. This encourages manufacturers to compete, lowering the direct and indirect expenses.

So, these may be some of the reasons the client has chosen to get maintenance service and purchase raw materials from multiple suppliers (as we can see from the data). If we look at the other services like Auditing, cleaning, analytics and consultancy, the client has opted for a single supplier only. Here, we can note that the client is spending mostly on maintenance and raw materials, which has multiple suppliers. So, maybe the client wants to make sure to get the supply or create competition between them to get the best prices.

How would a client choose between multiple suppliers? The answer is- based on price, location, service levels, deliverable date, etc.

- **Price**- Let us take three suppliers who charge various rates for the same service. So, the client will choose the supplier who is charging the lowest price, given the quality of the product/service is standard. Suppose if the price charged by each supplier is the same, say $100, then the transportation cost of delivering the product/ service will be considered. Suppliers charging the lowest transportation cost will be preferred. Now suppose, in a rarest of rare cases, the transportation charge is also the same. Then the **after-purchase service** will be one thing the client will look for. Supplier providing the best after-purchase servicing will be preferred over the others.
- **Location**- Suppose we have two suppliers providing a service at the same price- $100. The difference is their locations. Supplier 1 is located far away and charges

$40 for shipping. Supplier two is located nearby and will charge only $10 for shipping. In this case, it would be natural for the client to choose Supplier 2.

One more constraint that the far-away located supplier will face is the **delivery** time. They will take a longer time to deliver the product or service as compared to the locally situated supplier. So, in case of any urgency, that supplier will not be contacted for the supply of the product.

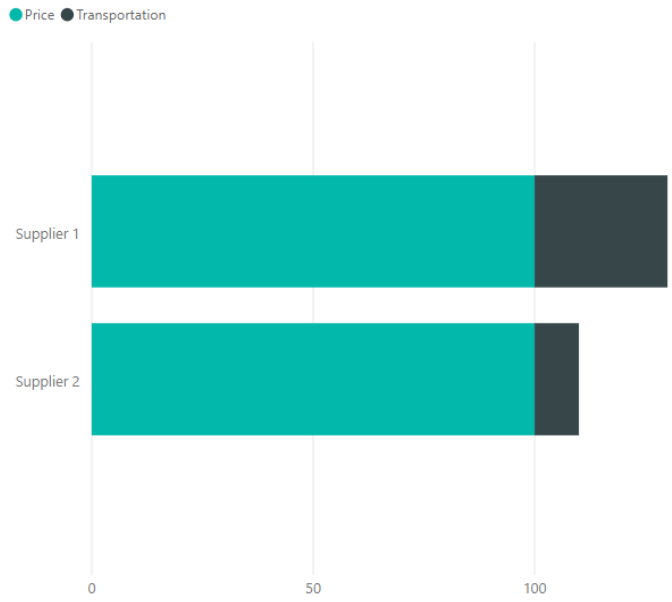


Figure 91: Price and transportation cost difference between two suppliers.
"Devised by the author".

10. **Quarter-wise spending-** If we get data on quarterly spending, we can give various analyses for each quarter like the expenses made quarterly, the different spends on L1 and L2 quarterly, generation of revenue per quarter, etc. These quarterly figures can then be compared with yearly figures to show different trends.

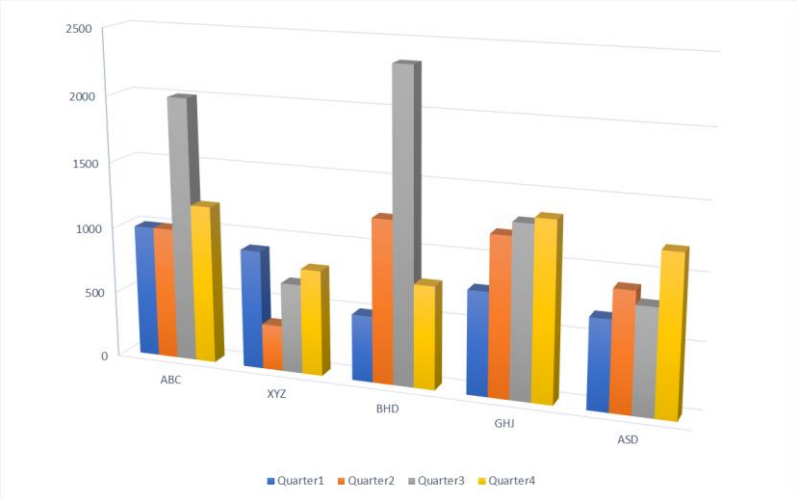


Figure 92: Quarterly expenditure on each supplier.
"Devised by the author".

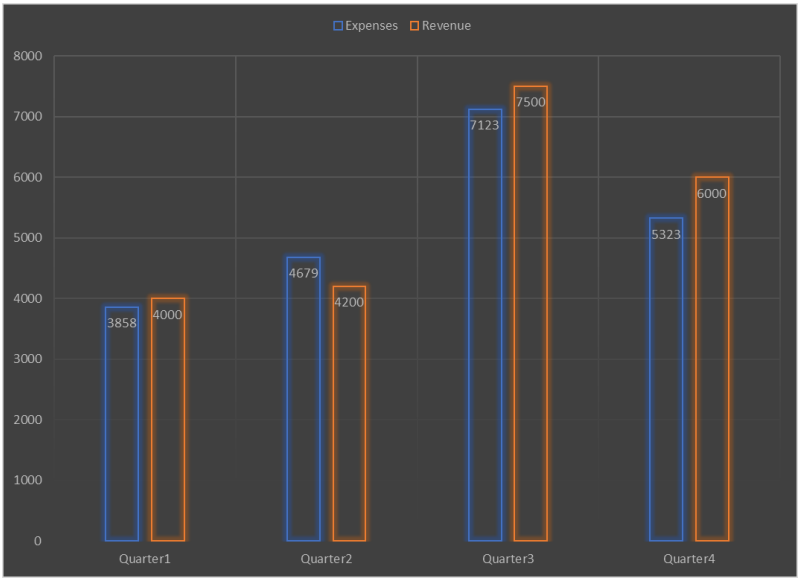


Figure 93: Quarterly expenses and revenue.
"Devised by the author".

11. **Increase in revenue as spend increases-** Increasing expenses over the years are not always bad news. An increase in expenses may also result in an increase in revenue

too, which is good news. This is what our next analysis shows. Given data with total expenses and revenues for several years, we plot it as below and see that there is an increasing trend in both cases. Revenue is above the expenditure, resulting in profits, for all the years except 2016, when the client suffered a loss but recovered back again in the next year.

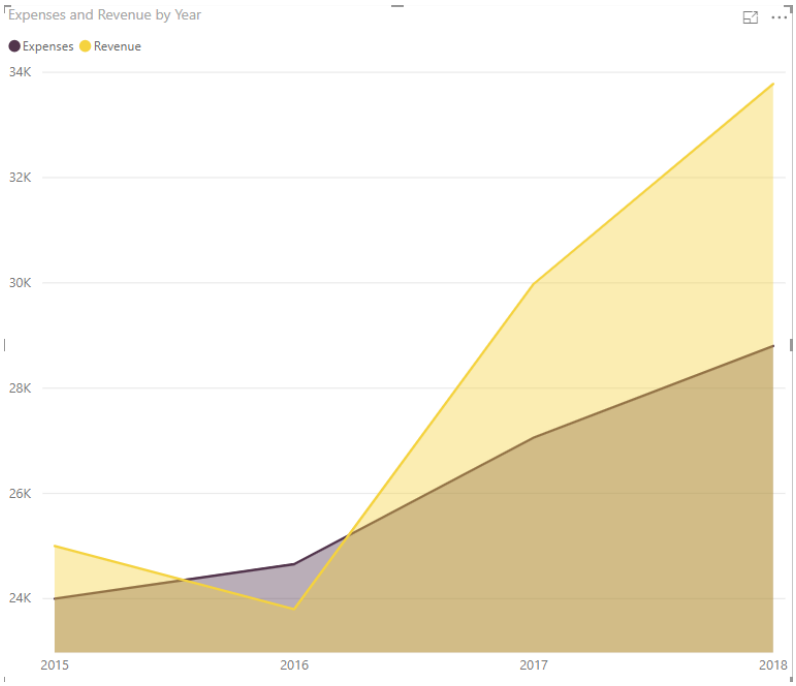


Figure 94: Comparison of revenue and expenses over the years.
"Devised by the author".

One thing worth analysing is the proportion with which the cost and revenue of the client have increased over the years. The above graph shows us that both the revenue and expenses are increasing but what if the expenses are increasing at a higher rate than the revenue? The client is technically losing money then. So, we find out the proportions of their increase by the graph below. We see that the proportionate increase in expenses in the initial years is higher than that of revenue, but after that, the revenue increases at a higher rate. So, the client is earning more than it is spending.

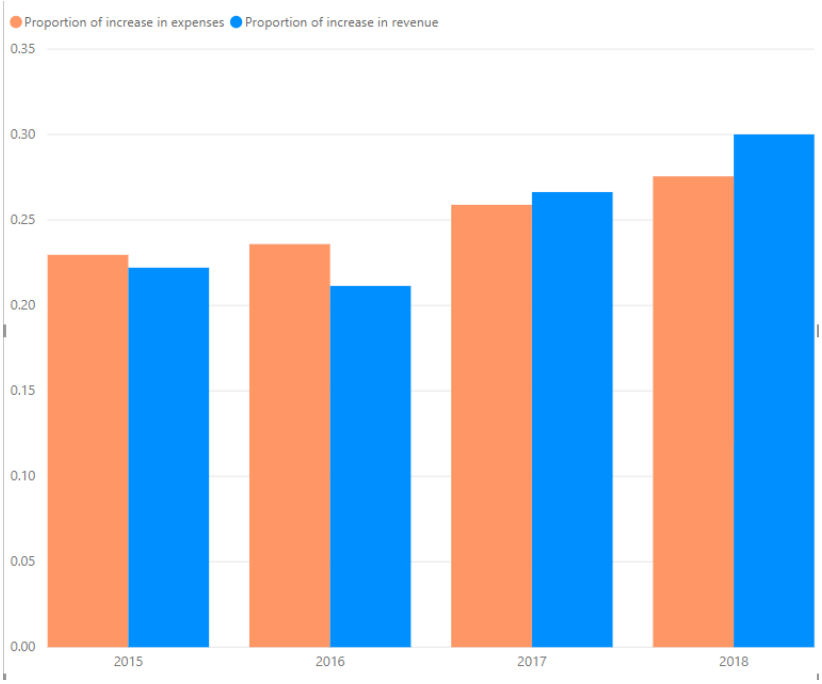


Figure 95: Proportionate increase in revenue and expenses.
"Devised by the author".

12. **Increase in supplier cost as spend increases-** Either the suppliers may increase the price of their goods and services over the years, or the client orders more supplies which leads to increased expenses of the client.

One such graph can be shown below. We can see that the price per unit is stable at $100 for the first three years, but still, the total expenses rise. This is because the quantity purchased increases in each of these three years- from 20 units to 25 units to 30 units in each year, respectively. For the 4th year, the price rises to $110 but the quantity ordered remains the same at 30 units. Therefore, the total expense goes up here.

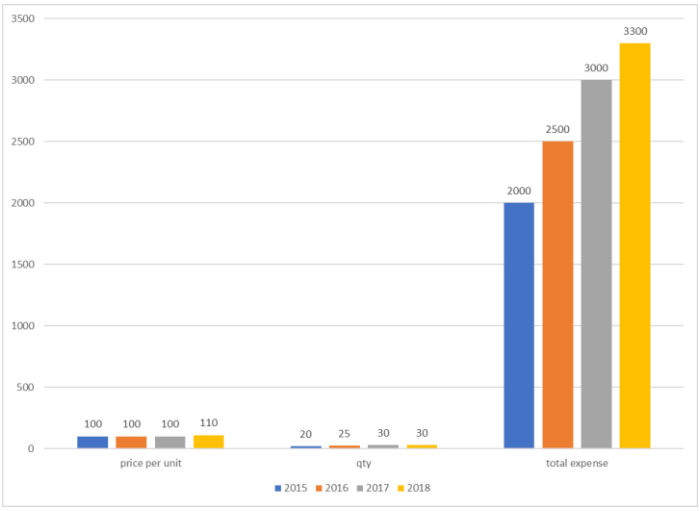


Figure 96: Change in expenses with a change in price and quantity.
"Devised by the author".

13. **Change in price with a change in quality-** There are times when the supplier increases their price and improves their product or service quality, but sometimes this may not be the case. So, we can analyse this by comparing the changes in supplier prices with the improvement in their quality. We can show this by the graph below.

We use the same data as the previous graph with an additional column of the quality rating of the supplier. So, the price remains stagnant in the first three years, but the quality falls each year, which means that the client is paying the same price for a worse quality product. In the 4th year, the price rises, but the quality does not exceed what is provided initially. It matched the quality of the year 2016 when the price was $100. So, this is something the client will have to think about.

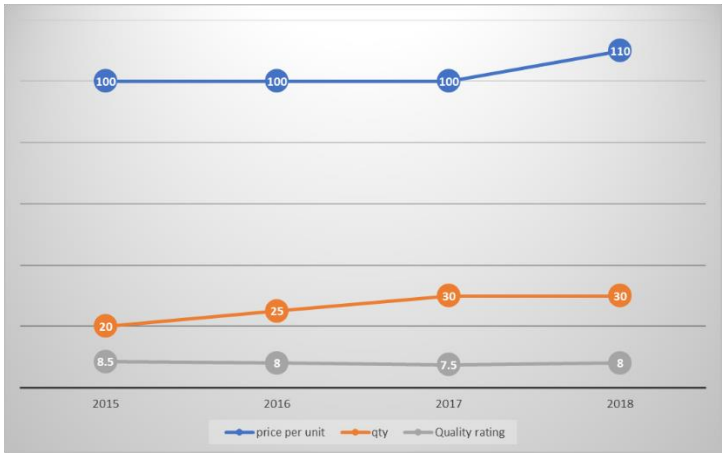


Figure 97: Changes in price and quality.
"Devised by the author".

14. **Country of origin according to expenses-** Below, we can show some clustering of data. Suppose the client has suppliers from 3 countries- India, Germany, and the Netherlands. We can show how much is the client spending on them based on their countries. The graph shows that the total invoice amount is plotted on the Y-axis and the category according to spend amounts on the X-axis. The supplier expenses above $5 million are categorized under A, between $1m and $5m is categorized under B, between $10,000 and $1 m as C and the rest as D.

So, from the graph we can conclude that the suppliers from Germany are the ones on whom maximum expenses are being made, next comes the Netherlands and then India. If the client were importing goods or services, we could show a similar graphical representation like this one.

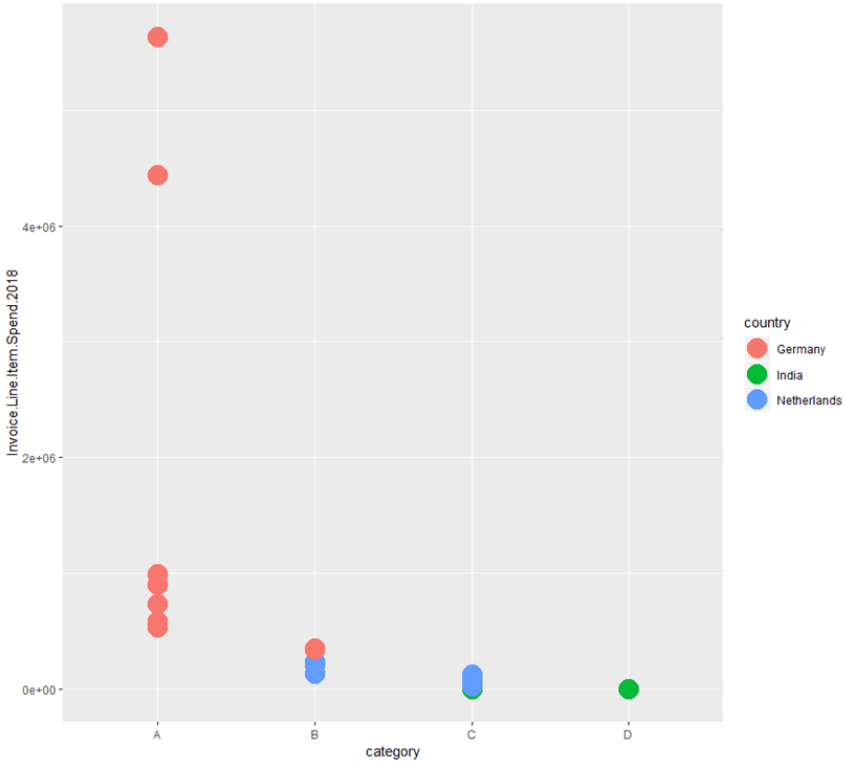


Figure 98: Clustering of data according to the spend done with the country of origin of the suppliers.
"Devised by the author".

15. **PEST Analysis**- PEST stands for Political, Economic, Social, Technological. These are the factors that could have affected the suppliers' prices. The supply of goods and services depends on various external factors that are not under the client's control, like inflation, wage rates, transportation cost, tax rates, cost of raw materials, etc. These factors result in a difference in the suppliers' prices.

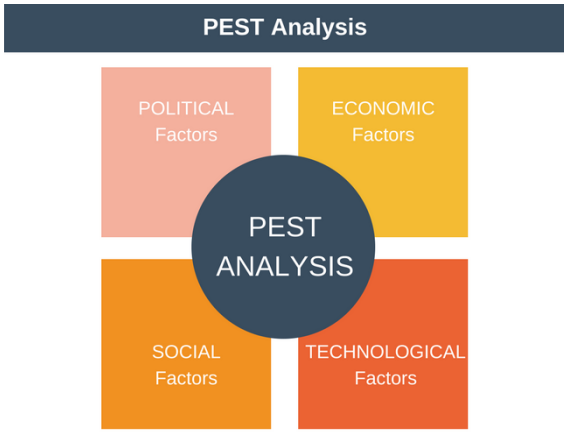


Figure 99: PEST Analysis.
"Devised by the author".

Suppose we take the following dummy data on the probability and reasons for changes in suppliers' prices:

reason	chances
Increase in price of raw materials	0.10
Increase in taxes	0.03
Increase in transportation cost	0.25
Increased profitability margin of suppliers	0.05
Inflation	0.35
Presence of alternate supplier	0.22
Total	**1.00**

Figure 100: Dummy data showing the effect of the external factors on prices.
"Devised by the author".

So, if a supplier increases its prices, it is 35% because of inflation, 25% because of an increase in transportation cost, etc. This data can be visualized by the pie-chart below:

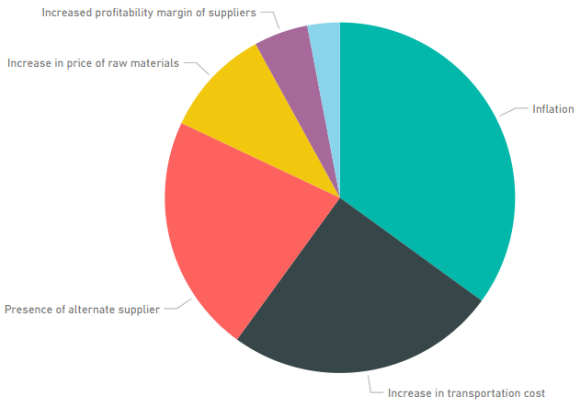


Figure 101: Chances of external factors affecting the price.
"Devised by the author".

References

ABBYY (2020) *ABBYY Cloud OCR SDK - Text recognition via Web API.* Available online: https://www.abbyy.com/cloud-ocr-sdk/ [Accessed.

Acharjya, D. P. & Ahmed, K. (2016) A Survey on Big Data Analytics: Challenges, Open Research Issues and Tools. *International Journal of Advanced Computer Science and Applications*, 7.

Agarwal, M., Shalika, V. T. & Gupta, P. (2019) Handwritten Character Recognition using Neural Network and Tensor Flow. *International Journal of Innovative Technology and Exploring Engineering (IJITEE).*

Agarwal, S., Godbole, S., Punjani, D. & Roy, S. (2007) *How Much Noise Is Too Much: A Study in Automatic Text Classification*. IEEE.

Ahmadzade, A. & Malekzadeh, S. (2021) Spell Correction for Azerbaijani Language using Deep Neural Networks. *arXiv preprint arXiv:2102.03218*.

Akhtar, P., Tse, Y. K., Khan, Z. & Rao-Nicholson, R. (2016) Data-driven and adaptive leadership contributing to sustainability: Global agri-food supply chains connected with emerging markets. *International Journal of Production Economics*, 181, 392-401.

Alippi, C., Pessina, F. & Roveri, M. (2005) An adaptive system for automatic invoice-documents classification, *IEEE International Conference on Image Processing 2005*. IEEE.

Alshehri, S. (2021) English Characters OCR Pertinent for Mobile Devices. *International Journal of Computing and Digital Systems*, 10(1), 135-141.

Amazon (2020) *Amazon Textract Features | AWS.* Available online: https://aws.amazon.com/textract/features/ [Accessed.

Amin, S. S. & Ragha, L. (2021) Text Generation and Enhanced Evaluation of Metric for Machine Translation, *Data Intelligence and Cognitive Informatics*Springer, 1-17.

Anand, G. S., Kuriakose, J., Sharma, S. & Guha, D. (2020) Deep Learning for Information Extraction in Finance Documents—Corporate Loan Operations, *2020 11th IEEE Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*. IEEE.

Anand, S. & Khan, Z. (2020) WA net: Leveraging Atrous and Deformable Convolutions for Efficient Text Detection.

Arai, K. & Tolle, H. (2011) Method for real time text extraction of digital manga comic. *International Journal of Image Processing (IJIP)*, 4(6), 669-676.

Aral, S. & Walker, D. (2012) Identifying influential and susceptible members of social networks. *Science (New York, N.Y.)*, 337, 337-41.

Arora, A., Chang, C. C., Rekabdar, B., BabaAli, B., Povey, D., Etter, D., Raj, D., Hadian, H., Trmal, J. & Garcia, P. (2019) Using ASR methods for OCR, *2019 International Conference on Document Analysis and Recognition (ICDAR)*. IEEE.

Arora, K., Bist, A. S., Prakash, R. & Chaurasia, S. (2020) Custom OCR for Identity Documents: OCRXNet. *Aptisi Transactions On Technopreneurship (ATT)*, 2(2), 112-119.

Arroyo, R., Tovar, J., Delgado, F. J., Almazán, E. J., Serrador, D. G. & Hurtado, A. (2019) Integration of Text-maps in Convolutional Neural Networks for Region Detection among Different Textual Categories. *arXiv preprint arXiv:1905.10858*.

Aslan, E., Karakaya, T., Unver, E. & Akgul, Y. S. (2015) *An optimization approach for invoice image analysis*. IEEE.

Ast, U. (2020) Systems and methods for generating and using semantic images in deep learning for classification and data extraction. Google Patents.

Avadesh, M. & Goyal, N. (2018) Optical character recognition for sanskrit using convolution neural networks, *2018 13th IAPR International Workshop on Document Analysis Systems (DAS)*. IEEE.

Babu, S. (2020) *Automating Receipt Digitization with OCR and Deep Learning.* Available online: https://nanonets.com/blog/receipt-ocr/ [Accessed.

Bajpai, N. (2011) Business Research Methods.

Bart, E. & Sarkar, P. (2010) Information extraction by finding repeated structure. *DAS '10: Proceedings of the 9th IAPR International Workshop on Document Analysis Systems*, 175-182.

Bartoli, A., Davanzo, G., Medvet, E. & Sorio, E. (2014) Semisupervised Wrapper Choice and Generation for Print-Oriented Documents. *IEEE Transactions on Knowledge and Data Engineering*, 26, 208-220.

Bartz, C., Yang, H. & Meinel, C. (2017) STN-OCR: A single neural network for text detection and text recognition. *arXiv preprint arXiv:1707.08831*.

Bassil, Y. & Alwani, M. (2012) Ocr post-processing error correction algorithm using google online spelling suggestion. *arXiv preprint arXiv:1204.0191*.

Baumann, S., Ali, M. B. H., Dengel, A., Jager, T., Malburg, M., Weigel, A. & Wenzel, C. (1997) Message extraction from printed documents-a complete solution, *Proceedings of the Fourth International Conference on Document Analysis and Recognition*. IEEE.

Bayer, T. & Mogg-Schneider, H. (1997) *A generic system for processing invoices*. IEEE Comput. Soc.

Beaufort, R. & Mancas-Thillou, C. (2007) A weighted finite-state framework for correcting errors in natural scene OCR, *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*. IEEE.

Belaïd, Y. & Belaïd, A. (2004) Morphological tagging approach in document analysis of invoices, *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*: IEEE.

Belay, B., Habtegebrial, T., Meshesha, M., Liwicki, M., Belay, G. & Stricker, D. (2020) Amharic OCR: An End-to-End Learning. *Applied Sciences*, 10(3), 1117.

Bentley, R. A., O'Brien, M. J. & Brock, W. A. (2014) Mapping collective behavior in the big-data era. *Behavioral and Brain Sciences*, 37, 63-76.

Berg Palm, R., Winther, O. & Laws, F. (2017) CloudScan-A configuration-free invoice analysis system using recurrent neural networks. *arXiv*, arXiv: 1708.07403.

Bettany-Saltikov, J. (2010) Learning how to undertake a systematic review: part 2. *Nursing Standard (through 2013)*, 24(51), 47.

Bhaire, V. V., Jadhav, A. A., Pashte, P. A. & Magdum, P. (2015) Spell checker. *International Journal of Scientific and Analisys Publication*, 5.

Bhatt, H. S., Roy, S., Bhatnagar, L., Lohani, C. & Jain, V. (2019) Digital Auditor: A Framework for Matching Duplicate Invoices, *2019 International Conference on Document Analysis and Recognition (ICDAR)*. IEEE.

Bissacco, A., Cummins, M., Netzer, Y. & Neven, H. (2013) Photoocr: Reading text in uncontrolled conditions, *Proceedings of the IEEE International Conference on Computer Vision*.

Blanchard, J., Belaïd, Y. & Belaïd, A. (2019) Automatic generation of a custom corpora for invoice analysis and recognition, *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)*. IEEE.

Bokrantz, J., Skoogh, A., Berlin, C. & Stahre, J. (2017) Maintenance in digitalised manufacturing: Delphi-based scenarios for 2030. *International Journal of Production Economics*, 191, 154-169.

Borisyuk, F., Gordo, A. & Sivakumar, V. (2018) Rosetta: Large scale system for text detection and recognition in images, *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*.

Brauer, F., Schramm, M., Barczynski, W., Loser, A., Do, H.-H., Löser, A. & Hong-Hai, D. (2008) *Robust recognition of complex entities in text exploiting enterprise data and NLP-techniques*. IEEE.

Breuel, T. M., Ul-Hasan, A., Al-Azawi, M. A. & Shafait, F. (2013) High-performance OCR for printed English and Fraktur using LSTM networks, *2013 12th International Conference on Document Analysis and Recognition*. IEEE.

Bryman, A. (2015) Social research methods, 747.

Brzeski, A., Grinholc, K., Nowodworski, K. & Przybyłek, A. (2019) Evaluating performance and accuracy improvements for attention-OCR, *IFIP International Conference on Computer Information Systems and Industrial Management*. Springer.

Burrell, G. & Morgan, G. (2008) Sociological paradigms and organisational analysis : elements of the sociology of corporate life, 432.

Busta, M., Neumann, L. & Matas, J. (2015) Fastext: Efficient unconstrained scene text detector, *Proceedings of the IEEE International Conference on Computer Vision*.

Cappelatti, E., Heidrich, R. D. O., Oliveira, R., Monticelli, C., Rodrigues, R., Goulart, R. & Velho, E. (2018) Post-correction of OCR Errors Using PyEnchant Spelling Suggestions Selected Through a Modified Needleman–Wunsch Algorithm, *International Conference on Human-Computer Interaction*. Springer.

Cesarini, F., Gori, M., Marinai, S. & Soda, G. (1998) INFORMys: a flexible invoice-like form-reader system. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20, 730-745.

Chakraborty, P. & Mallik, A. (2013) An open source tesseract based tool for extracting text from images with application in braille translation for the visually impaired. *International Journal of Computer Applications*, 68(16).

Chang, S. F., Smith, J. R. & Wang, H. (1995) Automatic Feature Extraction and Indexing for Content-Based Visual Query, CU/CTR 414.

Charan & Lina (2019) It GAN DO Better: GAN-based Detection of Objects on Images with Varying Quality. *arXiv pre-print server*.

Chen, D. & Bourlard, H. (2001) *Video OCR for sport video annotation and retrieval*.

Chen, D., Shearer, K. & Bourlard, H. (2001) *Text enhancement with asymmetric filter for video OCR*.

Cheng, Z., Xu, Y., Bai, F., Niu, Y., Pu, S. & Zhou, S. (2018) Aon: Towards arbitrarily-oriented text recognition, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Chiang, Y.-Y. & Knoblock, C. A. (2011) Recognition of multi-oriented, multi-sized, and curved text, *2011 International Conference on Document Analysis and Recognition*. IEEE.

Chien, C.-H. & Lin, D.-T. (2009) Uniform-Invoice Number Extraction.

Chiron, G., Doucet, A., Coustaty, M., Visani, M. & Moreux, J. P. (2017) Impact of OCR Errors on the Use of Digital Libraries: Towards a Better Access to Information. *Proceedings of the ACM/IEEE Joint Conference on Digital Libraries*.

Chu, H., Chai, Y., Liu, Y. & Sun, H. (2014) A novel E-Invoice Framework towards data-oriented taxation system, *Proceedings of the 2014 IEEE 18th International Conference on Computer Supported Cooperative Work in Design (CSCWD)*. IEEE.

Chun, B. T., Bae, Y. & Kim, T.-Y. (1999) *Automatic text extraction in digital videos using FFT and neural network*. IEEE.

Chung, Y.-N., Chiu, M.-S., Lin, C.-C., Wang, J.-Y. & Hsu, C.-H. (2019) An efficient pattern recognition technology for numerals of lottery and invoice, *International Conference on Genetic and Evolutionary Computing*. Springer.

Cinti, A., Bianchi, F. M., Martino, A. & Rizzi, A. (2020) A novel algorithm for online inexact string matching and its fpga implementation. *Cognitive Computation*, 12(2), 369-387.

Coustaty, M., Doucet, A., Jatowt, A. & Nguyen, N.-V. (2018) Adaptive Edit-Distance and Regression Approach for Post-OCR Text Correction, *International Conference on Asian Digital Libraries*. Springer.

Cristani, M. & Tomazzoli, C. (2014) A Multimodal Approach to Exploit Similarity in Documents, *Modern Advances in Applied Intelligence.* 490-499.

Crotty, M. (1998) The foundations of social research : meaning and perspective in the research process, 248.

Damerau, F. J. (1964) A technique for computer detection and correction of spelling errors. *Communications of the ACM*, 7(3), 171-176.

Dave, H., Gobse, A., Goel, A. & Bairagi, S. (2020) OCR Text Detector and Audio Convertor.

Davis, J., Edgar, T., Porter, J., Bernaden, J. & Sarli, M. (2012) Smart manufacturing, manufacturing intelligence and demand-dynamic performance. *Computers & Chemical Engineering*, 47, 145-156.

de Jager, C. & Nel, M. (2019) Business Process Automation: A Workflow Incorporating Optical Character Recognition and Approximate String and Pattern Matching for Solving Practical Industry Problems. *Applied System Innovation*, 2(4), 33.

Delie, M., Jian, L. & Jinwen, T. (2002) The design and implementation of a Chinese financial invoice recognition system, *International Symposium on VIPromCom Video/Image Processing and Multimedia Communications*. IEEE.

Denzin, N. K. & Lincoln, Y. S. (2011) The Sage handbook of qualitative research, 766.

Dong, R. & Smith, D. A. (2018) Multi-input attention for unsupervised OCR correction, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Dwivedi, A., Saluja, R. & Kiran Sarvadevabhatla, R. (2020) An OCR for Classical Indic Documents Containing Arbitrarily Long Words, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*.

Eddy, M. D. (2004) Fallible or inerrant? A belated review of the constructivists bible. *The British Journal for the History of Science*, 37, 93-98.

El Atawy, S. & Abd ElGhany, A. (2018) Automatic Spelling Correction based on n-Gram Model.

Elagouni, K., Garcia, C., Mamalet, F. & Sébillot, P. (2012) Combining multi-scale character recognition and linguistic knowledge for natural scene text OCR, *2012 10th IAPR International Workshop on Document Analysis Systems*. IEEE.

Esser, D., Muthmann, K. & Schuster, D. (2013) *Information extraction efficiency of business documents captured with smartphones and tablets*. New York, New York, USA: ACM Press.

Etoori, P., Chinnakotla, M. & Mamidi, R. (2018) Automatic spelling correction for resource-scarce languages using deep learning, *Proceedings of ACL 2018, Student Research Workshop*.

Fabrizio, J., Cord, M. & Marcotegui, B. (2009) Text Extraction From Street Level Images. *Isprsorg*, XXXVIII, 3-4.

Fataicha, Y., Cheriet, M., Nie, J. Y. & Suen, C. Y. (2006) Retrieving poorly degraded OCR documents. *International Journal of Document Analysis and Recognition (IJDAR)*, 8(1), 15.

Fisher, D., DeLine, R., Czerwinski, M. & Drucker, S. (2012) Interactions with big data analytics. *Interactions*, 19, 50.

Gandomi, A. & Haider, M. (2015) Beyond the hype: Big data concepts, methods, and analytics. *International Journal of Information Management*, 35, 137-144.

Gao, H., Rusinol, M., Karatzas, D., Llados, J., Jain, R. & Doermann, D. (2015) *Novel line verification for multiple instance focused retrieval in document collections*. IEEE.

Geetha, M., Pooja, R., Swetha, J., Nivedha, N. & Daniya, T. (2020) Implementation of text recognition and text extraction on formatted bills using deep learning. *Int J Contrl Automat*, 13(2), 646-651.

Gilani, A., Qasim, S. R., Malik, I. & Shafait, F. (2017) Table detection using deep learning, *2017 14th IAPR international conference on document analysis and recognition (ICDAR)*. IEEE.

Girshick, R. (2015) Fast r-cnn, *Proceedings of the IEEE international conference on computer vision*.

Gllavata, J., Ewerth, R. & Freisleben, B. (2004) A text detection, localization and segmentation system for OCR in images, *IEEE Sixth International Symposium on Multimedia Software Engineering*. IEEE.

Goel, V., Kumar, V., Jaggi, A. S. & Nagrath, P. (2019) Text extraction from natural scene images using OpenCV and CNN. *Int. J. Inf. Technol. Comput. Sci*, 11(9), 48-54.

Golder, S. A. & Macy, M. W. (2011) Diurnal and seasonal mood vary with work, sleep, and daylength across diverse cultures. *Science (New York, N.Y.)*, 333, 1878-81.

Griffin, E. P. & Kurup, P. U. (2017) Prediction of OCR and su from PCPT Data Using Tree-Based Data Fusion Techniques. *Journal of Geotechnical and Geoenvironmental Engineering*, 143, 04017045.

Grönlund, J. & Johansson, A. (2019) Defect Detection and OCR on Steel.

Grover, S., Arora, K. & Mitra, S. K. (2009) Text Extraction from Document Images Using Edge Information. *2009 Annual IEEE India Conference*, 1-4.

Gui, G. (2019) The Image Preprocessing and Check of Amount for VAT Invoices. *Communications, Signal Processing, and Systems: Proceedings of the 2018 CSPS Volume II: Signal Processing*, 516, 44.

Guo, H., Qin, X., Liu, J., Han, J., Liu, J. & Ding, E. (2019) Eaten: Entity-aware attention for single shot visual text extraction, *2019 International Conference on Document Analysis and Recognition (ICDAR)*. IEEE.

Gupta, N. & Mathur, P. (2012) Spell checking techniques in NLP: a survey. *International Journal of Advanced Research in Computer Science and Software Engineering*, 2(12).

Gupta, S. & Dutta, K. (2011) Modeling of financial supply chain. *European Journal of Operational Research*, 211, 47-56.

Gupta, V., Aggarwal, A. & Ghose, U. (2016) *Rule based information leveraging from business invoices*. IEEE.

Gurjar, T. & Parihar, A. (2020) A Survey on XML Data Processing Using Data Mining Techniques.

Ha, H. T., Nevěřilová, Z. & Horák, A. (2018) Recognition of ocr invoice metadata block types, *International Conference on Text, Speech, and Dialogue*. Springer.

Halima, M. B., Alimi, A. & Vila, A. F. (2012) Nf-savo: Neuro-fuzzy system for arabic video ocr. *arXiv preprint arXiv:1211.2150*.

Hamza, H., Belaïd, Y. & Belaïd, A. (2007) Case-Based Reasoning for Invoice Analysis and Recognition, *Case-Based Reasoning Research and Development*. Berlin, Heidelberg: Springer Berlin Heidelberg, 404-418.

Hauptmann, A. G., Jin, R. & Ng, T. D. (2002) Multi-modal information retrieval from broadcast video using OCR and speech recognition, *Proceedings of the 2nd ACM/IEEE-CS joint conference on Digital libraries*.

Henge, S. K. & Rama, B. (2016) *Neural fuzzy closed loop hybrid system for classification, identification of mixed connective consonants and symbols with layered methodology*. IEEE.

Ho, T. K. & Nagy, G. (2000) OCR with no shape training, *Proceedings 15th International Conference on Pattern Recognition. ICPR-2000*. IEEE.

Holeček, M., Hoskovec, A., Baudiš, P. & Klinger, P. (2019) Table understanding in structured documents, *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)*. IEEE.

Hua, X.-S., Yin, P. & Zhang, H.-J. (2002) Efficient video text recognition using multiple frame integration, *Proceedings. International Conference on Image Processing*. IEEE.

Huang, Z. (1997) A Fast Clustering Algorithm to Cluster Very Large Categorical Data Sets in Data Mining. *Research Issues on Data Mining and Knowledge Discovery*, 1-8.

Huang, Z., Chen, K., He, J., Bai, X., Karatzas, D., Lu, S. & Jawahar, C. (2019a) Icdar2019 competition on scanned receipt ocr and information extraction, *2019 International Conference on Document Analysis and Recognition (ICDAR)*. IEEE.

Huang, Z., Zhong, Z., Sun, L. & Huo, Q. (2019b) Mask R-CNN with pyramid attention network for scene text detection, *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE.

Iddo, G. (2018) Top 5 OCR (Optical Character Recognition) APIs & Software | RapidAPI, *The Last Call - RapidAPI Blog*Available online: https://rapidapi.com/blog/top-5-ocr-apis/.

İnce, E. Y. (2017) Spell Checking and Error Correcting Application for Turkish. *International Journal of Information and Electronics Engineering*, 7(2).

Insurance, T., Bureau, F., Abi, T. & Limited, B. (2013) Fraud detection – the unstructured data goldmine. *Blakehead Limited*.

Islam, M. R., Mondal, C., Azam, M. K. & Islam, A. S. M. J. (2016) Text detection and recognition using enhanced MSER detection and a novel OCR technique, *2016 5th International Conference on Informatics, Electronics and Vision (ICIEV)*. IEEE.

Isola, P., Zhu, J.-Y., Zhou, T. & Efros, A. A. (2017) Image-to-image translation with conditional adversarial networks, *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Jacobs, C., Simard, P. Y., Viola, P. & Rinker, J. (2005) Text recognition of low-resolution document images, *Eighth International Conference on Document Analysis and Recognition (ICDAR'05)*. IEEE.

Jain, M., Mathew, M. & Jawahar, C. (2017) Unconstrained ocr for urdu using deep cnn-rnn hybrid networks, *2017 4th IAPR Asian Conference on Pattern Recognition (ACPR)*. IEEE.

Jatowt, A., Coustaty, M., Nguyen, N.-V. & Doucet, A. (2019) Post-OCR Error Detection by Generating Plausible Candidates, *2019 International Conference on Document Analysis and Recognition (ICDAR)*. IEEE.

Javed, S. T. & Hussain, S. (2009) Improving Nastalique specific pre-recognition process for Urdu OCR, *2009 IEEE 13th International Multitopic Conference*. IEEE.

Javed, S. T., Hussain, S., Maqbool, A., Asloob, S., Jamil, S. & Moin, H. (2010) Segmentation free nastalique urdu ocr. *World Academy of Science, Engineering and Technology*, 46, 456-461.

Jenckel, M., Bukhari, S. S. & Dengel, A. (2016) anyocr: A sequence learning based ocr system for unlabeled historical documents, *2016 23rd International Conference on Pattern Recognition (ICPR)*. IEEE.

Jia, F., Gong, Y. & Brown, S. (2019) Multi-tier sustainable supply chain management: The role of supply chain leadership. *International Journal of Production Economics*, 217, 44-63.

Jiang, F., Chen, H. & Zhang, L.-J. (2018) FCN-biLSTM Based VAT Invoice Recognition and Processing, *International Conference on Edge Computing*. Springer.

Jin, R., Hauptmann, A. G. & Zhai, C. (2002) A content-based probabilistic correction model for OCR document retrieval, *The 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval Workshop Program (SIGIR 2002)*.

Jun, C., Suhua, Y. & Shaofeng, J. (2019) Automatic classification and recognition of complex documents based on Faster RCNN, *2019 14th IEEE International Conference on Electronic Measurement & Instruments (ICEMI)*. IEEE.

Kakhani, M., Kakhani, S. & Biradar, S. (2013) Research Issues in Big Data Analytics. *International Journal of Application or Innovation in Engineering & Management (IJAIEM)*, 2, 228-232.

Kanagarathinam, K. & Sekar, K. (2019) Text detection and recognition in raw image dataset of seven segment digital energy meter display. *Energy Reports*, 5, 842-852.

Karanje, U. B. & Dagade, R. (2014) Survey on Text Detection, Segmentation and Recognition from a Natural Scene Images. *International Journal of Computer Applications*, 108, 975-8887.

Karaoglu, S., Van Gemert, J. C. & Gevers, T. (2012) Object reading: text recognition for object recognition, *European Conference on Computer Vision*. Springer.

Kastelan, I., Kukolj, S., Pekovic, V., Marinkovic, V. & Marceta, Z. (2012) Extraction of text on TV screen using optical character recognition, *2012 IEEE 10th Jubilee International Symposium on Intelligent Systems and Informatics*. IEEE.

Katti, A. R., Reisswig, C., Guder, C., Brarda, S., Bickel, S., Höhne, J. & Faddoul, J. B. (2018) Chargrid: Towards understanding 2d documents. *arXiv preprint arXiv:1809.08799*.

Khan, N., Yaqoob, I., Hashem, I. A. T., Inayat, Z., Mahmoud Ali, W. K., Alam, M., Shiraz, M. & Gani, A. (2014) Big Data: Survey, Technologies, Opportunities, and Challenges. *The Scientific World Journal*, 2014, 1-18.

Khoddami, M. & Behrad, A. (2010) *Farsi and Latin script identification using curvature scale space features*. IEEE.

Kieninger, T. & Dengel, A. (2001) Applying the T-RECS table recognition system to the business letter domain, *Proceedings of Sixth International Conference on Document Analysis and Recognition*. IEEE.

Kim, D., Kwak, M., Won, E., Shin, S. & Nam, J. (2020) TLGAN: document Text Localization using Generative Adversarial Nets. *arXiv preprint arXiv:2010.11547*.

Kissos, I. & Dershowitz, N. (2016) OCR error correction using character correction and feature-based word classification, *2016 12th IAPR Workshop on Document Analysis Systems (DAS)*. IEEE.

Kissos, I. & Dershowitz, N. (2017) Image and text correction using language models, *2017 1st International Workshop on Arabic Script Analysis and Recognition (ASAR)*. IEEE.

Klampfl, S., Granitzer, M., Jack, K. & Kern, R. (2014) Unsupervised document structure analysis of digital scientific articles. *International Journal on Digital Libraries*, 14, 83-99.

Kluzner, V., Tzadok, A., Shimony, Y., Walach, E. & Antonacopoulos, A. (2009) Word-based adaptive OCR for historical books, *2009 10th International Conference on Document Analysis and Recognition*. IEEE.

KodeKnight (2011) BK-Treesal, *BK-Treesal*Available online: https://k2code.blogspot.com/2011/02/bk-treesal.html.

Kolak, O., Byrne, W. & Resnik, P. (2003) A generative probabilistic OCR model for NLP applications, *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*.

Koo, H. I. (2016) Text-line detection in camera-captured document images using the state estimation of connected components. *IEEE Transactions on Image Processing*, 25(11), 5358-5368.

Kooli, N. & Belaid, A. (2017) Inexact graph matching for entity recognition in OCRed documents. *Proceedings - International Conference on Pattern Recognition*, 4071-4076.

Köppen, M., Waldöstl, D. & Nickolay, B. (1998) A system for the automated evaluation of invoices, *Document Analysis Systems II*World Scientific, 223-241.

Kosiba, D. a. & Kasturi, R. (1996) *Automatic invoice interpretation: invoice structure analysis*. IEEE.

Krieger, F., Drews, P., Funk, B. & Wobbe, T. (2021) Information Extraction from Invoices: A Graph Neural Network Approach for Datasets with High Layout Variety.

Krishna, A., Majumder, B. P., Bhat, R. S. & Goyal, P. (2018) Upcycle your ocr: Reusing ocrs for post-ocr text correction in romanised sanskrit. *arXiv preprint arXiv:1809.02147*.

Krishnan, P., Sankaran, N., Singh, A. K. & Jawahar, C. (2014) Towards a robust ocr system for indic scripts, *2014 11th IAPR International Workshop on Document Analysis Systems*. IEEE.

Kruatrachue, B., Somguntar, K. & Siriboon, K. (2002) Thai OCR error correction using genetic algorithm, *First International Symposium on Cyber Worlds, 2002. Proceedings.*: IEEE.

Kuhn, T. S. (1962) The Structure of Scientific Revolutions. *Structure*, 2, 172.

Kumar, A. (2016) A survey on various OCR errors. *International Journal of Computer Applications*, 143(4), 8-10.

Kumar, P. & Revathy, S. (2021) An Automated Invoice Handling Method Using OCR, *Data Intelligence and Cognitive Informatics*Springer, 243-254.

Kumar, R., Bala, M. & Sourabh, K. (2018) A study of spell checking techniques for Indian Languages. *JK Research Journal in Mathematics and Computer Sciences*, 1(1).

Kumuda, T. & Basavaraj, L. (2017) *Edge Based Segmentation Approach to Extract Text from Scene Images*. IEEE.

Kundi, G. M. & Nawaz, A. (2010) From objectivism to social constructivism: The impacts of information and communication technologies (ICTs) on higher education, *Journal of Science and Technology Education Research*.

Kundu, S., Paul, S., Bera, S. K., Abraham, A. & Sarkar, R. (2020) Text-line extraction from handwritten document images using GAN. *Expert Systems with Applications*, 140, 112916.

Labrinidis, A. & Jagadish, H. V. (2012) Challenges and opportunities with big data. *Proceedings of the VLDB Endowment*, 5, 2032-2033.

Lacchia, M. (2017) Interesting data structures: the BK-tree, *Interesting data structures: the BK-tree*Available online: https://signal-to-noise.xyz/post/bk-tree/.

Laine, M. & Nevalainen, O. S. (2006) A standalone OCR system for mobile cameraphones, *2006 IEEE 17th International Symposium on Personal, Indoor and Mobile Radio Communications*. IEEE.

Law, K. M. & Gunasekaran, A. (2012) Sustainability development in high-tech manufacturing firms in Hong Kong: Motivators and readiness. *International Journal of Production Economics*, 137(1), 116-125.

Le Vine, N., Zeigenfuse, M. & Rowan, M. (2019) Extracting tables from documents using conditional generative adversarial networks and genetic algorithms, *2019 International Joint Conference on Neural Networks (IJCNN)*. IEEE.

Lee, C.-Y. & Osindero, S. (2016) Recursive recurrent nets with attention modeling for ocr in the wild, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Lehal, G. S. & Singh, C. (2002) A post-processor for Gurmukhi OCR. *Sadhana*, 27(1), 99-111.

Leon, M., Vilaplana, V., Gasull, A. & Marques, F. (2013) Region-based caption text extraction. *Lecture Notes in Electrical Engineering*, 158 LNEE, 21-36.

Levenshtein, V. I. (1966) Binary codes capable of correcting deletions, insertions, and reversals, *Soviet physics doklady*. Soviet Union.

Lhoussain, A. S., Hicham, G. & Abdellah, Y. (2015) Adaptating the levenshtein distance to contextual spelling correction. *International Journal of Computer Science and Applications*, 12(1), 127-133.

Li, X., Wang, W., Hou, W., Liu, R.-Z., Lu, T. & Yang, J. (2018) Shape robust text detection with progressive scale expansion network. *arXiv preprint arXiv:1806.02559*.

Li, Y., Gao, L., Tang, Z., Yan, Q. & Huang, Y. (2019) A GAN-based feature generator for table detection, *2019 International Conference on Document Analysis and Recognition (ICDAR)*. IEEE.

Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B. & Belongie, S. (2017) Feature pyramid networks for object detection, *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Liu, L., Muelly, M., Deng, J., Pfister, T. & Li, L.-J. (2019a) Generative Modeling for Small-Data Object Detection. *arXiv pre-print server*.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y. & Berg, A. C. (2016a) Ssd: Single shot multibox detector, *European conference on computer vision*. Springer.

Liu, W., Wan, B. & Zhang, Y. (2016b) Unstructured Document Recognition on Business Invoice.

Liu, X., Gao, F., Zhang, Q. & Zhao, H. (2019b) Graph convolution for multimodal information extraction from visually rich documents. *arXiv preprint arXiv:1903.11279*.

Liu, Y., Jin, Y., Huang, C. & Bao, W. (2020) Table detection method based on feature pyramid network with faster R-CNN, *Twelfth International Conference on Digital Image Processing (ICDIP 2020)*. International Society for Optics and Photonics.

Loginov, V., Valiukov, A., Semenov, S. & Zagaynov, I. (2020) Document Data Extraction System Based on Visual Words Codebook, *International Workshop on Document Analysis Systems*. Springer.

Lohani, D., Belaïd, A. & Belaïd, Y. (2018) An invoice reading system using a graph convolutional network, *Asian Conference on Computer Vision*. Springer.

Lu, Y.-F., Zhang, A.-X., Li, Y., Yu, Q.-H. & Qiao, H. (2019) Multi-Scale Scene Text Detection Based on Convolutional Neural Network, *2019 Chinese Automation Congress (CAC)*. IEEE.

Lund, W. B., Kennard, D. J. & Ringger, E. K. (2013) Combining multiple thresholding binarization values to improve OCR output, *Document Recognition and Retrieval XX*. International Society for Optics and Photonics.

Lyu, M. R., Song, J. & Cai, M. (2005) A comprehensive method for multilingual video text detection, localization, and extraction. *IEEE transactions on circuits and systems for video technology*, 15(2), 243-255.

Ma, X. & Hovy, E. (2016) End-to-end sequence labeling via bi-directional lstm-cnns-crf. *arXiv preprint arXiv:1603.01354*.

Magdy, W. & Darwish, K. (2006) Word-based correction for retrieval of Arabic OCR degraded documents, *International Symposium on String Processing and Information Retrieval*. Springer.

Majumder, P., Mitra, M. & Chaudhuri, B. (2002) N-gram: a language independent approach to IR and NLP, *International conference on universal knowledge and language*.

Mao, W. M. W., Chung, F.-l. C. F.-l., Lam, K. K. M. & Sun, W.-c. S. W.-c. (2002) Hybrid Chinese/English text detection in images and video frames. *Object recognition supported by user interaction for service robots*, 3, 2-5.

Marovic, M., Mikša, M., Šnajder, J. & Dalbelo Bašic, B. (2010) Croatian OCR Error Correction Using Character Confusions and Language Modelling, *Proc of the Central European Conference on Information and Intelligent Systems, CECIIS 2010,(in press)*.

Martınek, J., Lenc, L. & Král, P. (2020) Building an efficient OCR system for historical documents with little training data.

Medvet, E., Bartoli, A. & Davanzo, G. (2011) A probabilistic approach to printed document understanding. *International Journal on Document Analysis and Recognition (IJDAR)*, 14, 335-347.

Megan, E. (2018) The Best Receipt Apps for Scanning, Tracking, and Managing Bills. *MUO*.

Mei, J., Islam, A., Moh'd, A., Wu, Y. & Milios, E. (2018) Statistical learning for OCR error correction. *Information Processing & Management*, 54(6), 874-887.

Meng, Y., Liang, Y., Sun, Y., Pan, J. & Gui, G. (2019a) Smart Phone Aided Intelligent Invoice Reimbursement System, *International Conference on Advanced Hybrid Information Processing*. Springer.

Meng, Y., Wang, R., Wang, J., Yang, J. & Gui, G. (2019b) IRIS: Smart Phone Aided Intelligent Reimbursement System Using Deep Learning. *IEEE Access*, 7, 165635-165645.

Miloudi, F. E., Tchernev, N. & Rian, F. (2016) Scheduling payments optimization to drive working capital performance within a supply chain. *ILS 2016 - 6th International Conference on Information Systems, Logistics and Supply Chain*, 1-8.

Mithe, R., Indalkar, S. & Divekar, N. (2013) Optical character recognition. *International journal of recent technology and engineering (IJRTE)*, 2(1), 72-75.

ML, G. (2021) *Descending into ML: Training and Loss | Machine Learning Crash Course*. Available online: https://developers.google.com/machine-learning/crash-course/descending-into-ml/training-and-loss [Accessed.

Moens, M.-F. (2006) *Information extraction: algorithms and prospects in a retrieval context*, *21*.Springer Science & Business Media.

Mohapatra, Y., Mishra, A. K. & Mishra, A. K. (2013) Spell checker for OCR. *International Journal of Computer Science and Information Technologies*, 4(1), 91-97.

Moyne, J. & Iskandar, J. (2017) Big Data Analytics for Smart Manufacturing: Case Studies in Semiconductor Manufacturing. *Processes*, 5, 39.

Moyne, J., Samantaray, J. & Armacost, M. (2016) Big Data Capabilities Applied to Semiconductor Manufacturing Advanced Process Control. *IEEE Transactions on Semiconductor Manufacturing*, 29, 283-291.

Muhammad, M., ELGhazaly, T., Ezzat, M. & Gheith, M. (2016) A Spell Correction Model for OCR Errors for Arabic Text, *International Conference on Advanced Intelligent Systems and Informatics*. Springer.

Murugan, S., Bakthavatchalam, T. A. & Sankarasubbu, M. (2020) SymSpell and LSTM based Spell-Checkers for Tamil.

Myers, M. D. (2009) Qualitative Research in Business & Management. *Qualitative research in business management*, 284.

Nagabhushan, P. (2010) Text Extraction in Complex Color Document Images for Enhanced Readability. *Intelligent Information Management*, 02, 120-133.

Nagaoka, Y., Miyazaki, T., Sugaya, Y. & Omachi, S. (2017) Text detection by faster R-CNN with multiple region proposal networks, *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*. IEEE.

Namysl, M. & Konya, I. (2019) Efficient, lexicon-free OCR using deep learning, *2019 International Conference on Document Analysis and Recognition (ICDAR)*. IEEE.

Nartker, T. A., Taghva, K., Young, R., Borsack, J. & Condit, A. (2003) OCR correction based on document level knowledge, *Document Recognition and Retrieval X*. International Society for Optics and Photonics.

Naseem, T. & Hussain, S. (2007) A novel approach for ranking spelling error corrections for Urdu. *Language Resources and Evaluation*, 41(2), 117-128.

Nashwan, F., Rashwan, M. A., Al-Barhamtoshy, H. M., Abdou, S. M. & Moussa, A. M. (2018) A holistic technique for an Arabic OCR system. *Journal of Imaging*, 4(1), 6.

Niklas, K. (2010) Unsupervised post-correction of ocr errors. *Master's thesis. Leibniz Universität Hannover*.

OmniPage (2020) *OmniPage Capture SDK – OCR Systeme.* Available online: https://www.ocr-systeme.de/en/index/omnipage-sdk/ [Accessed.

Onnela, J.-P. & Reed-Tsochas, F. (2010) Spontaneous emergence of social influence in online systems. *Proceedings of the National Academy of Sciences of the United States of America*, 107, 18375-80.

Opresnik, D. & Taisch, M. (2015) The value of Big Data in servitization. *International Journal of Production Economics*, 165, 174-184.

Ormerod, P. (2012) Positive Linking: How Networks Can Revolutionise the World, 320.

Packer, T. L., Lutes, J. F., Stewart, A. P., Embley, D. W., Ringger, E. K., Seppi, K. D. & Jensen, L. S. (2010) Extracting person names from diverse and noisy OCR text, *Proceedings of the fourth workshop on Analytics for noisy unstructured text data*.

Pal, U., Kundu, P. K. & Chaudhuri, B. B. (2000) OCR error correction of an inflectional indian language using morphological parsing. *J. Inf. Sci. Eng.*, 16(6), 903-922.

Paliwal, S. S., Vishwanath, D., Rahul, R., Sharma, M. & Vig, L. (2019) TableNet: Deep Learning model for end-to-end Table detection and Tabular data extraction from Scanned Document Images, *2019 International Conference on Document Analysis and Recognition (ICDAR)*. IEEE.

Patel, S. & Bhatt, D. (2020) Abstractive Information Extraction from Scanned Invoices (AIESI) using End-to-end Sequential Approach. *arXiv preprint arXiv:2009.05728*.

Paul, D. & Chaudhuri, B. B. (2019) A BLSTM Network for Printed Bengali OCR System with High Accuracy. *arXiv preprint arXiv:1908.08674*.

Paul D. Leedy, J. E. O. (2010) Practical Research Planning and Design.

Paul, J. A. & Wang, X. J. (2015) Robust optimization for United States Department of Agriculture food aid bid allocations. *Transportation Research Part E: Logistics and Transportation Review*, 82, 129-146.

Pawar, K., Bhabal, P., Shinde, K. & Tekwani, B. (2020) Digital KYC with Auto Form Filling.

Pegu, B., Singh, M., Kant, K., Singh, K. & Bhowmik, T. (2021) MoDest: Multi-module Design Validation for Documents, *8th ACM IKDD CODS and 26th COMAD*, 332-340.

Perez-Cortes, J. C., Amengual, J.-C., Arlandis, J. & Llobet, R. (2000) Stochastic error-correcting parsing for OCR post-processing, *Proceedings 15th International Conference on Pattern Recognition. ICPR-2000*. IEEE.

Pervin, M. T., Afroge, S. & Huq, A. (2017) A feature fusion based optical character recognition of bangla characters using support vector machine, *2017 3rd International Conference on Electrical Information and Communication Technology (EICT)*. IEEE.

Petr, B. (2019) Why Manual Invoice Data Capture is Bad for a Company|Rossum. *Cognitive Data Capture | Rossum*.

Pham, V. A., Nguyen, D. T. K., Tran, M. D. & Pham, V. D. (2020) IMPROVED OCR QUALITY FOR SMART SCANNED DOCUMENT MANAGEMENT SYSTEM. *Journal of Science and Technique-Section on Information and Communication Technology*(210).

Pitou, C. & Diatta, J. (2016) *Textual Information Extraction in Document Images Guided by a Concept Lattice*.

Poignant, J., Besacier, L., Quénot, G. & Thollard, F. (2012) From text detection in videos to person identification, *2012 IEEE International Conference on Multimedia and Expo*. IEEE.

Poncelas, A., Aboomar, M., Buts, J., Hadley, J. & Way, A. (2020) A Tool for Facilitating OCR Postediting in Historical Documents. *arXiv preprint arXiv:2004.11471*.

Pourghahestani, F. A. & Rashedi, E. (2015) Object detection in images using artificial neural network and improved binary gravitational search algorithm, *2015 4th Iranian Joint Congress on Fuzzy and Intelligent Systems (CFIS)*. IEEE.

Prameela, N., Anjusha, P. & Karthik, R. (2017) Off-line Telugu handwritten characters recognition using optical character recognition, *2017 International conference of Electronics, Communication and Aerospace Technology (ICECA)*. IEEE.

Prasad, R., Saleem, S., Kamali, M., Meermeier, R. & Natarajan, P. (2008) Improvements in hidden Markov model based Arabic OCR, *2008 19th International Conference on Pattern Recognition*. IEEE.

Provost, F. & Fawcett, T. (2013) Data Science and its Relationship to Big Data and Data-Driven Decision Making. *Big Data*, 1, 51-59.

Purwantoro, D., Akbar, H. & Hidayati, A. (2019) OCR correction for Indonesian historic newspapers using word repetition, stemmer and n-gram, *Journal of Physics: Conference Series*. IOP Publishing.

Qi, W., Gu, L., Jiang, H., Chen, X.-R. & Zhang, H.-J. (2000) Integrating visual, audio and text analysis for news video, *Proceedings 2000 International Conference on Image Processing (Cat. No. 00CH37101)*. IEEE.

Qiu, M., Su, H., Chen, M., Ming, Z. & Yang, L. T. (2012) Balance of security strength and energy for a PMU monitoring system in smart grid. *IEEE Communications Magazine*, 50, 142-149.

Rabbi, J., Ray, N., Schubert, M., Chowdhury, S. & Chao, D. (2020) Small-object detection in remote sensing images with end-to-end edge-enhanced GAN and object detector network. *Remote Sensing*, 12(9), 1432.

Rahal, N., Benjlaiel, M. & Alimi, A. M. (2016a) Entity Extraction and Correction Based on Token Structure Model Generation, *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR).* Cham: Springer International Publishing, 401-411.

Rahal, N., Benjlaiel, M. & Alimi, A. M. (2016b) *Incremental structural model for extracting relevant tokens of entity*. IEEE.

Rahal, N., Tounsi, M., Benjlaiel, M. & Alimi, A. M. (2018) Information Extraction from Arabic and Latin scanned invoices, *2018 IEEE 2nd International Workshop on Arabic and Derived Script Analysis and Recognition (ASAR)*. IEEE.

Rahman, M., Watanobe, Y. & Nakamura, K. (2021) A Bidirectional LSTM Language Model for Code Evaluation and Repair. *Symmetry*, 13(2), 247.

Ramena, G., Nagaraju, D., Moharana, S., Mohanty, D. P. & Purre, N. (2020) An Efficient Architecture for Predicting the Case of Characters using Sequence Models, *2020 IEEE 14th International Conference on Semantic Computing (ICSC)*. IEEE.

Raoui-Outach, R., Million-Rousseau, C., Benoit, A. & Lambert, P. (2017) Deep Learning for automatic sale receipt understanding, *2017 Seventh International Conference on Image Processing Theory, Tools and Applications (IPTA)*. IEEE.

Rashid, S. F., Shafait, F. & Breuel, T. M. (2012) Scanning neural network for text line recognition, *2012 10th IAPR International Workshop on Document Analysis Systems*. IEEE.

Rastogi, M., Ali, S. A., Rawat, M., Vig, L., Agarwal, P., Shroff, G. & Srinivasan, A. (2020) Information Extraction From Document Images via FCA-Based Template Detection and Knowledge Graph Rule Induction, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*.

Ren, P., Fang, W. & Djahel, S. (2017) A novel YOLO-Based real-time people counting approach, *2017 International Smart Cities Conference (ISC2)*. IEEE.

Ren, S., He, K., Girshick, R. & Sun, J. (2015) Faster r-cnn: Towards real-time object detection with region proposal networks, *Advances in neural information processing systems*.

Ren, S., He, K., Girshick, R. & Sun, J. (2016) Faster R-CNN: towards real-time object detection with region proposal networks. *IEEE transactions on pattern analysis and machine intelligence*, 39(6), 1137-1149.

Ren, X. & Perrault, F. (1992) The typology of unknown words: an experimental study of two corpora, *COLING 1992 Volume 1: The 15th International Conference on Computational Linguistics*.

Reviso (2020) *What is an Invoice?* Available online: https://www.reviso.com/invoice/#:~:text=Definition%3A%20An%20invoice%20is%20a,to%20the%20seller's%20payment%20terms.&text=An%20invoice%20indicates%20that%20a%20buyer%20owes%20money%20to%20a%20seller [Accessed.

Reza, M. M., Bukhari, S. S., Jenckel, M. & Dengel, A. (2019) Table localization and segmentation using gan and cnn, *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)*. IEEE.

Reza, M. M., Rakib, M. A., Bukhari, S. S. & Dengel, A. (2018) A high-performance document image layout analysis for invoices. *DAS2018*.

Riba, P., Dutta, A., Goldmann, L., Fornés, A., Ramos, O. & Lladós, J. (2019) Table detection in invoice documents by graph neural networks, *2019 International Conference on Document Analysis and Recognition (ICDAR)*. IEEE.

Rigaud, C., Burie, J.-C. & Ogier, J.-M. (2017) Segmentation-free speech text recognition for comic books, *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*. IEEE.

Ritchie, J. & Lewis, J. (2013) QUALITATIVE RESEARCH PRACTICE A GUIDE FOR SOCIAL SCIENCE STUDENTS AND RESEARCHERS.

Rizvi, M., Raza, H., Tahzeeb, S. & Jaffry, S. (2019) Optical Character Recognition Based Intelligent Database Management System for Examination Process Control, *2019 16th International Bhurban Conference on Applied Sciences and Technology (IBCAST)*. IEEE.

Rousseau, D. M., Manning, J. & Denyer, D. (2008) 11 Evidence in management and organizational science: assembling the field's full weight of scientific knowledge through syntheses. *Academy of Management Annals*, 2(1), 475-515.

Roy, S. (2019) Denoising Sequence-to-Sequence Modeling for Removing Spelling Mistakes, *2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT)*. IEEE.

Roychoudhury, S., Bellarykar, N. & Kulkarni, V. (2016) *A NLP Based Framework to Support Document Verification-as-a-Service*. IEEE.

Sabbour, N. & Shafait, F. (2013) A segmentation-free approach to Arabic and Urdu OCR, *Document Recognition and Retrieval XX*. International Society for Optics and Photonics.

Sagar, B., Shobha, G. & Kumar, R. (2008) OCR for printed Kannada text to machine editable format using database approach. *WSEAS Transactions on Computers*, 7(6), 766-769.

Saluja, R., Adiga, D., Chaudhuri, P., Ramakrishnan, G. & Carman, M. (2017) Error detection and corrections in Indic OCR using LSTMs, *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*. IEEE.

Samanta, P. & Chaudhuri, B. B. (2013) A simple real-word error detection and correction using local word bigram and trigram, *Proceedings of the 25th conference on computational linguistics and speech processing (ROCLING 2013)*.

Sari, T. & Sellami, M. (2002) MOrpho-LEXical analysis for correcting OCR-generated Arabic words (MOLEX), *Proceedings Eighth International Workshop on Frontiers in Handwriting Recognition*. IEEE.

Saunders, M. N. K., Lewis, P. & Thornhill, A. (2019) *Research methods for business students*.

Schaback, J. & Li, F. (2007) Multi-level feature extraction for spelling correction, *IJCAI-2007 Workshop on Analytics for Noisy Unstructured Text Data*. Citeseer.

Seguin, P. (2019) *How AI Invoice Processing Works - ML, AI, etc. | Rossum.* Available online: https://rossum.ai/blog/how-ai-invoice-processing-works/ [Accessed.

Seppälä, T., Kenney, M. & Ali-Yrkkö, J. (2014) Global supply chains and transfer pricing. *Supply Chain Management: An International Journal*, 19, 445-454.

Shah, K., Sheth, J., Patel, M. & Lad, K. (2012) Comparative Study of Spell Checking Algorithms and Tools. *International Journal of Advanced Research in Computer Science*, 3(3).

Shah, S. (2017a) Spell checker and string matching using BK-trees, *International Conference on Academic Research in Engineering and Management*. 2017/04/30/. IETE, Lodhi Road, Delhi, India.

Shah, S. (2017b) SPELL CHECKER AND STRING MATCHING USING BK - TREES. *International Conference on Academic Research in Engineering and Management*, 5.

Shaker, M. & ElHelw, M. (2017) Optical character recognition using deep recurrent attention model, *Proceedings of the 2nd International Conference on Robotics, Control and Automation*.

Sharma, V. & Mishra, D. N. (2016) Using Big Data & Prediction Analysis (BDPA) in Effective Pricing Decisions, *Norwich Business School Colloquium 2016*. Norwich, UK, 18-Oct-2016.

Sharma, V. & Mishra, D. N. (2018) Impact of Automated Text Extraction from Invoices in Supply Chain Management, *Prolog Conference 2018*. Hull, UK, 29-Jun-2018.

Sharma, V. & Mishra, D. N. (2019) Big Data Analytics for Smart Operations in Service or Manufacturing Sector using Invoice Automation System, *FBLP PhD Colloquium 2019*. Hull, UK, 10-Jul-2019.

Shehzad, K., Ul-Hasan, A., Malik, M. I. & Shafait, F. (2020) Named Entity Recognition in Semi Structured Documents Using Neural Tensor Networks, *International Workshop on Document Analysis Systems*. Springer.

Shi, B., Bai, X. & Belongie, S. (2017) Detecting oriented text in natural images by linking segments, *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Shi, B., Bai, X. & Yao, C. (2016) An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. *IEEE transactions on pattern analysis and machine intelligence*, 39(11), 2298-2304.

Shin, H. (2012) Purchase Information Extraction Model From Scanned Invoice Document Image By Classification Of Invoice Table Header Texts. *Journal of Digital Convergence*, 10(11), 383-387.

Shinde, A. A. & Chougule, D. (2012) Text pre-processing and text segmentation for OCR. *International Journal of Computer Science Engineering and Technology*, 2(1), 810-812.

Shivakumara, P., Hemantha Kumar, G., Guru, D. S. & Nagablushan, P. (2005) *A new boundary growing and hough transform based approach for accurate skew detection in binary document images*. IEEE.

Sidhwa, H., Kulshrestha, S., Malhotra, S. & Virmani, S. (2018) Text extraction from bills and invoices, *2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN)*. IEEE.

Sievo (2018) *Spend Analysis 101 | Comprehensive Guide for Beginners.* Available online: https://sievo.com/resources/spend-analysis-101 [Accessed.

Sin, B.-K. S. B.-K., Kim, S.-K. K. S.-K. & Cho, B.-J. C. B.-J. (2002) Locating characters in scene images using frequency features. *Object recognition supported by user interaction for service robots*, 3, 0-3.

Singh, G., Dhandhania, K. & Jain, A. (2020) *Spell Checking and Correction.* Available online: https://www.commonlounge.com/discussion/5c3f34235fe943488e4e6c9906d64173 [Accessed.

Singh, R. & Kaur, M. (2010) OCR for Telugu script using back-propagation based classifier. *International Journal of Information Technology and Knowledge Management*, 2(2), 639-643.

Singh, S., Kumar, A., Shaw, D. K. & Ghosh, D. (2014) *Script separation in machine printed bilingual (Devnagari and Gurumukhi) documents using morphological approach*. IEEE.

Singh, S. & Singh, S. (2020) HINDIA: a deep-learning-based model for spell-checking of Hindi language. *Neural Computing and Applications*, 1-16.

Smith, M. A. & Kanade, T. (1995) Video skimming for quick browsing based on audio and image characterization. *Technical Report CMU-CS-95-186. School of Computer Science, Carnegie Mellon University*, ----.

Sorio, E., Bartoli, A., Davanzo, G. & Medvet, E. (2012) *A domain knowledge-based approach for automatic correction of printed invoices*.

Srigiri, S. & Saha, S. K. (2018) Spelling Correction of OCR-Generated Hindi Text Using Word Embedding and Levenshtein Distance, *International Conference on Nanoelectronics, Circuits and Communication Systems*. Springer.

SROIE (2020) *SROIE Dataset - Robust Reading Competition* (2021-02-10. Available online: https://rrc.cvc.uab.es/?com=contestant.

Stretch (2015) Supplier Qualification, Classification and Segmentation. *Stretch*.

Sundby, D. (2009) Spelling correction using N-grams. *Technical notes*, 3.

Suzuki, M., Tamari, F., Fukuda, R., Uchida, S. & Kanahori, T. (2003) INFTY: an integrated OCR system for mathematical documents, *Proceedings of the 2003 ACM symposium on Document engineering*.

Symon, G. & Cassell, C. (2012) Qualitative organizational research : core methods and current challenges, 523.

Taghva, K., Borsack, J. & Condit, A. (1994a) Expert system for automatically correcting OCR output, *Document Recognition*. International Society for Optics and Photonics.

Taghva, K., Borsack, J., Condit, A. & Erva, S. (1994b) The effects of noisy data on text retrieval. *Journal of the American Society for Information Science*, 45(1), 50-58.

Taghva, K. & Stofsky, E. (2001) OCRSpell: an interactive spelling correction system for OCR errors in text. *International Journal on Document Analysis and Recognition*, 3(3), 125-137.

Takeuchi, K. & Matsumoto, Y. (2000) Japanese OCR error correction using stochastic morphological analyzer and probabilistic word N-gram model. *International Journal of Computer Processing of Oriental Languages*, 13(01), 69-82.

Tang, P., Qiu, W., Huang, Z., Chen, S., Yan, M., Lian, H. & Li, Z. (2020) nAnomaly Detection in Electronic Invoice Systems Based on Machine Learning. *Information Sciences*.

Tang, P., Qiu, W., Yan, M., Huang, Z., Chen, S. & Lian, H. (2019) Association Analysis of Abnormal Behavior of Electronic Invoice Based on K-Means and Skip-Gram, *2019 IEEE Fourth International Conference on Data Science in Cyberspace (DSC)*. IEEE.

Tangsucheeva, R. & Prabhu, V. (2014) Stochastic financial analytics for cash flow forecasting. *International Journal of Production Economics*, 158, 65-76.

Tarawneh, A. S., Hassanat, A. B., Chetverikov, D., Lendak, I. & Verma, C. (2019) Invoice classification using deep features and machine learning techniques, *2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT)*. IEEE.

Teunissen, G. (2017) Helping SMEs Automate like Corporations: A Constraint Satisfaction Problem for Automatic Invoice Field Extraction. *University of Twente, The Netherlands, Tech. Rep*.

Thoben, K.-D., Wiesner, S. & Wuest, T. (2017) "Industrie 4.0" and Smart Manufacturing – A Review of Research Issues and Application Examples. *International Journal of Automation Technology*, 11, 4-16.

Tian, F., Wu, H. & Xu, B. (2021) Research on Fast Text Recognition Method for Financial Ticket Image. *arXiv preprint arXiv:2101.01310*.

Tian, S., Lu, S., Su, B. & Tan, C. L. (2013) Scene text recognition using co-occurrence of histogram of oriented gradients, *2013 12th International Conference on Document Analysis and Recognition*. IEEE.

Tian, Z., Huang, W., He, T., He, P. & Qiao, Y. (2016) Detecting text in natural image with connectionist text proposal network, *European conference on computer vision*. Springer.

Tranfield, D., Denyer, D. & Smart, P. (2003) Towards a methodology for developing evidence-informed management knowledge by means of systematic review. *British journal of management*, 14(3), 207-222.

Trepanier, S. (2019) Deep Dive: What is SmartScan and how it works, *How can we help you?*Available online: https://community.expensify.com/discussion/5541/deep-dive-what-is-smartscan-and-how-it-works.

Tsai, C. W., Lai, C. F., Chao, H. C. & Vasilakos, A. V. (2015) Big data analytics: a survey. *Journal of Big Data*, 2, 1-32.

Tsz Ching Sam, C. (2016) *A context -sensitive spell checker using trigrams and confusion Sets*Project Report. University of Manchester, 2016/02/05/. Available online:

http://studentnet.cs.manchester.ac.uk/resources/library/3rd-year-projects/2016/tsz.chan-5.pdf [Accessed.

Tutica, L., Vineel, K., Mishra, S., Mishra, M. K. & Suman, S. (2021) Invoice Deduction Classification Using LGBM Prediction Model, *Advances in Electronics, Communication and Computing*Springer, 127-137.

Uijlings, J. R., Van De Sande, K. E., Gevers, T. & Smeulders, A. W. (2013) Selective search for object recognition. *International journal of computer vision*, 104(2), 154-171.

Vadastreanu, A. M., Maier, D. & Maier, A. (2015) Is the Success Possible in Compliance with Ethics and Deontology in Business? *Procedia Economics and Finance*, 26, 1068-1073.

VATI (2021) *VATI Dataset* (2021-02-22. Available online: https://github.com/FuxiJia/InvoiceDatasets.

Vinitha, V. & Jawahar, C. (2016) Error detection in indic ocrs, *2016 12th IAPR Workshop on Document Analysis Systems (DAS)*. IEEE.

Vrasidas, C. (2000) Constructivism Versus Objectivism: Implications for Interaction, Course Design, and Evaluation in Distance Education, *International Journal of Educational Telecommunications*.

Wamba, S. F., Akter, S., Edwards, A., Chopin, G. & Gnanzou, D. (2015) How 'big data'can make big impact: Findings from a systematic review and a longitudinal case study. *International Journal of Production Economics*, 165, 234-246.

Wang, B., Xu, J., Li, J., Hu, C. & Pan, J.-S. (2017) Scene text recognition algorithm based on faster RCNN, *2017 First International Conference on Electronics Instrumentation & Information Systems (EIIS)*. IEEE.

Wang, J. & Hu, X. (2017) Gated recurrent convolution neural network for ocr, *Advances in Neural Information Processing Systems*.

Wang, K., Babenko, B. & Belongie, S. (2011) End-to-end scene text recognition, *2011 International Conference on Computer Vision*. IEEE.

Wang, R., Zhou, D. & He, Y. (2019) Open event extraction from online text using a generative adversarial network. *arXiv preprint arXiv:1908.09246*.

Wang, S., Wan, J., Li, D. & Zhang, C. (2016) Implementing Smart Factory of Industrie 4.0: An Outlook. *International Journal of Distributed Sensor Networks*, 12, 3159805.

Wang, W., Hong, W., Wang, F. & Yu, J. (2020) Gan-knowledge distillation for one-stage object detection. *IEEE Access*, 8, 60719-60727.

Wang, Y., Gui, G., Zhao, N., Yin, Y., Huang, H., Li, Y., Wang, J., Yang, J. & Zhang, H. (2018) Deep learning for optical character recognition and its application to VAT invoice recognition, *International Conference in Communications, Signal Processing, and Systems*. Springer.

Watcharabutsarakham, S. (2005) Spell checker for thai document, *TENCON 2005-2005 IEEE Region 10 Conference*. IEEE.

Wen, Y., Lu, Y., Yan, J., Zhou, Z., von Deneen, K. M. & Shi, P. (2011) An algorithm for license plate recognition applied to intelligent transportation system. *IEEE Transactions on intelligent transportation systems*, 12(3), 830-845.

Weng, Y. & Xia, C. (2019) A new deep learning-based handwritten character recognition system on mobile computing devices. *Mobile Networks and Applications*, 1-10.

Wickramarathna, S. & Ranathunga, L. (2019) Data Driven Approach to Brahmi OCR Error Correction and Sinhala Meaning Generation from Brahmi Character Array, *2019 19th International Conference on Advances in ICT for Emerging Regions (ICTer)*. IEEE.

Wilson, J. (2010) Essentials of Business Research - A Guide to Doing Your Research Project. *SAGE Publication*, 336.

Wu, F. & Huberman, B. A. (2007) Novelty and collective attention. *Proceedings of the National Academy of Sciences of the United States of America*, 104, 17599-601.

Xamena, E., Marmanillo, W. G. & Mechaca, A. L. (2019) Rebuilding the Story of a Hero: Information Extraction in Ancient Argentinian Texts, *V Simposio Argentino de Ciencia de Datos y GRANdes DAtos (AGRANDA 2019)-JAIIO 48 (Salta)*.

Yang, H., Siebert, M., Luhne, P., Sack, H. & Meinel, C. (2011a) Automatic lecture video indexing using video OCR technology, *2011 IEEE International Symposium on Multimedia*. IEEE.

Yang, H., Siebert, M., Luhne, P., Sack, H. & Meinel, C. (2011b) Lecture video indexing and analysis using video ocr technology, *2011 Seventh International Conference on Signal Image Technology & Internet-Based Systems*. IEEE.

Yang, J., Ren, P. & Kong, X. (2019) Handwriting Text Recognition Based on Faster R-CNN, *2019 Chinese Automation Congress (CAC)*. IEEE.

Ye, Q., Huang, Q., Gao, W. & Zhao, D. (2005) Fast and robust text detection in images and video frames. *Image and vision computing*, 23(6), 565-576.

Yeremia, H., Yuwono, N. A., Raymond, P. & Budiharto, W. (2013) Genetic algorithm and neural network for optical character recognition. *Journal of computer science*, 9(11), 1435.

Yi, F., Zhao, Y.-F., Sheng, G.-Q., Xie, K., Wen, C., Tang, X.-G. & Qi, X. (2019) Dual Model Medical Invoices Recognition. *Sensors*, 19(20), 4370.

Yin, Y., Zhang, W., Hong, S., Yang, J., Xiong, J. & Gui, G. (2019) Deep learning-aided OCR techniques for Chinese uppercase characters in the application of Internet of Things. *IEEE Access*, 7, 47043-47049.

Yindumathi, K., Chaudhari, S. S. & Aparna, R. (2020) Analysis of Image Classification for Text Extraction from Bills and Invoices, *2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*. IEEE.

Younes, B., Bouferguène, A., Al-Hussein, M. & Yu, H. (2015) Overdue Invoice Management: Markov Chain Approach. *Journal of Construction Engineering and Management*, 141, 04014062.

Yu, W., Lu, N., Qi, X., Gong, P. & Xiao, R. (2020) PICK: Processing Key Information Extraction from Documents using Improved Graph Learning-Convolutional Networks. *arXiv preprint arXiv:2004.07464*.

Yulianto, M. M., Arifudin, R. & Alamsyah, A. (2018) Autocomplete and spell checking Levenshtein distance algorithm to getting Text Suggest Error Data Searching in Library. *Scientific Journal of Informatics*, 5(1), 75.

Zaky, D. & Romadhony, A. (2019) An LSTM-based Spell Checker for Indonesian Text, *2019 International Conference of Advanced Informatics: Concepts, Theory and Applications (ICAICTA)*. IEEE.

Zhang, H., Liu, D. & Xiong, Z. (2017a) Cnn-based text image super-resolution tailored for ocr, *2017 IEEE Visual Communications and Image Processing (VCIP)*. IEEE.

Zhang, J., Chen, X., Hanneman, A., Yang, J. & Waibel, A. (2002) A robust approach for recognition of text embedded in natural scenes, *Object recognition supported by user interaction for service robots*. IEEE.

Zhang, M., Joshi, A., Kadmawala, R., Dantu, K., Poduri, S. & Sukhatme, G. S. (2009) Ocrdroid: A framework to digitize text using mobile phones, *International Conference on Mobile Computing, Applications, and Services*. Springer.

Zhang, P., Xu, Y., Cheng, Z., Pu, S., Lu, J., Qiao, L., Niu, Y. & Wu, F. (2020a) TRIE: End-to-End Text Reading and Information Extraction for Document Understanding, *Proceedings of the 28th ACM International Conference on Multimedia*.

Zhang, W. (2018) Online invoicing system based on QR code recognition and cloud storage, *2018 2nd IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC)*. IEEE.

Zhang, X.-Y., Bengio, Y. & Liu, C.-L. (2017b) Online and offline handwritten chinese character recognition: A comprehensive study and new benchmark. *Pattern Recognition*, 61, 348-360.

Zhang, Y., Bai, Y., Ding, M. & Ghanem, B. (2020b) Multi-task generative adversarial network for detecting small objects in the wild. *International Journal of Computer Vision*, 1-19.

Zhang, Y., Ming, Y. & Zhang, R. (2018) Object detection and tracking based on recurrent neural networks, *2018 14th IEEE International Conference on Signal Processing (ICSP)*. IEEE.

Zhao, J., Wang, Y., Xiao, B., Shi, C., Jia, F. & Wang, C. (2020) DetectGAN: GAN-based text detector for camera-captured document images. *International Journal on Document Analysis and Recognition (IJDAR)*, 23(4), 267-277.

Zhong, R. Y., Newman, S. T., Huang, G. Q. & Lan, S. (2016) Big Data for supply chain management in the service and manufacturing sectors: Challenges, opportunities, and future perspectives. *Computers & Industrial Engineering*, 101, 572-591.

Zhong, Z., Sun, L. & Huo, Q. (2019a) An anchor-free region proposal network for Faster R-CNN-based text detection approaches. *International Journal on Document Analysis and Recognition (IJDAR)*, 22(3), 315-327.

Zhong, Z., Sun, L. & Huo, Q. (2019b) Improved localization accuracy by LocNet for Faster R-CNN based text detection in natural scene images. *Pattern Recognition*, 96, 106986.

Zhou, X., Yao, C., Wen, H., Wang, Y., Zhou, S., He, W. & Liang, J. (2017a) EAST: an efficient and accurate scene text detector, *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

Zhou, Y., Porwal, U. & Konow, R. (2017b) Spelling correction as a foreign language. *arXiv preprint arXiv:1705.07371*.

Zhu, D., Li, T., Ho, D., Zhou, T. & Meng, M. Q. (2018) A novel ocr-rcnn for elevator button recognition, *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE.

Zhu, Y., Ma, C. & Du, J. (2019) Rotated cascade R-CNN: A shape robust detector with coordinate regression. *Pattern Recognition*, 96, 106964.

Zhuang, L., Bao, T., Zhu, X., Wang, C. & Naoi, S. (2004) A Chinese OCR spelling check approach based on statistical language models, *2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No. 04CH37583)*. IEEE.

Zhuang, L. & Zhu, X. (2005) An OCR post-processing approach based on multi-knowledge, *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems*. Springer.

Zidouri, A. (2004) ORAN: a basis for an Arabic OCR system, *Proceedings of 2004 International Symposium on Intelligent Multimedia, Video and Speech Processing, 2004.*: IEEE.