

University of Hull  
Faculty of Science and Engineering  
Department of Computer Science and Technology

**Deep Learning with Knowledge Graphs  
for Fine-Grained Emotion Classification in Text**

Annika M Schoene

Submitted in part fulfilment of the requirements for the degree of  
Doctor of Philosophy in Computer Science of the University of Hull,  
April 23, 2021



*To*

*Papa, Mama and Benthe*

*'Computing is too important to be left to men.'*

*- Karen Spärck*



## Acknowledgments

Undertaking this PhD has been a life-changing experience for me, and it would not have been possible without the help and support of many people. Especially during the past 12 months through a global pandemic and subsequent national lockdowns.

Firstly, I would like to thank the University of Hull for funding this work and providing me with a generous budget and access to research equipment and facilities.

I'm deeply grateful to my supervisor, Dr Nina Dethlefs, for her unwavering support, encouragement and invaluable feedback throughout my PhD journey. She has been incredibly kind and provided me with guidance in any way she could to help me develop as a researcher and person. I want to thank her for her patience when she had to reread the same misplaced quotation marks and subordinate clauses over and over again. I would also like to thank my co-supervisor, Dr Alexander Turner, for his insightful feedback on my work, guidance and interesting discussions on all things science. Furthermore, I want to thank Dr Yongqiang Cheng for chairing my PhD panels and his advice throughout my PhD. A special thanks goes to Dr Geeth de Mel, who first mentored me as an intern at IBM Research and later continued to guide me through the ups and downs of this PhD as my collaborator. I would also like to extend my thanks to the whole staff in the Department of Computer Science, the Viper Supercomputing team and administrative staff for their help and support. I want to thank Albert and Ashley for inspiring research chats, coffee breaks and making me laugh during stressful times. Furthermore, I want to thank my lab mates and colleagues in Milner and Turing Lab as well as the BDA research group for wonderful three years and many lunch breaks in Canham Turner. Also, I would like to thank everyone I have met at conferences and different summer schools for insightful feedback and conversations over the years.

I would like to thank my parents for always believing in me, being my safety net and having virtual breakfasts with me every Sunday. Thank you to Papa for patiently listening to my increasingly longer descriptions of my work and for giving me useful feedback. Also, I would like to thank my Mama for her rational advice and putting

the world to rights. I want to say thank you to the best sister in the world, Benthe, who has always inspired me with her determination and bravery. Thank you for spending countless hours on the phone with me and for giving me the courage to speak up. Also, I would like to thank my uncle, Klaus, for supporting my physical health and for his dry humour, which kept me going.

A very special thanks goes to my best friend Ange, not only for her kindness, positive attitude and rational advice but also for proofreading parts of this work, skyping me during national lockdowns to create a virtual office space and never losing faith in me. I would also like to thank my close friend and podcast co-host Sephora for her never-ending patience and support during my PhD, as well as her invaluable advice on all things careers, academia and industry. Furthermore, I want to thank Calvin for being there for me. For supporting me in every way possible and for keeping me grounded. Finally, I would like to thank all my friends in Germany, the UK and all over the world for countless adventures and holidays, interesting discussions and support over the last few years.

## Abstract

This PhD thesis investigates two key challenges in the area of fine-grained emotion detection in textual data. More specifically, this work focuses on (i) the accurate classification of emotion in tweets and (ii) improving the learning of representations from knowledge graphs using graph convolutional neural networks.

The first part of this work outlines the task of emotion keyword detection in tweets and introduces a new resource called the EEK dataset. Tweets have previously been categorised as short sequences or sentence-level sentiment analysis, and it could be argued that this should no longer be the case, especially since Twitter increased its allowed character limit. Recurrent Neural Networks have become a well-established method to classify tweets over recent years, but have struggled with accurately classifying longer sequences due to the vanishing and exploding gradient descent problem. A common technique to overcome this problem has been to prune tweets to a shorter sequence length. However, this also meant that often potentially important emotion carrying information, which is often found towards the end of a tweet, was lost (e.g., emojis and hashtags). As such, tweets mostly face also problems with classifying long sequences, similar to other natural language processing tasks. To overcome these challenges, a multi-scale hierarchical recurrent neural network is proposed and benchmarked against other existing methods. The proposed learning model outperforms existing methods on the same task by up to 10.52%. Another key component for the accurate classification of tweets has been the use of language models, where more recent techniques such as BERT and ELMO have achieved great success in a range of different tasks. However, in Sentiment Analysis, a key challenge has always been to use language models that do not only take advantage of the context a word is used in but also the sentiment it carries. Therefore the second part of this work looks at improving representation learning for emotion classification by introducing both linguistic and emotion knowledge to language models. A new linguistically inspired knowledge graph called RELATE is introduced. Then a new language model is trained on a Graph Convolutional Neural Network and compared against several other existing language models, where it is found that the proposed

embedding representations achieve competitive results to other LMs, whilst requiring less pre-training time and data. Finally, it is investigated how the proposed methods can be applied to document-level classification tasks. More specifically, this work focuses on the accurate classification of suicide notes and analyses whether sentiment and linguistic features are important for accurate classification.

# Contents

<b>Dedication</b>	<b>i</b>
<b>Acknowledgements</b>	<b>ii</b>
<b>Abstract</b>	<b>v</b>
<b>List of Tables</b>	<b>xv</b>
<b>List of Figures</b>	<b>xxii</b>
<b>Acronyms</b>	<b>1</b>
<b>1 Introduction</b>	<b>4</b>
1.1 Challenges in Sentiment Analysis . . . . .	6
1.1.1 Hypotheses . . . . .	8
1.2 Contributions . . . . .	9
1.2.1 Organisation . . . . .	10

---

<b>2</b>	<b>Related Work</b>	<b>11</b>
2.1	Emotion Theories . . . . .	13
2.2	Emotion and Sentiment Analysis . . . . .	17
2.2.1	Approaches to Sentiment Analysis . . . . .	20
2.2.2	Fine-grained emotion classification . . . . .	29
2.3	Deep Learning . . . . .	31
2.3.1	Graph Neural Networks . . . . .	37
2.3.2	Recurrent Neural Networks . . . . .	39
2.3.3	Attention Mechanisms . . . . .	46
2.3.4	Modelling Long Sequences . . . . .	51
2.4	Language Models . . . . .	59
2.4.1	Static Word Embeddings . . . . .	59
2.4.2	Contextualised or Dynamic Word Embeddings . . . . .	61
2.4.3	Sentiment Embeddings . . . . .	67
2.5	Knowledge Representation . . . . .	73
2.5.1	Emotion Knowledge Bases . . . . .	76
2.5.2	Creating Knowledge Graphs . . . . .	77
2.5.3	Knowledge Graph Embeddings . . . . .	80
2.5.4	Knowledge Embeddings in SA . . . . .	89
2.6	Conclusion . . . . .	92

---

<b>3</b>	<b>Fine-grained Emotion Classification in Tweets</b>	<b>94</b>
3.1	Ethical considerations in Data Collection . . . . .	95
3.2	Data . . . . .	98
3.3	Dilated LSTM . . . . .	102
3.3.1	Experimental setup . . . . .	103
3.3.2	Results . . . . .	104
3.3.3	Conclusion . . . . .	105
3.4	Bidirectional Dilated LSTM with Attention . . . . .	106
3.4.1	Bidirectional Dilated LSTM with Attention . . . . .	106
3.4.2	Experimental setup . . . . .	109
3.4.3	Results . . . . .	112
3.4.4	Conclusion . . . . .	118
3.5	Conclusion . . . . .	119
<b>4</b>	<b>Learning SSE representations using RELATE</b>	<b>121</b>
4.1	Creating an emotion knowledge graph . . . . .	123
4.1.1	Preprocessing . . . . .	124
4.1.2	Obtaining Triples . . . . .	127
4.1.3	Qualitative evaluation of RELATE . . . . .	130
4.1.4	Discussion . . . . .	136
4.2	Learning Embedding Representations . . . . .	137

---

4.2.1	Experimental setup . . . . .	139
4.2.2	Results and Evaluation . . . . .	141
4.3	Conclusion . . . . .	154
<b>5</b>	<b>AI for Social Good: Suicide Note Classification</b>	<b>156</b>
5.1	Related Work . . . . .	158
5.2	Data . . . . .	164
5.3	Linguistic Analysis . . . . .	168
5.3.1	Analysis for Experiment 1 . . . . .	168
5.3.2	Analysis for Experiments 2, 3 and 4 . . . . .	176
5.3.3	Cohen's d effect size . . . . .	186
5.4	Experiment 1: Bidirectional LSTM with attention . . . . .	189
5.4.1	Results and Evaluation . . . . .	190
5.5	Experiment 2: Dilated LSTM with attention . . . . .	192
5.5.1	Results and Evaluation . . . . .	194
5.6	Experiment 3 and 4: Dilated LSTM with ranked units . . . . .	198
5.6.1	Results and Evaluation . . . . .	200
5.7	Conclusion and future work . . . . .	206
<b>6</b>	<b>Conclusion</b>	<b>208</b>
6.1	Summary of contributions . . . . .	208
6.2	Limitations and future work . . . . .	210







# List of Tables

2.1	Ekman’s extended emotions . . . . .	15
3.1	All collected EEK data over time . . . . .	101
3.2	EEK Dataset . . . . .	102
3.3	Description of experiment series for the Dilated LSTM . . . . .	104
3.4	Results of experiments using a dilated LSTM using test set accuracy % . . . . .	105
3.5	Comparison of dataset distributions. . . . .	110
3.6	Results of test accuracy in %, precision, recall and F-1 score for capped sequences (IEST Dataset). . . . .	113
3.7	Results of test accuracy in %, precision, recall and F-1 score for capped sequences (EEK dataset). . . . .	113
3.8	Results of test accuracy in %, precision, recall and F-1 score full length (IEST dataset). . . . .	114
3.9	Results of test accuracy in %, precision, recall and F-1 score long (EEK dataset). . . . .	114

3.10	Evaluation metrics per emotion label - BiDLSMT with attention in % (IEST dataset). . . . .	116
3.11	Evaluation metrics per emotion label - BiDLSMT with attention in % (EEK dataset). . . . .	116
4.1	Types of Sentences in Dataset . . . . .	126
4.2	Overall number of triples in the knowledge base. . . . .	131
4.3	Emotion keywords that are part of a triple . . . . .	132
4.4	Overview of the different resources that were used as input to the GCN . . . . .	140
4.5	Experiment results for the GCN embeddings and existing language models using f-1 scores. . . . .	141
4.6	Overview of the different settings, training times (hours :minutes) and model parameters for each embedding layer. . . . .	142
5.1	Overview over the different outline resources for the GSN3 dataset . . . . .	165
5.2	Overview over the different datasets that were used in the following experiments. . . . .	167
5.3	Linguistic Features across all three corpora as provided by LIWC (Pennebaker et al. 2014). . . . .	169
5.4	Average reference to topic per note. . . . .	175
5.5	Sentiment features in % across all three corpora based on the work by Pestian et al. (2012). . . . .	176

5.6	LIWC Dimension Analysis for Experiment 2, 3 and 4 showing the average number of occurrence per note. . . . .	176
5.7	LIWC Function and Content Words for Experiment 2, 3 and 4. . . . .	179
5.8	LIWC Affect Analysis for Experiment 2, 3 and 4 . . . . .	181
5.9	LIWC Social Processes for Experiment 2, 3 and 4 . . . . .	182
5.10	LIWC Psychological Processes for Experiment 2, 3 and 4 . . . . .	183
5.11	LIWC Personal Concerns for Experiment 2, 3 and 4. . . . .	184
5.12	LIWC Time orientation for Experiment 2, 3 and 4. . . . .	185
5.13	Table 8: Cohens' d effect size for pairwise significance testing of linguistic features. . . . .	188
5.14	Description of experiment series. . . . .	189
5.15	Experiment results comparing a vanilla LSTM with a bidirectional LSTM with attention for classification from text only, emotions only and text and emotions. All results are the test accuracy in %. . . . .	190
5.16	Experiment results using test accuracy and F-1 score of different learning models. . . . .	194
5.17	Comparison of F-1 scores per dataset for both a Dilated LSTM with attention and BiLSTM with attention . . . . .	195
5.18	Results of Experiment 3 using precision, recall and f1-score . . . . .	200
5.19	Results of Experiment 2 using precision, recall and f1-score . . . . .	203



# List of Figures

2.1	A graphic showing Ekman's six basic emotions Ekman (1999)	15
2.2	Wheel of Emotion adapted from Plutchik (2001)	16
2.3	Sentiment Analysis Layers taken from Cambria, Poria, Gelbukh & Thelwall (2017)	17
2.4	Example of a 'negative' tweet in statistical SA	23
2.5	Example of a tweet containing the keyword 'happy' in knowledge-based SA	26
2.6	Example of a tweet containing the keyword 'fear' in a hybrid approach SA	29
2.7	MLP (adapted from (Gardner & Dorling 1998))	35
2.8	Simplified Convolutional Neural Network architecture	36
2.9	Simplified Memory Network architecture	36
2.10	Simplified Graph Neural Network architecture (adapted from Kipf & Welling (2016))	38
2.11	Unfolded RNN (adapted from (Goodfellow et al. 2016))	40
2.12	LSTM Cell (adapted from (Goodfellow et al. 2016))	41

2.13 LSTM full process (adapted from (Goodfellow et al. 2016)) . . . . .	43
2.14 Adapted from Goodfellow et al. (2016). Bidirectional RNN with $L$ hidden layers. . . . .	46
2.15 Simplified adaptation of the Transformer Network by Vaswani et al. (2017) showing (a) the overall Encoder-Decoder architecture and (b) the architecture within each encoder and decoder. . . . .	49
2.16 Bidirectional LSTM with attention mechanism (adapted from Yang et al. (2016)) . . . . .	51
2.17 Architecture of the Bidirectional Language Model ‘ELMO’. . . . .	64
2.18 Simplified architecture of the ‘BERT’ Language Model adapted from Devlin et al. (2018) . . . . .	65
2.19 Architecture of Language Model ‘ERNIE’ adapted from Zhang, Han, Liu, Jiang, Sun & Liu (2019) . . . . .	89
2.20 Architecture of SenticLSTM adapted from Ma, Peng, Khan, Cambria & Hussain (2018). . . . .	90
3.1 Example of a tweet . . . . .	100
3.2 Example of a tweet . . . . .	100
3.3 Dilated LSTM with an additional embedding layer. . . . .	103
3.4 Bidirectional DLSTM with attention. . . . .	108
3.5 Example of a tweet shown to annotators . . . . .	115
3.6 BiDLSTM attention (IEST) . . . . .	115
3.7 BiDLSTM attention (EEK) . . . . .	115



---

3.8	BiDLSTM attention (IEST) . . . . .	117
3.9	BiDLSTM attention (EEK) . . . . .	117
3.10	A tweet illustrating the difficulty of the task for a human annotator to choose one emotion keyword. . . . .	117
3.11	Visualisation of IEST Emotion labels based on the probability of accurate prediction - BiDLSTM with attention . . . . .	118
3.12	Visualisation of EEK Emotion labels based on the probability of accurate prediction - BiDLSTM with attention . . . . .	118
4.1	Example of a Emoji to textual description representation . . . . .	125
4.2	Example of a clause annotated with dependencies and the resulting triple. . . . .	130
4.3	Example of ‘angry’ relations in the knowledge graph. . . . .	132
4.4	Example of ‘sad’ relations in the knowledge graph. . . . .	133
4.5	Example of ‘happy’ relations in the knowledge graph . . . . .	133
4.6	Example of ‘disgust’ relations in the knowledge graph. . . . .	134
4.7	Example of ‘surprise’ relations in the knowledge graph. . . . .	134
4.8	Example of ‘fear’ relations in the knowledge graph. . . . .	135
4.9	Example of a feeling conveyed in tweet that is not captured . . . . .	136
4.10	Example of a clause going through preprocessing of ‘RELATE’ (orange box to the left) and then input into the GCN (blue box to the right - graphic adapted from Yao et al. (2019a)). . . . .	139

4.11	Visualisation of the emotion keyword ‘joy’ in the embedding representation of the GCN using RELATE. . . . .	143
4.12	Visualisation of the emotion keyword ‘joy’ in the embedding representation of GloVe. . . . .	144
4.13	Visualisation of the emotion keyword ‘fear’ in the embedding representation of the GCN using RELATE. . . . .	145
4.14	Visualisation of the emotion keyword ‘fear’ in the embedding representation of GloVe. . . . .	146
4.15	Visualisation of the emotion keyword ‘anger’ in the embedding representation of the GCN using RELATE. . . . .	147
4.16	Visualisation of the emotion keyword ‘anger’ in the embedding representation of GloVe. . . . .	147
4.17	Visualisation of the emotion keyword ‘surprise’ in the embedding representation of the GCN using RELATE. . . . .	148
4.18	Visualisation of the emotion keyword ‘surprise’ in the embedding representation of GloVe. . . . .	149
4.19	Visualisation of the emotion keyword ‘disgust’ in the embedding representation of the GCN using RELATE. . . . .	150
4.20	Visualisation of the emotion keyword ‘disgust’ in the embedding representation of GloVe. . . . .	150
4.21	Visualisation of the emotion keyword ‘sad’ in the embedding representation of the GCN using RELATE. . . . .	151
4.22	Visualisation of the emotion keyword ‘sad’ in the embedding representation of GloVe. . . . .	152

---

5.1	Example of a Suicide Note. . . . .	166
5.2	Example of a Depressed Note. . . . .	166
5.3	Example of a Love Note. . . . .	166
5.4	Example of a Last Statement. . . . .	166
5.5	Example of an excerpt of a neutral post. . . . .	166
5.6	Example of a GSN1 note . . . . .	191
5.7	Example of a DL1 note . . . . .	191
5.8	Example of a LH note . . . . .	191
5.9	Sentiment feature heatmap . . . . .	191
5.10	Legend for the sentiment feature heatmap . . . . .	191
5.11	Confusion Matrices of the predicted test set labels of the BiLSTM and DLSTM with attention. . . . .	195
5.12	Comparison of attention weights in a correctly classified LS note and wrongly classified LS note . . . . .	196
5.13	Comparison of attention weights in a correctly classified GSN2 note and wrongly classified DL1 note . . . . .	197
5.14	Comparison of attention weights in a correctly classified DL1 note and wrongly classified DL1 note . . . . .	197
5.15	Maximum Entropy Classifier . . . . .	201
5.16	Bidirectional LSTM with attention . . . . .	201
5.17	Dilated LSTM with ranked units . . . . .	201

5.18	Example of a correctly classified GSN3 note . . . . .	201
5.19	. . . . .	202
5.20	. . . . .	202
5.22	Maximum Entropy Classifier . . . . .	204
5.23	Bidirectional LSTM with attention . . . . .	204
5.21	Dilated LSTM with ranked units . . . . .	204
5.25	. . . . .	205
5.26	. . . . .	205
5.24	Example of a correctly classified NEU2 note . . . . .	205

# Acronyms

**ABSA** Aspect-based Sentiment Analysis

**AI** Artificial Intelligence

**BERT** Bidirectional Encoder Representation from Transformers

**CNN** Convolutional Neural Network

**CWA** Closed World Assumption

**EEK** Ekman Emotion Keyword

**ELMO** Embeddings for Language Models

**ERNIE** Enhanced Language Representation with Informative Entities

**GloVe** Global Vectors

**GRU** Gated Recurrent Unit

**GA** Genetic Algorithm

**GPT** Generative Pre-training

**GCN** Graph Convolutional Neural Networks

**GNN** Graph Neural Networks

**KB** Knowledge base

**KG** Knowledge Graph

**KGE** Knowledge Graph Embedding

**KRL** Knowledge Representation Learning

**LDA** Latent Dirichlet Allocation

- LIWC** Linguistic Inquiry and Word Count
- LM** Language Model
- LSA** Latent Semantic Analysis
- LSTM** Long-Short-Term-Memory
- MLP** Multilayer Perceptron
- MOO** Multi-objective optimisation
- NER** Named Entity Recognition
- NLG** Natural Language Generation
- NLP** Natural Language Processing
- NLTK** Natural Language Tool Kit
- NLU** Natural Language Understanding
- OWA** Open World Assumption
- POS** Part-Of-Speech-Tagging
- RNN** Recurrent neural networks
- ReLU** Rectified Linear unit
- SA** Sentiment Analysis
- SVM** Support Vector Machine
- TABSA** Target-Aspect-Based-Sentiment Analysis
- tanh** hyperbolic tangent function
- TDA** Translational Distance Model Approaches

**TTR** Type/Token Ratio

**SDG** Sustainable Development Goals

**SMA** Semantic Matching Approaches

**SSE** Sentiment Specific Embedding

**UN** United Nations

**WHO** World Health Organisation

# Chapter 1

## Introduction

Natural Language Processing (NLP) is a subtopic in the research area of Artificial Intelligence (AI) (Goodfellow et al. 2016) and has been defined by Cambria & White (2014) as *‘a range of computational techniques for the automatic analysis and representation of human language’*. This field encompasses several different tasks and techniques, which include the grammatical representation of language (syntax), the meaning of words and sentences (semantics), the knowledge that is not explicitly stated (pragmatics) as well as pronunciation (phonetics) and the various structures and interactions of different language data (discourse) (Chowdhury 2003). Over recent years deep learning techniques have been successfully applied to tasks such as Machine Translation (Sutskever et al. 2014), Natural Language Generation (NLG) (Konstas et al. 2017) and Sentiment Analysis (SA) (Dos Santos & Gatti 2014), producing consumer products such as Amazon’s voice assistant Alexa (Amazon 2018a) or Google Voice Assistant (Google 2020). The increasing usage of statistical methods in NLP (Gudivada et al. 2015, Manning & Schütze 1999) has strongly influenced both syntax and semantics research, and even more progress has been made since the advent of deep learning techniques. Nowadays successful NLP tasks rely heavily on the statistical representation of words (Cambria & White 2014) and



this success is also due to a large amount of data available to researchers, corporate entities and other organisations as the content is created by users across multiple online platforms (Bravo-Marquez et al. 2014). However, the produced content has been created with the intention of being understood and read by other humans and not necessarily machines. Therefore current methods are challenged by this largely unstructured data, and any automatic analysis requires algorithms to develop a more in-depth understanding of natural language (Cambria & White 2014). In particular, deep learning techniques have hugely contributed to this deeper understanding by providing researchers with the opportunity to find patterns within large datasets (Najafabadi et al. 2015). This also enables researchers to develop models that can make not only accurate classifications but also make structured predictions as well as learn representations on this data (LeCun et al. 2015). Despite this, there are still challenges left within the field of NLP where data-hungry deep learning techniques do not perform as well on sparse or noisy data sets (Najafabadi et al. 2015) and struggle to infer knowledge from natural language (Cambria & White 2014).

Knowledge representation and meaning research is another research area that is traditionally more focused on the machine comprehension of the text it is presented with (Winograd 1972) and used to rely largely on first-order logic (Fox 2015). This research area relies on semantic theories (Sowa 2014), that can include probabilistic latent semantic analysis, which is a technique used *‘for a better representation of sparse information in a text block, such as a sentence or a sequence of sentences’* (Brants et al. 2002). Other techniques employed include lexicons (Miikkulainen 1993) or ontologies which have been used for parsing natural language for applications such as search engines (Busch et al. 2006). However, it has been argued that ultimately understanding natural language will be impacted by advances in deep learning and that systems combining representation learning and complex reasoning will further advance Artificial Intelligence research (LeCun et al. 2015).

Both statistical methods and semantic theories have been applied to the field of SA, where researchers aim to accurately detect polarities or emotions within language data. These two tasks are closely associated with other subtasks in SA called opinion and subjectivity detection (Cambria 2016). It has been argued by Minsky (2006) that this growing subfield of NLP has not just a range of commercial and health applications, but is also key in further advancing the field of Artificial Intelligence.

The following section will outline some of the current challenges in the field of Sentiment Analysis.

## 1.1 Challenges in Sentiment Analysis

There are growing numbers of social media users, who increasingly express their opinions, beliefs and attitudes in online posts towards a range of different topics, events and products (Ravi & Ravi 2015). As a result of this trend, it becomes ever more important to not only accurately classify the topics in the posts, but also the sentiment and emotion that is conveyed explicitly or implicitly (Gievska et al. 2014). Successfully applying models to complete SA tasks holds increasing value across many different sectors such as the financial or retail sector where companies or organisations aim to improve the marketing of their products and services (Recupero et al. 2015). Furthermore, it also impacts on areas such as classifying depression (Morales & Levitan 2016) or other mental health (Ji, Pan, Li, Cambria, Long & Huang 2020) condition. There are, however, several challenges within the field of SA that currently limit the performance of SA models.

Recurrent neural networks (RNN) are well suited towards natural language processing tasks such as SA due to their ability to handle sequential data, and there are still shortcomings which ultimately effect the accurate classification of longer sequences. Traditionally, tweets have been categorised as short sequences

or sentence-level sentiment analysis (Kouloumpis et al. 2011), however, it could be argued that this should no longer be the case especially since Twitter increased its allowed character limit from 140 to 280 (Twitter 2018a). Subsequently, it could be argued that tweets mostly face also problems with classifying long sequences, similar to other natural language processing tasks (Hochreiter & Schmidhuber 1997a). As a result of this it is even more important to classify long sequences accurately, because often important information that indicates a person’s emotions, attitude or opinions are expressed through the use of emojis and often appear towards the end of a tweet (Novak et al. 2015). Furthermore, most current large-scale SA methodologies predominantly focus on the classification of polarities or pre-determined moods (Cambria, Poria, Gelbukh & Thelwall 2017). This is also true for social media data, where collecting and annotating large scale datasets for fine-grained emotions is often time-consuming and costly. However, for supervised NLP tasks annotating data is essential, where many researchers have to rely on external companies that provide annotation services (Glorot et al. 2011). Nonetheless, data produced on social media platforms such as Twitter are still important for generating insights into political events, product marketing or mental health. Therefore, only utilising polarities may lead to not generating the right insights into the data and may make people draw misleading conclusions from it (Wang, Liakata, Zubiaga & Procter 2017).

At the same time, SA has been performed with not only machine- and deep learning, but also knowledge-based approaches, where the former yield better classification accuracy and the latter produce better generality (Ravi & Ravi 2015).

Knowledge-based approaches to SA mainly rely on ontologies (Grassi 2009), lexicons (Mohammad 2016), knowledge graphs (Cambria et al. 2018) and other semantic theories. Traditional SA largely relies on explicit knowledge used in text data through affective or polarised words as well as their co-occurrence frequencies in a sequence (context) (Recupero et al. 2015). However, recent years has seen an

increased effort of including external knowledge into embedding representations (Liang et al. 2019) that are commonly used as input into neural networks. However, one of the major downsides has been that most learned embedding representations or larger Language Model (LM)s focus solely on incorporating context and not sentiment. This has often led to words carrying opposite affective meaning to occupy a similar or the same vector space in those LMs or embedding representations (Zhang, Wu & Dou 2019) and led to worse results on various SA tasks (Tang, Wei, Qin, Yang, Liu & Zhou 2015). In order to overcome this problem, previous research has often used approaches such as post-modifying existing embedding methods (Yu et al. 2017) and LMs (Xu et al. 2020). However, oftentimes the resources used for these approaches have been limited in themselves, where resources are either large in size, including a great number of words but only cover a limited amount of polarities/emotions. On the other hand, smaller resources require costly human annotation but have the benefit of covering fine-grained emotions. Therefore using knowledge resources to learn new embedding representations or LMs that include any sentiment, is either limited by low coverage of polarities/emotions or the overall size of the resource.

### 1.1.1 Hypotheses

In order to overcome the aforementioned challenges, this thesis will be investigating the following two hypotheses:

1. *It is hypothesised that tweets should no longer be treated as short sequences or sentence-level SA, where when the full sequence length is utilised better classification accuracies can be achieved.*

This will be explored by proposing the use of hierarchical multi-scale RNNs that can overcome the ‘vanishing’ and ‘exploding’ gradient descent problem

in the task of fine-grained emotion classification in tweets. This assumption is also extended to document-level classification tasks, such as suicide note classification.

2. *It is hypothesised that learning word embeddings that not only incorporate context but also emotion/sentiment knowledge will lead to better performance in fine-grained emotion classification.*

This will be implemented by using a knowledge resource (e.g.: a knowledge graph) that includes fine-grained emotions and a large number of words.

## 1.2 Contributions

This thesis makes several contributions to overcome the aforementioned challenges in SA.

- Establishing emotion classification in tweets as a long sequence learning problem and outlining current challenges in SA (Schoene 2020).
- Introduction of a new Twitter dataset for fine-grained emotion classification and the proposal of a new learning model, called the Bidirectional Dilated LSTM with attention (Schoene et al. 2020).
- A new large, linguistically inspired knowledge graph that was build from Twitter data and utilises emotion knowledge. Furthermore, the use of Graph Convolutional Neural Network for learning embedding representations is introduced and compared against existing methods (Schoene et al. n.d.).
- First the use of deep learning models, specifically recurrent neural networks, for the task of suicide note classification is established. Then the dilated LSTM is applied to the task of distinguishing suicide notes from depressed notes

and last statements. Furthermore, the dilated LSTM with ranked units is proposed alongside a new task in suicide note classification, where the focus is on distinguishing suicide notes from other types of social media posts. Finally, a linguistic analysis is conducted for each experiment series. (Schoene & Dethlefs 2018, Schoene, Turner & Dethlefs 2019, Schoene, Lacey, Turner & Dethlefs 2019).

### 1.2.1 Organisation

This thesis is organised as follows: Chapter 2 gives an overview over the literature and closely related work in Sentiment Analysis, Deep Learning, Language Modelling and Knowledge Representation. Chapter 3 first introduces the data used in this work and the task of fine-grained emotion classification. Then a hierarchical RNN architecture is proposed to overcome the issue of accurately classifying emotion keywords in tweets and two experiment series are conducted. Chapter 4 introduces a new knowledge graph resource, called RELATE, and proposes a new method of learning embedding representations that include emotion keywords. Work in Chapter 5 shows how the methods proposed in Chapter 3 can be applied to a document-level classification task of suicide note classification. Finally, a conclusion will be provided that summarised the contributions of this thesis and describes future work in this field.

# Chapter 2

## Related Work

Emotion and sentiment have been widely investigated in several different research areas, which include but are not limited to psychology and neuroscience (Panksepp 2004, Ekman & Davidson 1994). However, it has always been hard to exactly define what an emotion is and how emotion is different from other affective states, such as mood or affect. Therefore, scientists from a variety of disciplines are still trying to define conclusively what an emotion is and how it can be described (Coppin & Sander 2016*a*). For this work, the definition of emotion will be taken from the fields of psychology and neuroscience, where an emotion has been defined as a concept closely associated to the human nervous system (Ekman & Davidson 1994). Emotions are also an essential part of human communication and behaviourism, where emotion, cognition and resulting action are seen as feedback loops, and it is seen desirable to accurately interpret and/or identify them in many research areas of Computer Science (Cambria et al. 2019, Gill et al. 2008). In Computer Science, the field of ‘Affective Computing’ has emerged as a new discipline specifically to build and design technologies and artificial intelligence that can process and/or express human emotion. Work by Picard (2000) has often been accredited with creating this new branch of computer science that encompasses many sub-disciplines, including

natural language processing. NLP itself is a broad field that encompasses several other disciplines that work towards understanding human language in speech and text data (Russell & Norvig 2002). Both emotion and sentiment analysis have often been used in the same context or sometimes even interchangeably in NLP, where it has previously been pointed out that this inconsistent use of terminology can not only be confusing but also results in a lack of understanding how the two concepts relate to one another (Munezero et al. 2014). It is believed that emotions are predecessors of sentiments and only occur for a short period of time, whilst sentiments are longer-lasting (Ben-Zeev & Ben-Zeev 2001). Therefore in this work, firstly the field of sentiment analysis will be reviewed in order to get a comprehensive insight into the field. Then there will be an overview of the literature in emotion classification, which could be seen as a subtask of SA. Over recent years, sentiment analysis tasks have often become closely associated with NLP tasks such as polarity detection or opinion mining (Munezero et al. 2014). However, in section 2.2, it will be outlined that Sentiment Analysis encompasses a far wider-ranging variety of tasks that are performed using textual data. At the same time, there has been a huge increase in popularity of statistical methods, which are now not only used in a variety of SA tasks but also in NLP (Young et al. 2018). More specifically Deep Learning methods, such as Recurrent Neural Networks, have been utilised due to their ability to handle sequential data (Hochreiter & Schmidhuber 1997*b*). However, there have also been many research efforts to include knowledge into deep learning methods as well as SA tasks by developing lexicons and knowledge graphs for this task. These resources have then been used to generate new embedding representations (Ma, Peng & Cambria 2018) or recognise entities in textual data (Exner & Nugues 2012).

This chapter will firstly describe existing emotion theories that are used in a variety of research areas, including emotion and sentiment analysis. Then an in-depth analysis of the field of emotion and sentiment analysis will be given, including



an overview of where the aforementioned emotion theories are used in research. Secondly, a review of existing deep learning methodologies will be given, and it will be illustrated how these methodologies are used in learning embedding representations. Then it will be shown how these statistical methods are used in combination with knowledge-based approaches. Finally, it will be examined how the combination of knowledge graphs and deep learning is used for learning sentiment specific embedding representations.

## 2.1 Emotion Theories

There are two main schools of thought within the field of Emotion Measurement, which can be divided into ‘Theoretical Emotion Theory’ (Meiselman 2016) and ‘Constructionist Emotion Theory’ (Averill 1980). Both of these theories are in stark contrast to each other where Theoretical Emotion Theorists argue for an innate emotion essence each human has and Constructionist Theorists believe that emotion is developed through social interactions (Meiselman 2016).

**Theoretical Emotion Theories** One of the main challenges of defining the term emotion is establishing boundaries between the concept of ‘emotion’ and other affective states such as mood or attitude (Meiselman 2016). Within the field of Theoretical Emotion Theory, it has been outlined by Coppin & Sander (2016b) that there are three leading emotion theories that can be categorised into basic (Izard 1992), dimensional and appraisal theories. All three theories have a common set of characteristics as defined by Meiselman (2016): “(1) *expression*, (2) *action tendency*, (3) *bodily reaction*, (4) *feeling*, and (5) *appraisal*”. It has been argued that these common characteristics have greatly contributed to the overall definition of emotions in this research area (Meiselman 2016). The most commonly accepted and used definition of emotion has been coined by Sander (2013) who described

*“emotion as an event-focused, two-step, fast process consisting of (1) relevance-based emotion elicitation mechanisms that (2) shape multiple emotional responses (i.e., action tendency, automatic reaction, expression, and feeling)”*.

**Constructionist Theories** Researchers working in the field of constructionist theories have long opposed the definition given by theoretical emotion theory researchers, and two alternative key assumptions have been developed. Firstly it is argued that all emotions are highly variable and domain-specific to a given context (Barrett 2009). This also entails the belief that there is no direct relationship that links emotion to a word, which is why there are so many different behaviours or physiological patterns (Lindquist et al. 2013). Secondly, it has been proposed that any kind of occurrence within or between an emotion category and a word, thoughts or beliefs are part of a more *“fundamental common or domain-general process within the nervous system”* (Meiselman 2016). Therefore the existence of an emotion or mental category has been defined as the brain’s response to assign meaning to any sensory input (Barrett et al. 2016).

Although both theories have appealing ideas and arguments, it is beyond the scope of this research to elaborate on the credibility of either theory. However, it is important to note that the field of traditional emotion theory has contributed to the interdisciplinary field of Affective Science, which is a multi-disciplinary research area that includes collaborations of Computer Science, Linguistics and Psychology (Meiselman 2016). This trend has also occurred within the field of Computer Science where ‘Affective Computing’ was conceived as a research area combining knowledge from Computer Science, Engineering, Neuroscience and Psychology (Calvo et al. 2015). Cambria (2016) argues that SA is also a part of this broader field, which uses a range of different techniques to develop models that can detect affect in language data. Therefore in this work, the assumptions and definitions of ‘theoretical emotion theories’ will be used. Two theories have been used predominately in the field of

Sentiment Analysis, namely Ekman’s six basic emotions (Ekman et al. 1987) and Plutchik’s eight basic emotions (Plutchik 1984).

**Ekman’s Six Basic Emotions** This popular emotion theory was introduced by Ekman et al. (1987), who argued that there are six basic emotions based on facial expressions. The key argument of this theory is that the six basic emotions are universal across the world and can be found in any human species (Ekman 1992). Furthermore, Ekman (1999) argues that emotions are essential for humans to develop and handle interpersonal relationships. This

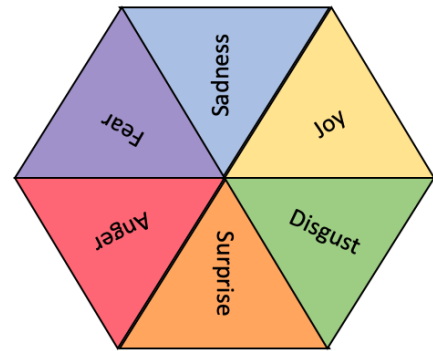


Figure 2.1: A graphic showing Ekman’s six basic emotions Ekman (1999)

research was initially proposed and tested for the emotions of facial expressions; however, it was found that the expression of emotions in language and communication is just as important (Ekman 1999). Whilst the list of basic emotions has been extended in his later work (see Table 2.1) (Ekman 1999) most researchers use the original six basic emotions for fine-grained emotion detection in textual data (Strapparava & Mihalcea 2008, Balabantaray et al. 2012).

Extended Emotions
Amusement
Contempt
Contentment
Embarrassment
Excitement
Guilt
Pride in achievement
Relief
Satisfaction
Sensory pleasure
Shame

Table 2.1: Ekman’s extended emotions

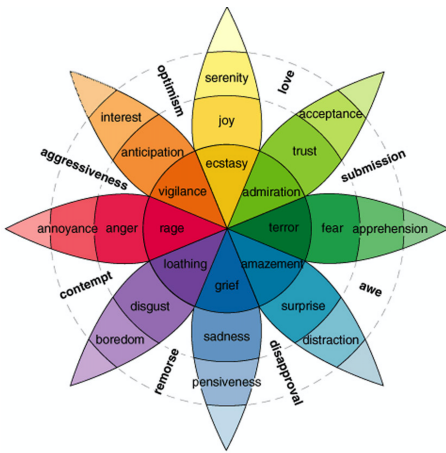


Figure 2.2: Wheel of Emotion adapted from Plutchik (2001)

**Plutchik’s Eight Basic Emotions** An interesting argument in recent years has focused on the appropriateness of Ekman’s six basic emotions for their applicability to online discourse, such as social media posts. Therefore several authors have taken to extending Ekman’s six basic emotions (Rothkrantz 2014) to include further emotions relevant to online discourse. This is especially relevant in the context of studies that have shown that restricting a

system’s classification label set to a small number of pre-defined emotions can lower accuracy in the face of ambiguities (Mohammad 2016), e.g., when the actual emotion is not part of the label set. Therefore researchers have taken to use Plutchik’s eight basic emotion theory (Plutchik 1984) or the ‘wheel of emotion’ (Plutchik 2001), where an illustration is shown in Figure 2.2. Plutchik (1984) extended Ekman et al. (1987) six basic emotions and added ten postulates to his existing theory (Plutchik 1990).

For this work, the focus remains on Ekman’s six basic emotions for mainly practical reasons. Firstly, collecting this kind of data from public sources comes with its own limitations and restrictions as outlined in section 3.1. Secondly, in order to adhere to current guidelines and ensure the completion of this project, it was decided to develop a proof of concept for the aforementioned hypothesis first (see section 1.1.1) and then consider later to scale this work up. Finally, at the time of starting the data collection process, no previous work was found that had undertaken the same or similar task of (i) collecting this kind of data and (ii) trying to establish a new task of classifying emotion keywords. Therefore, it seemed sensible to establish this task first with a more limited set of emotions.

## 2.2 Emotion and Sentiment Analysis

SA has often been defined as the “*computational study of opinions, sentiments and attitudes concerning different topics, as expressed in textual input*” (Ravi & Ravi 2015). As a field, SA often relies on other NLP subtasks to be able to carry out tasks such as Target-Aspect-Based-Sentiment Analysis (TABSA) or polarity detection (Cambria, Poria, Gelbukh & Thelwall 2017). Work by Cambria, Poria, Gelbukh & Thelwall (2017) described these different tasks into three distinct categories, called the syntactic, semantic and pragmatic layer (see Figure 2.3). It is argued that all of these layers play an essential role in achieving the overarching goal of developing human-like sentiment analysis performance (Cambria, Poria, Gelbukh & Thelwall 2017).

The syntactic layer is mainly concerned with normalisation and preprocessing tasks of text data, which helps to transform informal or colloquial language into plain language. In the semantic layer, the normalised text is analysed for different concepts or the subjectivity of the content. The final layer

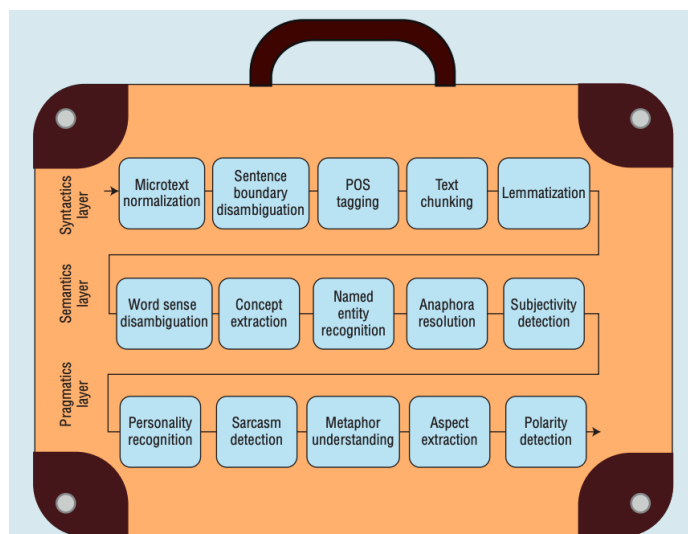


Figure 2.3: Sentiment Analysis Layers taken from Cambria, Poria, Gelbukh & Thelwall (2017)

“*aims to extract meaning from*

*both sentence structure and semantics*” (Cambria, Poria, Gelbukh & Thelwall 2017).

The tasks depicted in Figure 2.3 are all essential to the aforementioned goal of SA but does not give a holistic overview of SA. Therefore the following section will briefly outline several popular SA tasks, resources and approaches in order to gain

insight into how diverse and quickly evolving this field is.

**Data types and sources** There are multiple different data types that have been used in the field of SA at word-, sentence- and message-level or in micro-blogs, product reviews, blogs and whole documents (Mohammad 2016). Common sources of data from social media platforms, include Twitter (Go et al. 2009), Reddit (Rakib & Soon 2018, Demszky et al. 2020) or Tumbler (Kumar & Jaiswal 2017), which have often been used in SA of sentences, micro-blogs and blogs. Sources for SA of reviews include Amazon (Haque et al. 2018), IMDB (Lin et al. 2011) and Rotten Tomatoes (Socher, Perelygin, Wu, Chuang, Manning, Ng & Potts 2013). Sources that are not taken from social media platforms, but are used for SA are paper reviews (Appel et al. 2016) or clinical records (Deng et al. 2014).

**Affect categories** For lack of a better term that encompasses all possible types of categories that describe some kind of emotion, mood or polarity, the term ‘affect’ has been chosen. Polarities are often used in a variety of SA tasks, where the most basic categorisation is ‘positive’ and ‘negative’ and other methods also introduce the label ‘neutral’ (Bouazizi & Ohtsuki 2016). More fine-grained polarity categories use categories similar to a five-star rating system that have a sliding scale from ‘very positive’ to ‘very negative’ (Bhatt et al. 2015). Moods are often predetermined as categories based on the source or platform the data was collected from, where Mishne et al. (2006) used as a tool for blog mood analysis, on which users assign mood descriptors to their social media posts. Emotion categories are often associated to existing emotion theories that were developed in other sciences, such as neuroscience or biology, where popular emotion theories include Ekman (Ekman 1999), Plutchik and (Plutchik 2001).

**Tasks and application areas** Common tasks in the field of SA can be broadly summarised into polarity detection of various textual data types, stance and target detection as well as identifying subjectivity, valence and emotions in tweets (Mohammad 2016). Furthermore, there have been many different shared tasks, such as SemEval (Semantic Evaluation) (Wiki 2019) or dedicated workshops (e.g., WASSA (Workshop on Computational Approaches to Subjectivity and Sentiment Analysis) (Anthology 2020)), that have pushed the field further and introduced a number of new tasks. The first high-profile shared task specifically including SA, was held in 2007 (Strapparava & Mihalcea 2007), which looked at ‘Affective text’. Since then, WASSA workshops have proposed a variety of new subtasks since 2011 (Balahur et al. 2011), where Balahur et al. (2013) first included the sentiment analysis of social media data. Shared tasks in SA for Twitter have targeted many different sub-tasks that also include languages other than English and moved to 5-scale polarity detection of tweets (Rosenthal et al. 2017). Balahur et al. (2017) introduced the shared task on emotion intensity and Balahur et al. (2018) hosted the Implicit Emotion shared task (Klinger et al. 2018). Recent efforts have increasingly focused on tasks such as social network analysis, humour detection and applications to real-world tasks (Balahur et al. 2019). Real-world tasks span a range of different areas, such as finance, health and fitness (Cunha et al. 2016), moderating social media for abusive language (Kiritchenko & Nejadgholi 2020, Kiritchenko & Mohammad 2018), cyberbullying (Rakib & Soon 2018), politics (Ramteke et al. 2016) as well as mental health research (Coppersmith et al. 2014) and news (Godbole et al. 2007). With the introduction of these new tasks and application areas to tackle real-world problems, several new resources were also introduced. These often include task- or domain-specific dataset, for Twitter (Saif et al. 2013) or lexicons (Mohammad & Turney 2013, Baccianella et al. 2010) and other knowledge bases (Cambria et al. 2010). However, this advance of the field has also brought on its own challenges, which include but are not limited to, colloquial language and

non-standard language used on social media (Minaee et al. 2020), lack of large labelled datasets, and cross-cultural differences (Mohammad 2016). It is important to note that this summary is by no means comprehensive and does not include considerations for languages other than English.

The above section shows how diverse and multifaceted the field of SA and its application areas are. Furthermore, it outlines how fast-evolving SA is, where most recently a survey paper by Cambria, Das, Bandyopadhyay & Feraco (2017) summarises the field of SA.

### **2.2.1 Approaches to Sentiment Analysis**

Many different subject areas play an important role in conducting research in SA, which include but are not limited to neuroscience, psychology, computer science, linguistics and social media (Meiselman 2016). As outlined in section 2.2, the most common tasks of SA are emotion and polarity detection, where Cambria, Das, Bandyopadhyay & Feraco (2017) argue that the former can often be viewed as a subtask of the latter. Polarity detection is often performed to determine whether the content is positive, negative or neutral, whereas emotion detection is often tied to an emotion theory with a set number of emotion labels (Cambria, Das, Bandyopadhyay & Feraco 2017). With the rise of social media platforms, such as Twitter, there has been increased attention from the SA community to use this kind of data for their research (Rosenthal et al. 2017, Nakov et al. 2016). Furthermore, it has been argued that due to the nature of Twitter content, it should be seen as a sentence-level SA task (Kouloumpis et al. 2011) rather than a document-level task.

Overall, there are three main approaches to SA which have been called statistical, knowledge or lexicon-based and hybrid approaches (Cambria, Das, Bandyopadhyay & Feraco 2017). The following section will define what these three approaches entail



and give examples in the context of SA, with a specific focus on SA for Twitter data.

## Statistical approaches

Statistical approaches encompass both traditional machine learning techniques such as Support Vector Machines (Suykens & Vandewalle 1999) as well as deep learning models (Ravi & Ravi 2015). This approach has been on the rise over recent years due to the popularity and success of deep learning models that have achieved competitive results in several SA tasks (Rosenthal et al. 2017).

One of the most well-known examples of SA is the task of classifying movie reviews into positive and negative categories, where Pang et al. (2002) used three traditional machine learning algorithms (SVM, NB and Maximum Entropy) achieving an accuracy of 82.2%. Many other approaches have been applied to this task, where Kim (2014) proposed the use of a CNN, achieving a classification accuracy of 81.5%. Other work by Socher, Perelygin, Wu, Chuang, Manning, Ng & Potts (2013) also used a movie review dataset that was annotated for sentiments at a clause- and sentence-level. Since then, statistical SA methods have become increasingly sophisticated, utilising many different deep learning techniques. Research conducted by Yang et al. (2016) introduces attention to the task of document classification, testing the model on a movie review dataset to classify binary polarities and obtains state-of-the-art results. Another example of this is work by Li et al. (2018), who proposed a hierarchical attention network for cross-domain binary sentiment classification on Amazon reviews. Go et al. (2009) introduced one of the first datasets for sentiment analysis of tweets, where the data was collected based on emoticons and categorised as positive and negative accordingly. Work by Wang et al. (2015) introduces an Long-Short-Term-Memory (LSTM) neural network for binary polarity prediction in tweets, achieving an accuracy of 87.2%. Later on, research by Baziotis et al. (2017a) proposed the use of LSTMs with attention mechanisms for Twitter data during a shared task, achieving first place (in sub-task A) for classifying tweets into three polarity categories. Yin et al. (2020) modify an existing LM, called BERT for phrase-level sentiment classification and test it on the Twitter emotion intensity

classification task.

However, there are drawbacks to only using statistical approaches. Most methods only work if the following conditions are full-filled, where (i) a large amount of data is needed that (ii) is annotated with polarity or emotion labels (Glorot et al. 2011). Furthermore, it has previously been established that statistical approaches are semantically weak, which means that lexical elements have only small predictive power (Cambria, Das, Bandyopadhyay & Feraco 2017). Therefore, these methods do not perform as well on smaller ‘linguistic units’ such as sentences or clauses, because the semantic value cannot always be derived from the frequency of lexical items or their co-occurrence with each other (Cambria 2016).

Figure 2.4 shows how a typical tweet, where the annotated label is ‘negative’. This example highlights how a tweet is typically not as long as a document or paragraph, which means that statistical method would struggle to classify this tweet accurately.

**listening to everglow still makes me  
all soft and sad inside and makes  
me want to just curl up and cry for  
days 🥺😞😭**

Figure 2.4: Example of a ‘negative’ tweet in statistical SA

Furthermore, it is important to note that especially deep learning approaches rely heavily on word embeddings, which represent words based on their co-occurrence with each other in a vector (Socher et al. 2011). The use of word embeddings for SA is further outlined in section 2.4.3, where both the advantages and disadvantages will be discussed.

## Knowledge-based approaches

Research in this area has often relied on using Knowledge base (KB)s to classify or annotate textual data based on the presence of an explicit word such as ‘happy’, where keywords often carry a clear sentiment meaning (Cambria 2016). Several different studies have applied KBs to SA classification looking at a number of different aspects, such as polarity of tweets, opinion extraction, subjectivity or topic detection. These approaches are especially popular due to their efficiency and accessibility (Cambria, Das, Bandyopadhyay & Feraco 2017), where section 2.5 outlines several different resources that are used in SA research.

In the following section, a number of research areas will be introduced, where the work uses sentiment KBs.

Nithish et al. (2013) propose a method to extract opinions on products and their features from micro-blogging websites such as Twitter by using ontologies. There are three parts involved in building the model ‘(i) the creation of the domain ontology for the problem under consideration (ii) the extraction of relevant reviews on product features from Twitter (iii) the process of sentiment analysis.’ Step 1: vocabulary and objects for the domain are identified, and product specifications were found on online shopping websites for mobile phones. Step 2: Using the Twitter API to stream data based on keywords related to mobile phone features. Step 3: Every feature is assigned an opinion score, and the Stanford Natural Language Processor (SNLP) is used to detect dependencies between words in the sentences (i.e.:POS). To determine the score, a feature SentiWordNet is used, given a score of positive, negative or neutral. The more objective a word is, the less opinionated it is. Afterwards, the opinion scores are assigned to each feature for every product; they are fed into the mobile phone ontology, which produces a more detailed rating for each feature for every phone. However, one of the problems is that not enough feature data is available for every phone. Work by Kontopoulos et al. (2013) proposes the

development of a more fine-grained sentiment analysis system based on a domain ontology. The data collected for this experiment was streamed from Twitter (110 tweets) according to relevant keywords and then preprocessed. Afterwards, tweets were submitted to a web service which tags the data for sentiments based on the subject domain and assigns a sentiment score to each aspect of the tweet. The highest recall was achieved (min. 0.79 - 0.94) for the semantically-enabled structure that was proposed. Research conducted by Fan, Yan, Du, Gui, Bing, Yang, Xu & Mao (2019) utilises a sentiment lexicon in a task called Emotion-Cause Analysis (ECA), where the main aim is to identify the clause that caused emotion in a sentence. In their work, they use a regularised hierarchical RNN and an attention mechanism to take into account the context of a sentence and attend to the most important words. An important indicator of an emotion cause is usually a sentiment word, but there is no guarantee that the attention mechanism focuses on such words. Therefore, a sentiment regularised based on a lexicon is used to increase the margins of attention words if the word is part of the lexicon. Research conducted by Thakor & Sasi (2015) has presented a model that looks at retrieving tweets that contain negative feedback and comments on postal services in the UK, USA and Canada with the goal to extract the reasons for customers' dissatisfaction. As part of this study, they build their own ontology based on Twitter data and then used it to define the problem from the negative tweet. A total of 250 tweets from each postal service were used for analysis. However, this number dropped for all three categories after data cleaning. The highest accuracy achieved for identifying products/services was 66.66% for Canadian postal services.

Figure 2.5 shows an example of a tweet that would be analysed by using a keyword from a KB to classify the overall sentiment. In this example, the overall sentiment might be 'happy', but it could be argued that this approach misses a second approach such as the anger or disappointment expressed in the second half of the tweet (see: *'so go off'*).

it's our time !! <user> and i  
doing our happy dance right  
now 🥳❤️💋🌈 #voicebattles

Figure 2.5: Example of a tweet containing the keyword ‘happy’ in knowledge-based SA

However, relying on knowledge alone can have some disadvantages, which includes that much importance is given to the breadth and quality of the knowledge source (Cambria 2016). Furthermore, it found that often a KB does not contain the different facets of a concept, and therefore any representation of it is ultimately fixed. Cambria, Das, Bandyopadhyay & Feraco (2017) also argue that KBs often lack the ability to detect any kind of linguistic rule, where concepts such as ‘negations’ are often not correctly processed.

### Hybrid approaches

These approaches are utilising both knowledge-based and statistical methods for several different SA tasks such as polarity detection or subjectivity analysis (Ravi & Ravi 2015). Cambria (2016) has argued that the main goal of hybrid approaches is to ‘*better grasp the conceptual rules that govern sentiment*’, where there is a clear shift from pure syntax-based techniques to more incorporation of semantic-aware frameworks. During this literature review, it has been noted that the term ‘*hybrid approach*’ has not been clearly defined yet, which means that any combination of techniques used fall under this term. However, it is outside of the scope of this work to define hybrid approaches and their different variations and therefore for the purpose of this work the term ‘hybrid approaches’ will be used when referring to any kind of combination of statistical and knowledge-based approaches. Examples will be given to demonstrate the breadth of techniques used in hybrid approaches and their success on different SA tasks.

Gievska et al. (2014) developed a model to help people deal with negative emotions through a mobile application. This hybrid model was built on a range of lexical resources that contain affective words, and a Support Vector Machine (SVM) algorithm was used for the final classification. SVMs are supervised machine learning algorithms that can be used for both classification or regression tasks (Goodfellow et al. 2016). In this approach, the model classifies the text input based on a valence score that was assigned by a lexical resource. Valence has its origins in psychology and has been used instead of the term polarity within SA research (Meiselman 2016). The overall emotion classification of the input is built on Ekman’s six emotions (Ekman et al. 1987), which are chosen based on the highest valence scores. The model achieved an accuracy of 83.7%, which was significantly better than pure lexical (78% accuracy) or machine learning-based approaches (66% accuracy). It is important to note that the dataset for this work is based on several different public datasets, which have been annotated for the different emotion categories. Work by Akhtar et al. (2016) applied a hybrid model to a resource-poor language in order to perform aspect-based SA. Aspect-based Sentiment Analysis (ABSA) looks at content from social media in order to determine the polarity of every aspect in a text, where each aspect is a feature discussed in the text (i.e., ‘*shower gel*’, ‘*razor*’) (Pontiki et al. 2016). The data for this hybrid model was acquired from a shared task that previously released a Hindi Twitter dataset as well as product and movie reviews, which were annotated with polarities at the sentence level and aspect-level. The model uses a CNN in order to learn sentiment embedded vectors and replaces the softmax layer with an SVM for the final prediction. A Multi-objective optimisation (MOO) based Genetic Algorithm (GA) was used to compute optimised feature sets to form the sentiment optimised vectors, which were used as input into the SVM for classification. Overall classification accuracy for the Twitter dataset was lower (65.2%) than for the review dataset (65.96%), and accuracy was higher than on other comparable datasets. Research by Recupero et al. (2015) focuses on further

developing ‘*Sentilo*’ an unsupervised, domain-independent hybrid system which combines NLP and Semantic Web technologies. *Sentilo* classifies a number of topics and subtopics from an expressed opinion and then evaluates the sentiment. This model uses both a lexicon as well as an ontology, which is used to define concepts and relationships between opinion holders. All experiments were conducted using 100 TripAdvisor reviews for hotel stays, that were either 5 or 1-star rated reviews. Accuracies for opinion holder detection have been as high as 95%, whereas topic detection 68% and subtopic detection 78% were lower. Ma, Peng & Cambria (2018) propose Sentic LSTM, which is an aspect - based sentiment analysis and target-based sentiment analysis system that uses of commonsense knowledge. A standard LSTM is extended with target and sentence-level attention, where it assumes that polarity is associated with a specific aspect rather than a whole text unit. The proposed model constitutes of two parts: (1) Bidirectional LSTM and (2) sentence-level attention. Commonsense knowledge is induced through Senticnet (Cambria et al. 2016); this concept-level knowledge is then transformed into low-dimensional embeddings using AffectiveSpace (Cambria et al. 2015). The model was tested on the Sentihood (Saeidi et al. 2016) and Semeval-2105 (Pontiki et al. 2016) dataset, where it improves aspect categorisation on the first dataset (up to 20%), only achieves a slight improvement on the latter dataset. More recently, Li et al. (2020) integrate lexicons into sequence into two different CNN-LSTM models and tested it on two different SA tasks, where it was found that the proposed model outperforms baselines on both English movie and Chinese tourism reviews.

The example in Figure 2.6 shows one type of hybrid approach based on the work of Recupero et al. (2015) and it performs well on detecting different opinion holders, keywords and topics. However, the model was trained on a small dataset and heavily relied on the quality and breadth of both the ontology and lexicon. This could mean that this type of hybrid model will perform less well on larger datasets and miss bigger linguistic patterns. Also, it could be argued that this type of approach



would adapt less well to changes in the language used. An example of this could be applying this model to less formal and larger datasets collected from social media platforms.

i fear that one day , the person i loved  
the most is not gonna love me anymore  
❤️

Figure 2.6: Example of a tweet containing the keyword ‘fear’ in a hybrid approach SA

### 2.2.2 Fine-grained emotion classification

Over recent years the role of emotions in SA tasks have become increasingly important, where the focus has shifted from detecting emotions in fairy tales or stories (Alm et al. 2005) to analysing social media data (Cambria et al. 2019). Emotion detection has always been seen as one of the essential tasks of SA, but most current SA models are limited to detecting either polarity or limited sets of domain-specific emotions (Cambria 2016). Both Ekman’s six basic emotions (Ekman 1999) and Plutchik’s eight emotions (Plutchik 2001) have been popular in the SA community and have been applied to a range of different tasks. The following section gives an overview of the recent advances in the subfield of fine-grained emotion detection.

**Ekman’s basic emotions** Holzman & Pottenger (2003) have annotated chat messages from conversations online to determine if the content is emotional or not. Work by Aman & Szpakowicz (2007) proposed an emotion annotation scheme for blog posts, identifying the words and expressions that are most likely to be associated with emotions. Lu et al. (2011) use topic modelling and propose a visualisation of the affective structure in a text document, where each sentence was annotated with the six emotion categories. Research by Mohammad (2012) has looked into

the use of affect lexicons for classifying Ekman’s six emotions in sentences. Other researchers working in the field of SA have focused on more domain-specific emotion prediction, where Ekman’s six emotions were used in order to develop a mobile application helping people to deal with negative emotions (Gievska et al. 2014). Work by Schuff et al. (2017) manually annotated an existing Twitter shared task dataset with fine-grained emotions, where the emotions used were a combination of both Ekman and Plutchik. Similarly, Mohammad et al. (2018) introduced a shared task that looked at emotion classification using tweets that were manually annotated for eleven emotions. More recently, a shared task (Klinger et al. 2018) has taken place using Tweets that were collected based on Ekman’s six emotions, where the main task was to classify tweets accurately when ‘*emotion keywords*’ were removed. The winning model of this task was able to outperform this score significantly by achieving an accuracy of 71.45% (Rozenal & Fleischer 2018). The learning model used is a bidirectional Gated Recurrent Unit (GRU) with an additional attention mechanism inspired by Bahdanau et al. (2014) and a number of sub-Recurrent Neural Networks used at different layers. It has been reasoned that the model’s success has been due to a specific type of transfer learning. For this task, a baseline model was established, where the  $F_1$  score reached an accuracy of 59.1% on the test data.

**Plutchik’s eight basic emotions** There have been several approaches to this work including the use of gated recurrent neural networks (Abdul-Mageed & Ungar 2017), ontologies (Kontopoulos et al. 2013) and topic modelling techniques (Roberts et al. 2012). Research conducted by Mohammad (2012) used hashtags that contain emotion words based on eight basic emotions using SVMs for classification. Suttles & Ide (2013) use distant supervision to classify emotion pairs from Twitter using eight basic emotions in hashtags or emoticons. Research by Kim et al. (2012) looks at Twitter conversations to understand the emotions, where an emotion lexicon is built

based on the findings. Brooks et al. (2013) used the whole taxonomy of Plutchik's emotions to analyse chat messages for thirteen affect codes. Work conducted by Rothkrantz (2014) investigated eight basic emotions in online discourse.

This section has outlined the broader field of SA and described different approaches to polarity and emotion classification tasks. Then the literature in the subfield of SA, called fine-grained emotion detection, was reviewed.

## 2.3 Deep Learning

Deep learning is a subfield in the area of machine learning that uses multiple, stacked layers of artificial neurons for identifying features and learning representations from the input data to solve complex problems. These algorithms have been loosely inspired by the human brain's function and structure and have been able to process large amounts of raw data, which was previously not possible for traditional machine learning (LeCun et al. 2015). Learning representations from raw data for feature discovery is an essential step in deep learning approaches, which allows the neural network to compute models that transform the representation from one level to the next more abstract level until very complex functions can be learned (Najafabadi et al. 2015).

Arguably, the tipping point, which marked the rise of deep learning techniques was when Krizhevsky et al. (2012) introduced GPUs to train larger neural networks, which lowered the error rate significantly and allowed faster training of neural networks. Since then deep learning techniques have been applied to different problems within a wide range of subject areas such as speech recognition (Amodei et al. 2016), computer vision (Szegedy et al. 2016) or robotics (Levine et al. 2018).

It has been shown that deep learning techniques thrive on large amounts of data,

which are increasingly easier to collect due to the digitalisation of society (Goodfellow et al. 2016). This means that there are more datasets available for several different research areas and can be used for deep learning experiments, such as Natural Language Processing (Kingma & Ba 2014), Image Recognition (Krizhevsky et al. 2012) or Recommender System task (Harper & Konstan 2016). This, however, also means that not just the commercial value of data is increasing but also the value of deep learning technologies (Chen et al. 2014). A range of different deep learning techniques have been successfully applied to not just SA but also other NLP tasks (Young et al. 2018), including machine translation (Bahdanau et al. 2014) and text-to-speech systems (Arik et al. 2017).

A common approach to deep learning is supervised learning, which means that datasets for classification experiments are labelled (LeCun et al. 2015). In the case of SA tasks, supervised learning means that each sentence, word or aspect is given a label with a polarity, emotion or other defining categories. However, there are other types of learning approaches that have been successful for not only deep learning tasks but also in SA tasks.

**Supervised and Unsupervised learning** As previously mentioned, supervised learning is a common task not just in deep learning, but also traditional machine learning. In this approach, a given algorithm learns to map an input variable  $x$  to the corresponding output  $Y$ , where the algorithm's goal is to approximate the mapping function (Russell & Norvig 2002). In this process, the correct answer is known, and the algorithm makes predictions on the training data, where learning stops once a good level of performance is achieved. Supervised learning can be split into two subcategories, called 'Classification' and 'Regression', where the former's output value is a category, and the latter's is output a real value (Goodfellow et al. 2016). Unsupervised learning is in contrast to supervised learning, where the algorithm is given some input data with no corresponding output labels (Goodfellow et al. 2016).

The main aim of this approach is to detect underlying structures or distributions in order to learn more about the input data. Again, there are two broad subcategories of unsupervised learning, which are called ‘Clustering’ and ‘Association’ (Russell & Norvig 2002). Clustering algorithms take the input data and generate different groups that share common features so that for example, newspaper articles would be grouped into topics. In an association problem, the input is used to discover a set of rules for large chunks of the data.

**Semi-supervised learning** This approach falls between both supervised and unsupervised learning, where the input data is partially labelled. It has been argued that this is most closely related to real-world tasks, where in many cases not all data for a given task is already labelled (Goodfellow et al. 2016). Good quality annotations for large datasets are both expensive and time-consuming (Weiss et al. 2016) and therefore this approach has seen an increase in popularity over recent years. The main aim of this approach is to learn a data ‘representation so that examples from the same class have similar representations’ (Goodfellow et al. 2016).

**Transfer learning** Ruder (2019) gives a great overview of the field of Transfer Learning, which could be summarised as a set of methods that transfer knowledge from a pre-trained learning model into a different learning setup. Furthermore, it shares some assumptions that are closely related to the ideas of semi-supervised learning, where there is only partially labelled data available in real-world scenarios (Weiss et al. 2016). Therefore, some of the main benefits include a reduction in computation time and memory efficiency. In the context of language modelling, transfer learning has been hugely popular, because the resulting models overcome previously common issues such as lack of linguistic knowledge and context (Weiss et al. 2016). In this work, the focus will be on classification problems (see Chapter 3). However, some aspects of this PhD also touch upon the concepts of semi-supervised

and transfer learning (see Chapter 4).

**Multilayer Perceptron** A now classic and the most basic example of a neural network is the Multilayer Perceptron (MLP), which was originally invented by Rosenblatt (1957) and rose to popularity again after seminal work by Rumelhart et al. (1985), who introduced the idea of backpropagation. MLPs are a type of feedforward neural network that represents a nonlinear mapping between an input vector and an output vector, see Figure 2.7. This type of model consists of an input layer and one or more hidden layers, which are connected to each other through nodes that have weights and biases associated with it. The weights of a neural network are often initialised at random; however, they can be set to more specific values. During the *forward pass* information flows from the input layer, through the hidden layers, where an activation function is used to calculate the value of each hidden layer using the weighted inputs. There are different types of activation functions in use, some of the most popular ones being a sigmoid function (Han & Moraga 1995) or Rectified Linear unit (ReLU) (Nair & Hinton 2010). The information then flows to the output layer, where the output is measured against the ground truth label. At this point, the error is calculated and propagated back through the network, layer by layer, and the weights are updated according to the amount that they contributed to the error. This process is called *backpropagation* and one of the most popular methods of computing this *backward pass* is called stochastic gradient descent (Bottou 2010).

In order to move from simple MLP networks to RNNs, researchers have taken advantage of the idea that encourages sharing parameters across different parts of a model (Goodfellow et al. 2016).

The following section outlines many different types of neural networks that have been popularised both in NLP and used in SA as well as LMs.

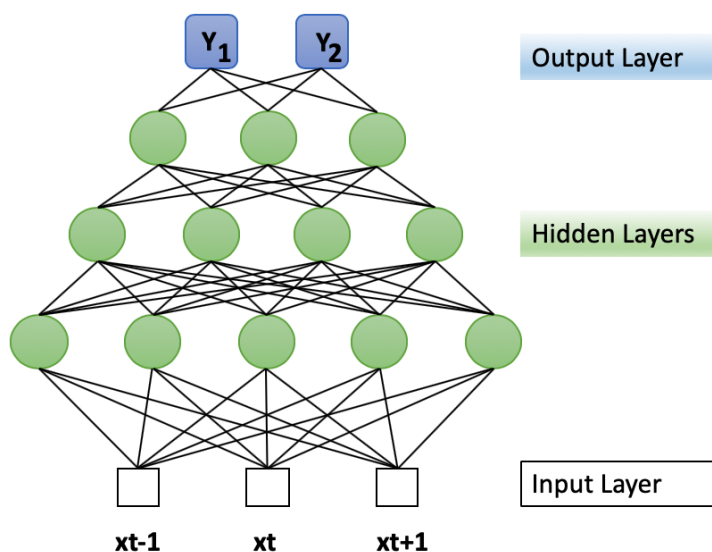


Figure 2.7: MLP (adapted from (Gardner & Dorling 1998))

**Convolutional Neural Networks** were first made popular in the field of computer vision for tasks such as image classification (Wei et al. 2015) and emotion recognition in videos (Fan et al. 2016). A traditional Convolutional Neural Network (CNN) usually consists of an input layer, one or more convolutional layers, a pooling layer, an activation function and an output layer (see 2.8). Within the convolutional layer, filters are utilised that scan across an input image, and then a value is calculated based on the filter using a convolution operation, where each 2-D slice of a filter is referred to as a Kernel, and they introduce parameter sharing across the network. Then a feature map is created for each filter and taken through an activation function that decides whether a certain feature is present in an image. A common technique in deep CNNs is to use a pooling layer to select the largest values in the feature map and input them into the next convolutional layer. Finally, there is a fully connected layer, which flattens the output before the image is classified with a softmax function (see Figure 2.8).

In NLP and SA, CNNs have been commonly used for tasks such as text classification (Kim 2014) and polarity detection (Shin et al. 2017, Duque et al. 2019).

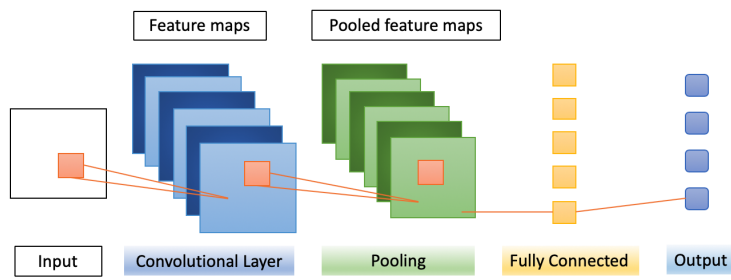


Figure 2.8: Simplified Convolutional Neural Network architecture

**Memory Networks** were first introduced by Weston et al. (2014), where the network architecture uses a specific memory component that can be read and written to with the aim to make a prediction. The network was first introduced to combat existing issues with Recurrent Neural Network architectures and was tested on the task of Question-Answering because there the memory can operate as a *dynamic knowledge base* and produce a textual output. The memory network contains 4 different components, abbreviated with the letters  $I, G, O$  and  $R$ . The architecture first takes as an input  $m$ , which is an indexed array, such as a vector or array of strings. Then the input feature map  $I$ , converts the incoming input to the internal feature representation. Afterwards, the generalisation component  $G$  updates the old memories with the newly converted input. The output feature map  $O$  generates a new representation and infers through calculation which memories are relevant for producing a good response. Finally, there is the response  $R$  component, where the final output is produced in the form of a textual response for example based on the results of the  $O$  component (see Figure 2.9).

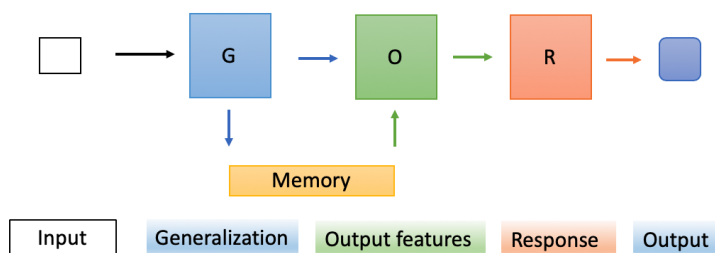


Figure 2.9: Simplified Memory Network architecture



Memory networks have been further developed to end-to-end memory networks (Sukhbaatar et al. 2015) or EntNet (Henaff et al. 2016). Different types of memory networks have been used in a number of different NLP tasks such as Question-Answering (Kumar et al. 2016), but also SA (Tang et al. 2016).

### 2.3.1 Graph Neural Networks

Graph Neural Networks (GNN) were first introduced by Scarselli et al. (2008) and have since then impacted on many different research disciplines such as social science (Kipf & Welling 2016), protein-protein- interaction networks (Fout et al. 2017) and knowledge graphs (Hamaguchi et al. 2017). GNNs are also classified as connectionist models and belong to the group of neural network approaches that operate on graphs (Zhou et al. 2018). Since the introduction of the original GNN by Scarselli et al. (2008), several new variations have been developed which include GNNs that use gating and attention mechanisms (Zhou et al. 2018). A GNN is a neural network architecture that is applied to learn on graphs, such as knowledge graphs with the goal to learn feature representations of each node in the graph. There are different types of GNNs, including Gated GNNs and Graph Attention Neural Networks that have become popular for a range of different tasks. Gated GNNs have looked at using GRU (Cho et al. 2014) or LSTM (Hochreiter & Schmidhuber 1997c) gating mechanisms to address limitations of GNNs to help improve learning long-term information across the graph. One of the first GNNs taking advantage of gating mechanisms was the Gated Graph RNN developed by Li et al. (2015). Another promising model developed introduced Tree-LSTMs (Tai et al. 2015), where graph-structured LSTMs (Zayats & Ostendorf 2018) are a variation of such a model. A variety of Graph LSTMs have been extended and applied to a number of different NLP tasks, such as relation extraction (Peng et al. 2017), text encoding (Zhang et al. 2018) or semantic object parsing (Liang et al. 2016).

The most popular variation of Graph Attention Neural Networks utilising attention mechanisms in GNNs is called "GAT" by Veličković et al. (2017).

One of the most popular learning models used in NLP and text classification (Yao et al. 2019a) is the model developed by Kipf & Welling (2016). The remainder of this section will give a brief overview of the workings of Graph Convolutional Neural Networks (GCN) based on this work. The input to a graph  $G$  is usually a feature matrix which is a  $N$  (nodes)  $\times$   $F$  (features per node) matrix and a  $N \times N$  matrix of the graph structure, which is called the adjacency matrix  $A$  of the graph  $G$ . Each hidden layer  $H$  of the network then uses a propagation function  $f$ . Thus each layer  $H$  corresponds to a feature matrix in which the rows represent the features of a node and where the output features are aggregated to the next layer. This leads to the increased abstraction of the feature representations in each hidden layer. The propagation function is a non-linear activation function, which is a Rectified Linear Unit (ReLU). The output vector of the network then represents the whole graph as the sum of all embedding representations learned in the hidden layers. Different GNNs have been used in NLP for text classification (Zhou et al.

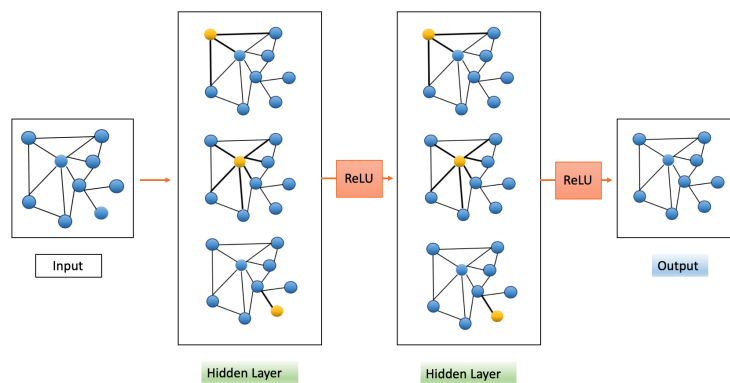


Figure 2.10: Simplified Graph Neural Network architecture (adapted from Kipf & Welling (2016))

2018), stance detection in news (Li & Goldwasser 2019) and in other SA tasks, such as ABSA (Zhang, Li & Song 2019, Wang et al. 2020).

### 2.3.2 Recurrent Neural Networks

Recurrent Neural Networks were first introduced in the late '80s, primarily for modelling time series (Elman 1990, Rumelhart et al. 1985, Werbos 1988). The principle idea of RNNs was similar to MLPs, where both networks share features such as neurons and multiple layers. However, the main distinguishing factor was and still is connections among hidden units with an added time delay, that allow the network to 'remember' information of the past. This allows the network to discover temporal correlations in a sequence that might be far away from each other.

There are a number of different types of RNNs, which include GRUs (Cho et al. 2014), LSTMs (Hochreiter & Schmidhuber 1997c) or bidirectional RNNs (Schuster & Paliwal 1997). Traditionally, RNNs lend themselves very well to NLP tasks such as machine translation or speech recognition (Young et al. 2017), because of their ability to process sequential data. 'Recurrent' refers to the RNN's ability to perform the same computation for each element of a sequence, where the output is dependend on the previous computations of that sequence. These recurrent connections add 'memory' to the neural network and can therefore learn larger abstractions from the input. Another advantage of an RNN compared to an MLP is that in theory, they can use arbitrarily long sequences as input, allowing them to capture context and meaning ranging from small sequences (e.g., one sentence) to very long sequences (e.g., long documents).

**Vanilla RNNs** Goodfellow et al. (2016) give an overview of how a general or vanilla RNN works that accommodates a whole input ( $\mathbf{x}$ ) sequence across time. A vanilla RNN can be thought of as a neural network containing many loops which pass on the output ( $\mathbf{o}$ ) of each timestep ( $\mathbf{t}$ ) to the next until an output is given ( $\hat{\mathbf{y}}$ ). Each hidden state ( $\mathbf{h}$ ) at the same timestep works as a '*memory*' unit where information from previous timesteps is gathered (Goodfellow et al. 2016). Similarly

to an MLP, RNNs also use activation functions (e.g., hyperbolic tangent function ( $\tanh$ )) in their hidden state and back-propagation is used to increase the accuracy of its predictions.  $(\mathbf{U}), (\mathbf{V})$  and  $(\mathbf{W})$  represent respectively the input-to-hidden, hidden-to-hidden and hidden-to-output weights, whereas  $(\mathbf{b})$  and  $(\mathbf{c})$  represent the bias vectors. Forward-propagation for each timestep in a Vanilla RNN (see Figure 2.11) would start with the initial state  $(\mathbf{h}_0)$  and then continue from  $\mathbf{t}=1$  to  $\mathbf{t} = \tau$ :

$$\mathbf{a}_t = \mathbf{b} + \mathbf{W}\mathbf{h}_{t-1} + \mathbf{U}\mathbf{x}_t \quad (2.1)$$

$$\mathbf{h}_t = \tanh(\mathbf{a}_t) \quad (2.2)$$

$$\mathbf{o}_t = \mathbf{c} + \mathbf{V}\mathbf{h}_t \quad (2.3)$$

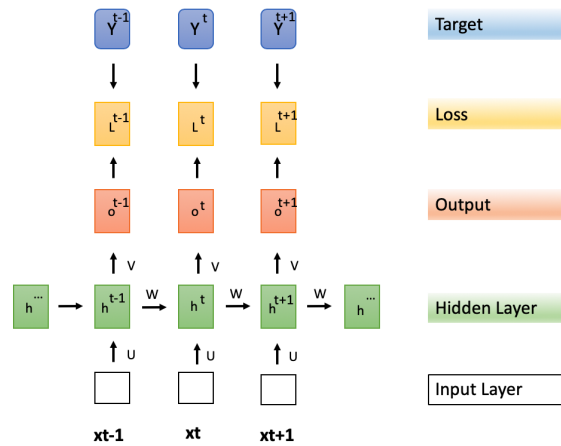


Figure 2.11: Unfolded RNN (adapted from (Goodfellow et al. 2016))

However, there is one main limitation to vanilla RNNs, because RNNs struggle to learn long-term dependencies due to the vanishing gradient descent problem (Hochreiter et al. 2001). In order to overcome this issue, gated RNNs have been developed such as LSTMs and GRUs and both have been used in many NLP application, such as grammar learning (Gers & Schmidhuber 2001), predicting clinical events (Choi et al. 2016) or music composition (Eck & Schmidhuber 2002).

**Long Short-Term Memory** These types of recurrent neural networks were developed by Hochreiter & Schmidhuber (1997c), which are capable of learning these long-term dependencies and overcome the vanishing gradient problem. Hochreiter & Schmidhuber (1997c) introduced an additional mechanism to the RNN that acts as a gate, which *'forgets'* certain information. The difference between a normal RNN and LSTM is that the hidden state in an LSTM has been given *'LSTM cells'*. LSTM cells also have an internal recurring self-loop with the most important feature called a state unit. This means that LSTMs have two recurring units, where one recurring unit is nested within the other (see Figure 2.12).

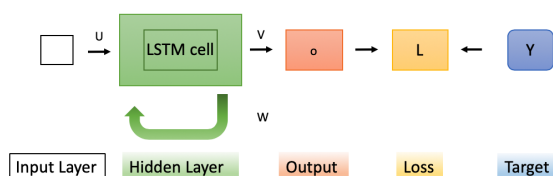


Figure 2.12: LSTM Cell (adapted from (Goodfellow et al. 2016))

However, LSTM cells still have the same input and output layer as normal RNNs, and three new gates (forget, input and output gate) have been added to control the flow of information (Goodfellow et al. 2016). One of the most important features is called the state unit, which can only be modified by the three LSTM gates and is continuously changing as the individual gates modify the information flowing through the cell (Goodfellow et al. 2016). The process within this LSTM cell is outlined as follows and can also be seen in Figure 2.13.

The *'forget'* gate  $f$  uses a sigmoid function  $\sigma$  which ultimately decides which information will be flowing through the other gates. It takes its input from  $\mathbf{h}_{t-1}$  and  $\mathbf{x}_t$  and its output adjusts the first state unit and therefore the cell state  $C_{t-1}$  (Goodfellow et al. 2016).

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (2.4)$$

The input gate layer consists of two different functions that are multiplied (Hochreiter & Schmidhuber 1997c).

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (2.5)$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \quad (2.6)$$

Afterwards the cell state  $C_{t-1}$  is updated to the new cell state  $C_t$ , where the sigmoid function and the output ( $i_t$ ) are multiplied by the output of a hyperbolic tangent function  $\tanh$ . This creates a new vector  $\tilde{C}_t$ . The output of those two functions ( $\tilde{C}_t$ ) will be used to update the cell state by adding the output of the input gate to the output of the forget gate, which creates a new cell state (Goodfellow et al. 2016).

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \quad (2.7)$$

The output gate also contains a sigmoid and  $\tanh$  function, which are multiplied (Hochreiter & Schmidhuber 1997c). Ultimately, the output gate filters what will be used as the output from the whole LSTM cell (Goodfellow et al. 2016). The cell state is run through another sigmoid function and then through another  $\tanh$  function independently. Then the output of the  $\tanh$  function is multiplied by the output of the sigmoid function and the result is the final output of the LSTM.

$$\sigma_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \quad (2.8)$$

$$h_t = \sigma_t * \tanh(C_t) \quad (2.9)$$

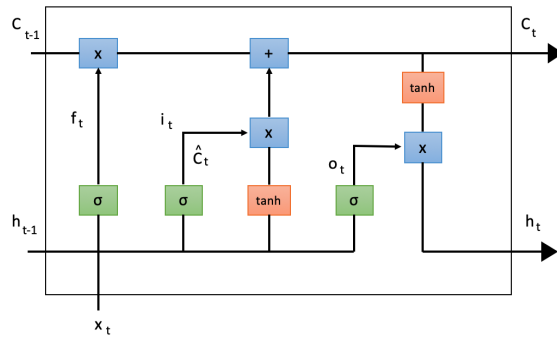


Figure 2.13: LSTM full process (adapted from (Goodfellow et al. 2016))

The increased usage of LSTMs in SA was first noted during the Sentiment Analysis Shared Task, where LSTMs were part of the winning model for all five subtasks in English (Rosenthal et al. 2017).

**Gated Recurrent Unit** After the popularization of LSTMs researchers have developed a range of different other methods that utilise different types of gates (Goodfellow et al. 2016). GRUs (Gated Recurrent Units) have a similar performance to LSTMs and work in a similar way to them but have no ‘*memory*’ unit (Young et al. 2017). The first successful Gated Recurrent Unit was developed by Cho et al. (2014), who argued that this new type of hidden unit will be easier to implement as well as compute. The model was created with two different gates, called ‘reset’  $r$  and ‘update’ gate  $z$ , where the update gate is a combination of the input and forget gate (Cho et al. 2014). Both gates are computed using a sigmoid function  $\sigma$  and can ‘choose’ to ignore or keep information (Goodfellow et al. 2016). The update gate has control over how much information is flowing into the next hidden state, which acts similarly to a memory cell found in LSTMs (Cho et al. 2014). Likewise, the reset gate can ‘choose’ to ignore information when the output is close to 0 and it will reset itself with the current input (Cho et al. 2014).

$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t]) \quad (2.10)$$

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t]) \quad (2.11)$$

Furthermore each hidden unit was given its own update and reset gate, which means that the model is able to capture dependencies over different time scales (Cho et al. 2014). This means that units learning short-term dependencies had more active resets gates, compared to units which stored long-term dependencies (Cho et al. 2014).

$$\tilde{h}_t = \tanh(W \cdot [r_t * h_{t-1}, x_t]) \quad (2.12)$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t \quad (2.13)$$

Due to these changes it has been argued by Chung et al. (2014) that GRUs are more efficient due to the lack of several control functions. Tang, Qin & Liu (2015) applied a similar GRU method successfully for document level SA, where the GRU was used to calculate the document representation of a movie review dataset.

**Bidirectional Recurrent Neural Network** Bidirectional Recurrent Neural Networks were first introduced by Schuster & Paliwal (1997) as an extension to existing RNNs and have been made popular in tasks such as handwriting recognition (Graves & Schmidhuber 2009). All previous RNN structures learned representations from a previous time step, the bidirectional RNN learns from both previous and future time steps which means that any output  $o^{(t)}$  is dependent on both the past and future (see Figure 2.14). It has been argued that this removes ambiguity in processing sequential data and enables the network to better understand context of the data (Schuster & Paliwal 1997). At each time step  $t$  the bidirectional RNN has two hidden layers at any given time to perform the left-to-right propagation and the right-to-left propagation. In order to compute a multi-layer bidirectional RNN,



with  $L$  layers, where the input to each intermediate neuron is at level  $i$  is the output of the RNN at layer  $i - 1$  at the same time step  $t$ .

The forward propagation in the hidden layer  $\vec{h}_t$  is computed as follows:

$$\vec{h}_{t^{(i)}} = f(\vec{W}^{(i)}h_t^{(i-1)} + \vec{V}^{(i)}\vec{h}_{t-1}^{(i)} + \vec{b}^{(i)}) \quad (2.14)$$

The backward propagation in the hidden layer  $\overleftarrow{h}_t$  is computed in the same manner, only with the difference of recursing in the opposite direction:

$$\overleftarrow{h}_{t^{(i)}} = f(\overleftarrow{W}^{(i)}h_t^{(i-1)} + \overleftarrow{V}^{(i)}\overleftarrow{h}_{t-1}^{(i)} + \overleftarrow{b}^{(i)}) \quad (2.15)$$

The final result  $\hat{y}_t$  of a classification task is computed by combining the scores of the hidden layers  $\vec{h}_t$  and  $\overleftarrow{h}_t$ . The output  $\hat{y}$  at each time step  $t$  is the result of propagating input parameters through all layers.

$$\hat{y}_t = g(Uh_t + c) = g(U[\vec{h}^{(L)}_t; \overleftarrow{h}^{(L)}_t] + c) \quad (2.16)$$

There is one limitation in this architecture, which arises when the context vector  $C$  output by the encoder RNN is too small to summarise a longer sequence (Goodfellow et al. 2016). Therefore, finding solutions to the problem of modelling long sequences has become a popular and active area of recurrent neural network research.

Overall, a variety of RNNs have been applied to SA tasks, such as sentiment classification in documents (Tang, Qin & Liu 2015) or opinion mining (Irsoy & Cardie 2014).

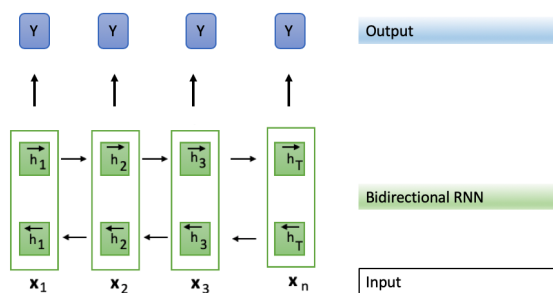


Figure 2.14: Adapted from Goodfellow et al. (2016). Bidirectional RNN with  $L$  hidden layers.

### 2.3.3 Attention Mechanisms

Attention was originally introduced by Bahdanau et al. (2014) for neural machine translation as part of an encoder-decoder RNN. Since then attention mechanisms have been developed further and used in a range of different tasks outside of NLP, such as computer vision (You et al. 2016) and recommender systems (Li, Ren, Chen, Ren, Lian & Ma 2017). Attention mechanisms have been broadly categorised into groups of self-attention or intra-attention mechanisms (Cheng et al. 2016, Vaswani et al. 2017), soft and hard attention (Xu et al. 2015) as well as global and local (Luong et al. 2015). These distinctions are not always clear and some attention mechanisms fall under more than one of those mentioned categories. Furthermore, many attention mechanisms have been used in combination with other neural network structures, including but not limited to RNNs or CNNs. However, recent advancements have led to new models being developed that solely rely on attention (Vaswani et al. 2017). Attention has also been proposed as a way of explaining the output of a neural network architecture in different English language tasks, however it has been noted that this heavily depends on one’s notion of explanation (Wiegrefe & Pinter 2019).

**Original Attention** The attention mechanism was first used for the task of neural machine translation by Bahdanau et al. (2014), where the goal was to

match an input of an arbitrarily long sequence  $x$  to an output  $y$ . In their work a RNN encoder-decoder architecture is used and extended by an additive *attention* mechanism to overcome the issue of only being able to learn fixed length vector representations. The bidirectional RNN (encoder) concatenates the forward ( $\vec{h}_i$ ) and backward ( $\overleftarrow{h}_i$ ) hidden state of an input sequence ( $x$ ). Therefore input sequence is annotated for both the previous and following word by the bidirectional RNN:

$$h_i = [\vec{h}_i^T; \overleftarrow{h}_i^T]^T, i = 1, \dots, n \quad (2.17)$$

The decoder outputs a word at position  $t$ ,  $t = 1, \dots, m$  and has a hidden state:

$$s_t = f(s_{t-1}, y_{t-1}, c_t) \quad (2.18)$$

The context vector  $c_t$  is the sum of the hidden states:

$$c_t = \sum_{i=1}^n \alpha_{t,i} h_i \quad (2.19)$$

The hidden states are then weighted with alignment scores, where a score  $\alpha_{t,i}$  is used to match  $(y_t, x_i)$  the input at position  $i$  and output  $t$ :

$$\alpha_{t,i} = \text{align}(y_t, x_i) = \frac{\exp(\text{score}(s_{t-1}, h_i))}{\sum_{i'=1}^n (\exp(\text{score}(s_{t-1}, h_{i'})))} \quad (2.20)$$

This score ( $\alpha_{t,i}$ ) is then used to assess how much information from the hidden input state should be used for each given output. Finally, the score  $\alpha$  is parameterised by a single-layer feed-forward neural network, where a hyperbolic tangent function ( $\tanh$ ) is used as the activation function.

The following section will give an overview of the different types of attention mechanisms in use and broadly outline how attention has been used in Sentiment Analysis.

**Soft and Hard Attention** Xu et al. (2015) propose two attention mechanisms, called soft and hard attention for describing the content of images within an encoder-decoder architecture. *Soft* attention is described as similar to work by Bahdanau et al. (2014) and where the attention mechanism's alignment weights are learned from all the patches of an image. *Hard* attention on the other hand focuses on selected patches of an image at a time.

**Global and Local Attention** The distinction between *global* and *local* attention was introduced by Luong et al. (2015) for the task of neural machine translation. Their description of these two types of attention mechanisms is similar to that of the previously mentioned *soft* and *hard* attention. The *global* attention mechanism has similarities to *soft* attention, whereas *local* attention is a mixture of both *hard* and *soft* attention.

**Self-Attention** Cheng et al. (2016) used self-attention also known as *intra-attention* when proposing a Long-Short-Term Memory network (LSTMN) for machine reading. In their work they firstly adapt the standard LSTM by removing the existing memory cell and adding a memory network (Weston et al. 2014). This allows the new structure to store each read token in a memory slot, where the memory continues to grow until its upper bound is reached. Furthermore they use attention to then induce relations between tokens and distinguish between shallow and deep attention fusion. For this they use an encoder-decoder architecture that utilises LSTMNs and two forms of attention called *inter-* and *intra/self-attention*. The main difference between the shallow and deep attention fusion is that for deep

attention fusion the inter-alignment vector is stored in the target memory network so it can review source information. The first model relying solely on self-attention was proposed by Vaswani et al. (2017) called a ‘*Transformer Network*’. This new network is also based on an encoder-decoder architecture and relies on multi-head attention only instead of LSTM cells. Both encoder and decoder of the network have a layer of *multi-head attention* followed by a feed forward layer. Furthermore, each layer within the encoder and decoder architecture is connected by residual connections and is followed by layer normalisation. However, the decoder structure has an additional layer prior to the multi-head attention layer called *masked multi-head attention* (see Figure 2.15b). The network’s depth is usually achieved by stacking multiple layers of equal numbers of encoders and decoders on top of each other, where input sequences are passed to the first encoder to produce embedding representations. Once all representations have been fed through all encoder layers, the last encoder feeds this representation to all decoders (see Figure 2.15a).

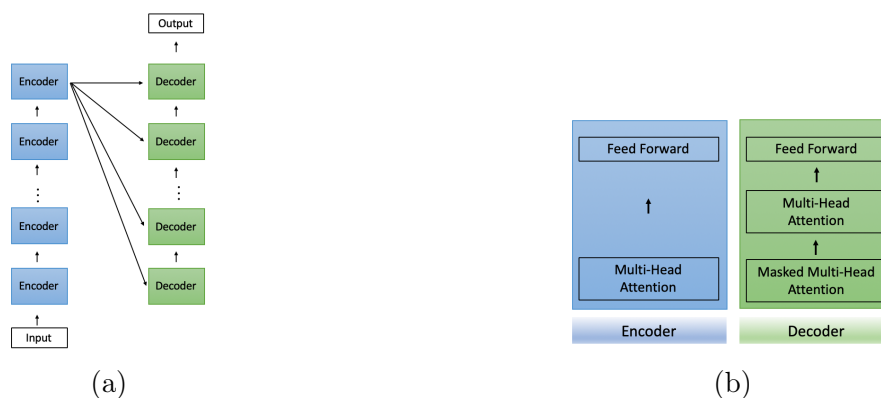


Figure 2.15: Simplified adaptation of the Transformer Network by Vaswani et al. (2017) showing (a) the overall Encoder-Decoder architecture and (b) the architecture within each encoder and decoder.

In order to calculate *multi-head attention*, one must first calculate the *scaled dot product self-attention* multiple times and in parallel. Each *scaled dot product self-attention* is produced by each encoder generating a *query*, *key* and *value* vector for each word in the input sequence. An example of input sequence ‘*Dogs love cats*’, would result in the following representations:  $q\_Dogs$ ,  $q\_love$ ,  $q\_cats$ ,  $k\_Dogs$ ,

$k\_love$ ,  $k\_cats$  and  $v\_Dogs$ ,  $v\_love$  and  $v\_cats$ . The first step then proceeds to calculate the dot product between the query vector  $q\_Dogs$  and all key vectors:

$$score\_1 = q\_Dogs \cdot k\_Dogs \quad (2.21)$$

$$score\_2 = q\_Dogs \cdot k\_love \quad (2.22)$$

$$score\_3 = q\_Dogs \cdot k\_cats \quad (2.23)$$

The next step involves dividing each score by 8 which is the square root of the dimension of the key vector and each result is normalised through a softmax function. Then each normalised score is then multiplied by the *value* vectors  $v\_Dogs$ ,  $v\_love$  and  $v\_cats$  and the sum of the three values produces the final output of the *self-attention layer*.

The *query*, *key* and *value* vectors in the encoder are generated from the input sequence, however in the decoder the *query* vector is produced by the target sequence whilst the other two vectors are still generated from the input sequence. The Transformer network has no mechanism to store time dependencies and only encodes this indirectly in embeddings (by taking the sum of word embeddings and position embeddings). Therefore in order to prevent the decoder from paying attention to subsequent positions the layer is '*masked*'. This also serves the purpose of ensuring that the predictions can only be made for known outputs.

Adding attention to other types of neural networks such as a LSTM has brought on state-of-the-art results in a variety of SA tasks (Zhou et al. 2016). More specifically, the model proposed by Yang et al. (2016) has been hugely successful in polarity classification using the original attention mechanism proposed by Bahdanau et al. (2014), where Figure 2.16 shows the outline of the model. Essentially this model

draws on the strength of the bidirectional LSTM to capture context and uses attention to identify the most important words in a sequence.

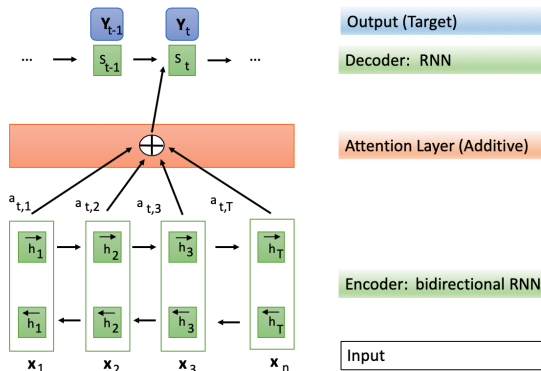


Figure 2.16: Bidirectional LSTM with attention mechanism (adapted from Yang et al. (2016))

### 2.3.4 Modelling Long Sequences

RNNs have been very powerful in a number of different sequence learning tasks (Cho et al. 2014, Zhao et al. 2017), but they still remain hard to train due to the problem of *vanishing* and *exploding gradient descent* (Bengio et al. 1994, Pascanu et al. 2012). This problem impacts on the RNN's ability to learn long sequences whilst maintaining mid- and short-term memory, when using gradient descent. Any instability of the gradient over multiple layers will impede the learning process due to the recurrent use of weight matrices that aggravate the instability. The decay (*vanishing gradient*) of the error signal over time leads to the error signal being lost. In the case of growth (*exploding gradient descent*) the opposite problem occurs where there are so many long-term error signals, that the short-term error signals are overpowered by them. Bengio et al. (1994) argues that RNNs must learn to retain information over a long period of time in the presence of noise.

Therefore there have been three main approaches to utilising RNNs for long sequence learning. Firstly, there are specialised neurons and cell structures in

RNNs (Koutnik et al. 2014) as well as hierarchical grouping of network modules. Popular approaches of specialised neurons and cell structures include the LSTM or GRU (see section 2.3.2). Hierarchical approaches can be broadly summarised into multiscale RNNs which group the hidden units of the network into multiple modules that operate on different times. Often specialised cell structures and hierarchical approaches are used in combination with each other. Overall these approaches resolve some of the common problems vanilla RNNs have, whilst also making them more computationally efficient. This is due to the network making less updates to higher layers, which also means that long term dependencies are updated less frequently at a higher level. This includes the ability of having a more flexible resource allocation (e.g.: more units can be used when learning long-term dependencies and less units can be used when learning short term information). Secondly, there have been RNNs using second-order optimisation techniques as well as random initialisation (Koutnik et al. 2014), these methods mainly utilise orthogonal matrices for initialisation. Finally, the attention mechanism (see section 2.3.3) was invented to solve the problem of learning long sequences for the problem of machine translation (Bahdanau et al. 2015).

### Specialised neuron and cell structures

Most notably work by Hochreiter & Schmidhuber (1997b) has proposed the LSTM cell for RNNs and other ‘gating’ structures have been introduced for tackling this problem such as the GRU (Cho et al. 2014). El Hihi & Bengio (1996) propose the first hierarchical recurrent neural network to model long-term dependencies, where different layers work on different time scales. Research conducted by Lin et al. (1996) introduces the ‘Nonlinear autoregressive exogenous’ (*NARX*) RNN using an orthogonal mechanism. A delay line is used for a single input and the output signal is fed back by a delay line to the input. Whilst the architecture



does not completely eradicate the problem it is shown that it can retain long-term information better than traditional RNNs. Work by Gers & Schmidhuber (2000) extends traditional LSTMs with weighted ‘*Peephole Connections*’ that allow each gate in the LSTM to look at the cell state. Therefore all gates in the LSTM cell add the ongoing cell-state to their input, where the previous cell state is added to the input gate and the current cell state is added to both the forget and output gate. The *Echo-State-Network* was introduced by Jaeger & Haas (2004) and consists of a RNN with sparsely connected hidden layers, where both the weights and connectivity of each neuron are fixed and assigned randomly. Work by Graves et al. (2006) proposes a task called *Connectionist Temporal Classification* (CTC), which uses a RNN structure to label unsegmented sequences. The key idea is to analyse the outputs of a network as probability distributions over each possible label sequence that is conditioned on a input sequence. The CTC model has a softmax output layer that has more than one unit than there are labels. This extra unit is used to observe when a sequence has no label. Unlike other sequence labelling algorithms CTC does not make assumptions about statistical properties of the data, patterns or relations. This work was extended by Fernández et al. (2007), who introduced a hierarchical approach of stacking multiple LSTM that have CTCs, where at each layer sequences of labels are predicted and fed forward to the subsequent layer. This approach is called *Hierarchical Connectionist Temporal Classification* (HCTC) and is used for sequence labelling in structured domains. The error rate is back-propagated using gradient descent through all lower levels, but can be adjusted depending on the degree of uncertainty of the target label. The level of uncertainty depends greatly on the variability of data. Sutskever & Hinton (2010) introduced *Temporal-Kernel Recurrent Neural Networks* (TKRNN) to overcome the problem of learning long-term dependencies by applying it to a serial recall task that uses an arbitrary sequence of characters with variable length. The TKRNN is similar to the NARX RNN in its nature, however the TKRNN has direct connections between units at each time

step.

Work conducted by Koutnik et al. (2014) introduces a modification to the vanilla RNN, called Clockwork-RNN (CRNN). The network's hidden layers are split into modules, where each module has its own *clock rates*. Each separate module is fully interconnected, however recurrent connections only exist between two modules, if the time period or *clock rate* 'a' of the first module is larger than the *clock rate* of the subsequent module 'b'. Sorting modules based on the clock rate enables the network to propagate the hidden state from right-to-left or in other slower to faster modules. The *clock rate* is chosen arbitrarily. Therefore the network is able to focus on both long-term dependencies through the slower rate modules processing context information as well as short-term dependencies processing high frequency information through the faster modules.

Chung et al. (2015) introduced a *Gated Feedback Recurrent Neural Network* (GF-RNN), which has a global gating unit that controls signals from upper RNN layers to flow to lower layers in stacked RNNs. It is a simplified version of the CW-RNN, where the connectivity pattern is not based on two consecutive time steps and no *clock rate* is introduced for each module. Similarly to the CW-RNN the hidden units in this network are separated into modules, where each module is linked to a different layer in the hierarchy and each module is connected to all other modules. Each recurrent connection between two modules is gated by a logistic unit, which is called the *global reset gate* and its values are derived based on the recurrent input and the previous hidden states.

Research conducted by Neil et al. (2016) introduces a recurrent neural network that overcomes the problem of discontinuous input sequences, such as temporal sequences collected from sensors. Using discontinuous input sequences subsequently leads to inputs with variable sequences lengths and therefore most RNN models cannot perform well on such tasks as they have fixed time steps. In order to circumvent this

problem a new gating mechanism - called *time gate* - is proposed as an addition to a standard LSTM. The *time gate*, which is controlled by a timestamp  $t$  is composed of an independent rhythmic oscillation that has three defining parameters. All parameters are learned during training and control when the gate is open or closed. The cell state and hidden output of the standard LSTM cell can only be updated during an *open* period of the *time gate*. Parameter one controls the real-time period of the oscillation, whilst parameter two controls the ratio of the ‘open’ time until the full period is reached and the third parameter controls the phase shift to each phased LSTM cell. The time gate goes through three different stages, where the gate is *open* in phase one and two, where it rises (0 to 1) in the first stage and falls (1 to 0) in the second stage. There is only error decay during stages of openness and during the last state the gate is closed and maintains the previous cell state.

Chang et al. (2017) introduced a dilated recurrent neural network to model long sequences, which comprises of dilated recurrent skip connections. Their experiments are carried out on three datasets, including MNIST and Penn-Treebank achieving state-of-the-art results on all benchmark datasets.

Dilations have first been introduced by Van Den Oord et al. (2016) in Convolutional Neural Networks for modelling audio waves. Since then dilations have been applied in a number of different settings, including reinforcement learning tasks where fixed dilations were used (Vezhnevets et al. 2017). Work by Chang et al. (2017) introduced a Dilated RNN by using dilations and skip connections to overcome the challenge of learning long sequences. So far these models have not been considered for sentiment analysis tasks yet, however the model proposed by Chang et al. (2017) was tested on a natural language processing task called tree parsing. The proposed model makes three advances that allow the learning of longer sequences. Firstly, dilated skip connections are introduced, which are computed where  $s^{(l)}$  is the skip length; or dilation of layer  $l$ ;  $x_t^{(l)}$  is the input to layer  $l$  at time  $t$ ; and  $f(\cdot)$  denotes any RNN

cell and output operations

$$c_t^{(l)} = f(x_t^{(l)}, c_{t-s^l}^{(l)}). \quad (2.24)$$

The main difference to a normal skip connection here is that the dependency  $c_{t-1}^{(l)}$  is removed from traditional skip-connections (outlined in equation 3.26).

$$c_t^{(l)} = f(x_t^l, c^{(l)t-1}, c_{t-s^{(l)}}^{(l)}). \quad (2.25)$$

One of the key benefits of this model is that it allows dilations to be increased exponentially, which was not possible previously.  $s^{(l)}$  denotes the dilation of the  $l$ -th layer

$$s^{(l)} = M^{(l-1)}, l = 1, \dots, L. \quad (2.26)$$

Therefore information of longer sequences can be learned at different layers, which circumvents loss of information over time and avoids the problem of exploding/vanishing gradient descent.

Chung et al. (2016) argue that it is essential for RNNs to ‘dynamically adapt its timescales’ to different lengths for each input sequence. If RNNs are given fixed boundary information it becomes easy to learn hierarchical representations in temporal data. However, this is not the case when no boundary information is provided. To overcome this problem a novel learning model called ‘Hierarchical multiscale RNN’ (HM-RNN) is proposed, which uses a parameterised boundary-detector at each layer. The HM-RNN employs a binary boundary detector that is considered to be ‘turned on’ once a segment has been completely processed, given that the value of the boundary state is 1. The boundary state is selected based on two conditions - the current time step in the layer below and the boundary state of the

previous time step in the same layer. The boundary state is dynamically determined based on three different operations: UPDATE, COPY and FLUSH. The ‘UPDATE’ operation is similar to the original LSTM update gate proposed by Hochreiter & Schmidhuber (1997c), but is executed sparsely. The ‘COPY’ operation retains the whole state without loss of information and occurs when the value is 0. Finally, the ‘FLUSH’ operation is utilised when a boundary is detected and a summary of the representation is ejected into the upper layer and the state is reset for a new incoming segment.

### Second-order optimisation and informed random initialisation

Another research direction has looked at the optimisation of the recurrence matrix in RNNs by proposing approaches such as orthogonal or unitary matrices for initialisation (Saxe et al. 2013, Arjovsky et al. 2016).

Martens & Sutskever (2011) used Hessian-Free optimisation with a ‘structural-damping’ function to solve the problem of learning long-term dependencies in RNNs. The model was tested on both synthetic and real-world datasets and it outperforms a standard LSTM. However, there are some disadvantages to this approach which include longer training times and the LSTM cannot be paralleled. Research conducted by Pascanu et al. (2013) introduces ‘*gradient clipping*’, which is used to solve the vanishing gradient descent problem, the original idea here was first proposed by Mikolov et al. (2011). The ‘clipping method’ introduced prunes the gradient based on the temporal component, which introduces an additional hyper-parameter called threshold. Experiments are run using the Penn Tree Bank and a music prediction task, where the method achieves state-of-the-art results on a language modelling task. Work by Le et al. (2015) introduces a RNN with rectified linear units (ReLUs), called the ‘IRNN’ and have used an ‘initialisation trick’ to train a RNN comprised of ReLUs

with the same success as LSTMs, where the recurrent weight matrix is set to be the identity matrix and all biases are set to 0. Therefore each new hidden state vector is produced by copying the previous state vector and adding on effect of the current inputs, where all negative states are replaced by 0. This means that the state of the RNN stays the same indefinitely, if there is no input given to the RNN. Thus, the IRNN mirrors the behaviour of the LSTMs gates, because the error is back-propagated through time and remains constant when assuming that no additional error-derivatives are added. The network is benchmarked against LSTMs, RNNs with tanh units and RNNs with ReLUs with random Gaussian initialisation. Experiments are conducted on the adding problem, language modelling, speech recognition and MNIST. Results indicate that the proposed method can compete with existing methods such as LSTMs, but is outperformed by a bidirectional LSTM in the speech recognition task. Work by Arjovsky et al. (2016) has proposed the use of *unitary weight matrices* to circumvent the problem of vanishing and exploding gradients when learning long sequences. Their network is tested on a set of pre-determined tasks, including the copying memory problem, adding problem and the MNIST classification task, where the proposed network outperforms LSTMs.

Overall, it can be seen that whilst various leaning models adapted for long sequences have been deployed for a variety of different task, including ‘sentiment analysis’, there has been little evidence of how these networks would perform on other real-world tasks, such as fine-grained emotion detection.

## 2.4 Language Models

The idea of using distributed word representations that are generated by neural networks and take advantage of the surrounding context was first popularised by Bengio et al. (2003) in the form of statistical language models (hereafter LMs). Other work has noted that encoding data into distributed word embeddings can improve performance in many different NLP tasks (Bordes et al. 2011) and this has been further advanced through the use of neural network based LMs that incorporate contextual information (Krasnowska-Kieraś & Wróblewska 2019). Over recent years, LM research has mostly relied on RNN structures, such as LSTMs, to obtain strong benchmark results. However, issues traditionally associated to RNN architectures, such as learning long-term dependencies, have still not been resolved and therefore impact on LM research (Dai et al. 2019).

The following section will give an overview of existing LMs that produce word embeddings for NLP tasks, such as *Word2Vec* and *BERT*. Word embeddings can be grouped into *static word embeddings* and *contextualised word embeddings* (Ethayarajh 2019a, Peters et al. 2019). There will be a further distinction between unidirectional LMs and bidirectional LMs in order to gain a better understanding of the neural network varieties used in current LM research. Furthermore, it will be outlined how these types of LM and embeddings representations may fall short when applied in the context of SA. Then an overview of existing embedding representation for SA purposes will be given.

### 2.4.1 Static Word Embeddings

The two most popular static word embeddings are called Word2Vec (Mikolov, Sutskever, Chen, Corrado & Dean 2013) and Global Vectors (GloVe) (Pennington et al. 2014). These models have been used in a number of different domains, such

as sentiment analysis (Xue et al. 2014) or document classification tasks (Lee & Dernoncourt 2016). Other research has also focused on understanding the theory behind these LMs (Allen et al. 2019).

**Word2Vec** was first introduced by Mikolov, Chen, Corrado & Dean (2013) as a novel neural network architecture to generate continuous word vector representations for large datasets. The key idea behind this work is that unlike previous LMs, Word2Vec accounts for similarity of words to each other so that words that appear in the same context share a similar embedding space. In their work they propose two different ways in which embedding representations can be learned: (1) Continuous Bag-of-Words model (CBOW) and (2) Continuous Skip-gram model (CSGM). The main difference between these two approaches is that CBOW predicts the target word corresponding to its context and that CSGM takes the target word to predict the context. While both approaches have their own benefits, it has been found that the CBOW method performs faster on larger datasets and CSGM works better for smaller data with rare words.

**Global Vectors** is another popular static word embedding method, which was introduced by Pennington et al. (2014). GloVe is based on an unsupervised learning algorithm - called a *log bilinear model* - where the main motivation is that the probability of word co-occurrences can encode meaning. Therefore GloVe first computes word co-occurrence matrices based on a given window size and then computes the co-occurrence ratios between words. Words that have similar meaning would co-occur more frequently with each other, e.g.: *apple* occurs more often with *fruit*, than it would with *cars*. Therefore GloVe relies on global count statistics and not just on local information. The main difference between GloVe and Word2Vec is that GloVe learns representations through co-occurrence matrices and Word2Vec learns representations through predicting context words or a target word.



## 2.4.2 Contextualised or Dynamic Word Embeddings

One of the main disadvantages of static word embeddings has been that they cannot take advantage of larger context when producing embedding representations as these are all based on co-occurrence of words with each other or predicting words from a very short context (skip gram). The following section will give an overview over both unidirectional and bidirectional language models (also known as *contextualised word embeddings*) that are able of incorporating context.

### Unidirectional LMs

Unidirectional neural LMs share the way in which they assign a probability over an input sequence and how it is estimated. The main differentiating factor of these LMs is in which way the output vector is computed. The following section will outline examples based on CNN, RNN and Transformer LMs that are used to generate word embeddings.

**RNN based** Research conducted by McCann et al. (2017) introduces Contextual Word Vectors (CoVe), which are pre-trained by an LSTM encoder on a number of machine translation datasets. The output of these representations is then appended to word embeddings that are commonly used in a variety of NLP tasks, such as GloVe. Apart from issues that arise through the usage of LSTMs, this type of representation is dependent on the amount of available datasets during pre-training. Work by Khandelwal et al. (2018) has investigated to which extent context is utilised in a standard LSTM LM. For this an ablation study is conducted and the main results are as follows: (i) the model is able to use around 200 tokens of context, (ii) it distinguishes nearby context from information further in the past through the 50 most recent tokens and (iii) the model is sensitive towards word order when processing the 50 most recent tokens, but is less sensitive to tokens beyond that.

**CNN based** Work by Dauphin et al. (2017) introduced the first LM approach for large scale language tasks that was not based on a recurrent architecture and achieved competitive results. For this they introduced stacked gated convolutional networks that have the ability to extract features over larger context, which enables them to capture long-term dependencies in a similar manner to RNNs. Another concept borrowed and adapted from RNN structures is the gating mechanism used, where it used to combat the issue for vanishing gradient descent in the network (named Gated Linear Units). Simply put, this LM has the following architecture: firstly an input sequence is fed to a look-up table. Then the output is fed into a convolutional layer and afterwards to the gating mechanism. The final step uses a variation of a softmax, called *adaptive* softmax to produce the final output. One of the benefits of this approach is that it is able to process inputs in parallel and is not restricted by the sequential nature of traditional RNN architectures.

**Self-attention based** Transformer XL networks were first introduced by Dai et al. (2019) to overcome the issue of Transformers only being able to learn fixed-length representations. Borrowing the idea of recurrence from RNNs, the hidden states of each layer are reused as the input of the next segment. Due to these recurrent connections the network is able to take advantage of long-term information that can be accessed at any layer within the network. Furthermore this also helps to combat the issue of context fragmentation due to the information passing through the whole network. Finally, attention lengths are generalised to be longer than the ones observed during training.

## **Bidirectional LMs**

Bidirectional LM approaches have gained increasing popularity due to their ability to produce state-of-the-art results in a number of downstream NLP tasks, such as

text classification (Sun, Qiu, Xu & Huang 2019) or Sentiment Analysis (Sun, Huang & Qiu 2019). This is due to their ability to learn word representations that are contextualised.

Research conducted by Peters et al. (2017) reviews the first existing bidirectional LM models that have been used in handwriting recognition and machine translation tasks. In this work they use stacked bidirectional RNNs to learn representations by concatenating the hidden states from the forward and backward pass in a sequence tagging task. This enables the network to encapsulate both past and future information into the context sensitive representation. The proposed model is called ‘TagLM’ and was evaluated on a NER task and chunking task, outperforming previously proposed methods.

**ELMO** was motivated by the idea that a word can have multiple meanings in different contexts, e.g.: the word ‘*play*’ can have different meanings depending on the context it is used in. Previously proposed word representations however would assign the same representation of ‘*play*’, regardless of the context the word is used in. To rectify this problem Embeddings for Language Models (ELMO) (see Figure 2.17) was introduced, which stands for *Embeddings for Language Models*. ELMO is a deep bidirectional LM that is based on multiple layers of LSTMs, where the architecture’s lower level LSTM states are capable of capturing syntactic information, whilst higher level LSTMs capture the context of a word (Peters et al. 2018). More specifically, ELMO first uses a character-level CNN to represent raw word vectors, which are then used as input representations into the first BiLM. The forward and backward pass of this first BiLM then forms the intermediate word representation that is then fed into the next BiLM. The final output of this process is the weighted sum over the initial word vectors and their intermediate representations, resulting in deep contextualised word embeddings. Unlike previous LMs, ELMO looks at the entire input sequence before computing an embedding representation for each word, which

means that the same word can have a different representation in a different context.

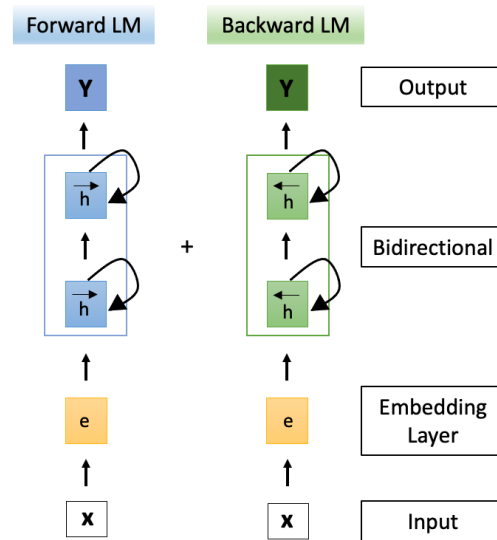


Figure 2.17: Architecture of the Bidirectional Language Model ‘ELMO’.

**GPT2** was introduced by Radford et al. (2019) and is the successor to the previously introduced GPT-1, which stands for ‘Generative Pre-training’. The first iteration of Generative Pre-training (GPT) proposed a generative learning model using the decoder of a transformer (12-layers) and masked self-attention. This LM was trained on the BookCorpus dataset using a combination of the unsupervised and supervised training approach, which resulted in an LM with 117M parameters. Later, GPT-2 was proposed, which is based on the Transformer network, where some extensions were made to it to also produce contextual word embeddings. An additional

normalisation layer was added after the last *self-attention* mechanism and the layer normalisation that takes place after calculating *self-attention*. The Transformer network has now been moved to the input of each new encoder. GPT-2 was trained on over 8 million documents, which is a huge increase over the previous version and resulted in a LM with 1.5B parameters. Since then GPT-3 has been proposed

(Brown et al. 2020), which has surpassed GPT-2’s size with 175 billion parameters and used dense and locally banded sparse attention patterns in turns.

**BERT** Bidirectional Encoder Representation from Transformers (BERT) was introduced by Devlin et al. (2018) as a new LM that was trained on over 3 billion tokens (see Figure 2.18). In their work two different types of BERT were introduced, *BERT Base*, which is 12-layers deep and *BERT large*, which is 24-layers deep. One of the key features of BERT is that it randomly masks words of a sentence and then predicts them, which is unlike previous LMs that predict the next word in a sequence. Whilst BERT is based on the popular Transformer architecture, it uses *self-attention* bidirectionally. Since BERT has been build it has drawn much attention from the research community, inspiring work that applies BERT to be adapted for different domains (Yin et al. 2020, Yao et al. 2019b), investigate how it can be made more efficient (Lan et al. 2019) and analysed what BERT focuses on (Clark et al. 2019).

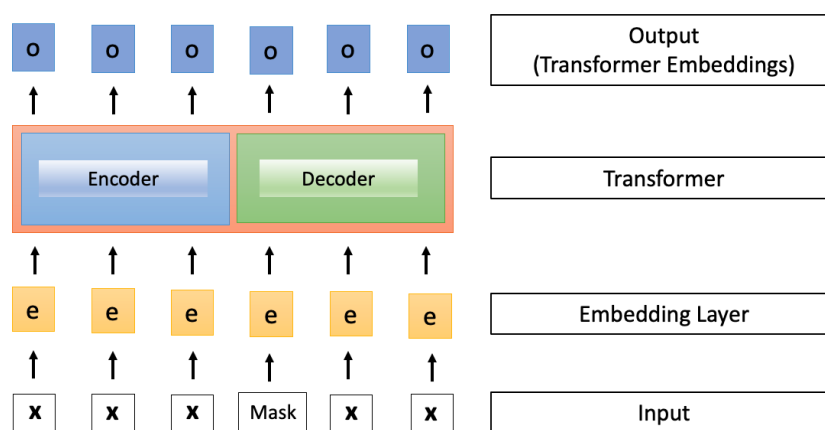


Figure 2.18: Simplified architecture of the ‘BERT’ Language Model adapted from Devlin et al. (2018)

Work by Ethayarajh (2019b) looks at using orthogonal and linear transformations to learn relationships between words and outlines how *attention-based* LMs are more sensitive towards syntax or positional information. The use of *attention* in LMs has also sparked the question to which extended these new LMs can be used to learn

from KBs (Petroni et al. 2019), where researchers have looked into which extend existing LMs, such as BERT or ELMO already store commonsense knowledge and have found that especially BERT seems to outperform on the tasks they introduced. Chen, Zha, Liu, Chen, Yan & Su (2019) aims to extract general-purpose relation embeddings in textual data, where they use global co-occurrence statistics between text and KB relations as their supervision signal. For this they train a Transformer network and project the resulting output to a new vector, which is the final embedding representation.

Most of the previously mentioned static word embeddings and LMs have been applied to SA tasks, however, this has often been limited to polarity detection only. Furthermore, most work in the space of LM models has focused on finding the semantic and syntactic similarities of words (Bengio et al. 2003), which is natural given the tasks they were originally used for such as coreference resolution (Dhingra et al. 2018) or analogy prediction (Reed et al. 2015). However, less attention has been paid to incorporating emotional context or meaning. Work by Ren et al. (2016) has argued that one of the limitations of current approaches is that one embedding is generated for each word and it does not take into account that a sentiment-bearing word could be polysemous. Whilst approaches like ELMO were specifically developed to take into account polysemous words and currently are more successful in SA tasks, they still do not incorporate emotional context or meaning explicitly. Research conducted by Ethayara, jh (2019a) has found that the same word in contextual word representations can occur in different contexts but have identical vector representations and the context-specificity is different in different models. Arguably this also impacts on the performance in SA tasks, because an emotion word such as *'surprise'* can have different meanings in different context. Recent research has found that pre-trained LMs *'teach'* neural networks about the structure of language (Clark et al. 2019) in subsequent tasks such as machine translation or question answering. Therefore it could be argued that utilising pretrained LMs

that incorporate sentiment could also yield better performance in SA specific tasks. Furthermore, it has been argued that whilst neural network based LMs do not require feature engineering, we cannot conclusively know what kind of features are being trained on (Krasnowska-Kieraś & Wróblewska 2019).

In order to compensate for the lack of emotional meaning and context in existing embedding representations and LMs, some research in SA has focused on developing sentiment embeddings.

### 2.4.3 Sentiment Embeddings

Including additional information such as sentiment into embedding representations has been an active research area in SA, because traditionally word embeddings and LMs have only focused on capturing syntactic and semantic information (Yu et al. 2017, Tian et al. 2020) and not considered the impact of missing sentiment information. This means that words carrying opposing sentiment or emotion can have similar vector representations, which could impact on worse SA performance on a range of different tasks (Tang, Wei, Qin, Yang, Liu & Zhou 2015). Examples of this could be words such as ‘good’ and ‘bad’ or more subtle and fine-grained ‘sad’ and ‘upset’. Furthermore it has been argued that words carrying sentiment can also have different polarities or emotions, when used in different topics or context (Ren et al. 2016).

Similarly, Zhang, Wu & Dou (2019) noted that oftentimes words are put in the same category based on common statistics in many LMs. However, these words may have different polarities and a common approach to overcome this issue relies on using fixed embeddings or fine-tuning existing LMs or embedding representations.

The following section outlines different approaches for generating word embeddings that integrate some form of sentiment information into their representation. The

main focus here is on statistical approaches, where embedding representations learned by machine and deep learning algorithms are considered.

Early work by Maas et al. (2011) uses a probabilistic topic model that derives polarities based on the embeddings of each word. The model is then compared to other existing topic models, including Latent Dirichlet Allocation (LDA) and Latent Semantic Analysis (LSA). Work by Labutov & Lipson (2013) uses logistic regression to re-embed existing embeddings. Research conducted by Liang et al. (2019) proposes the use of refined word embeddings for a task within the broader field of SA, called TABSA which aims to identify aspects in relation to their targets and infer a sentiment from target-aspect pairs. More specifically, this is motivated by the fact that most previous work in ABSA and TABSA utilised random vector representations for both targets and aspects, which means that both semantic information and independence between targets, aspects and context is lost. This is done by feeding a randomly initialised embedding representation into the proposed model, called the *RE+SenticLSTM*. The embedding representations are initialised using GloVe and then used as input into the model to learn refined representations of targets and aspects. This approach outperforms other approaches, however, it has only been tested on one specific task using datasets that depend on existing annotations for targets and polarities.

Yu et al. (2017) proposed a vector refinement model that can be applied to pretrained word embeddings (e.g.: Word2Vec and GloVe), where existing word embeddings are adjusted so that semantically and sentiment similar words are closer to each other and vice versa. This is done by utilising a sentiment lexicon that contains real-valued sentiment scores. The first step then calculates the semantic similarity between the affective word of the lexicon and words using the cosine similarity of their respective pretrained vectors. Next, the top-k most similar words are considered as nearest neighbours and then re-ranked according to the sentiment scores provided by the



lexicon. This results in words that are closer in sentiment to be ranked higher. The model is then benchmarked against a range of different neural network approaches and evaluated on binary and fine-grained classification on the Stanford Sentiment Tree Bank (SST). Unlike previous models this work does not require an already labelled dataset, but is trained on a limited lexicon of around 13k individual words. Furthermore, it has to be considered if such an approach would work on other types of datasets or tasks. More specifically, a huge amount of data is drawn from social media platforms for SA tasks, which often contains colloquial language and other non-standard words.

Especially in binary or polarity classification tasks for SA there has been an improvement upon using sentiment-aware word embeddings, where a polarity label is often used to optimise word vectors.

Research conducted by Lan et al. (2016) introduces three CNNs that learn Sentiment Word Vectors (SWV) through an additional sentiment channel. The Mixed-Sentiment Word Vectors (SWV-M) mix both semantic and sentiment information into the same dimensional vectors, which also means that both semantic and sentiment error are back-propagated and updated simultaneously. The Combined Sentiment Word Vector (SWV-C) trains the semantic and sentiment vector separately and both vectors are then combined. The final model called Hybrid-Sentiment Word Vector (SWV-H) is a combination of the SWV-M and SWV-C model. Here the SWV-H model also learns two separate vectors for semantic and sentiment vectors, where there is a sentiment-semantic (based on SWV-M) and sentiment-alone representation (SWV-C). The models are evaluated on two tasks, where the first task is to predict the sentiment polarity of words in a lexicon using an SVM. The second task is a tweet classification task based on three polarities (positive, neutral and negative).

Tang et al. (2014) introduced Sentiment Specific Word Embeddings (SSWE), where

the nearest neighbours in an embedding representation are not only semantically close, but also close in sentiment. For this task sentiment embeddings are learned from tweets using emoticons as labels for positive and negative polarities. For this they use a C&W model, which is a feed-forward neural network that generates word embeddings based on n-grams. This model is then refined through polarities by modifying the loss function, which is a weighted linear combination of the context loss and sentiment loss. The embeddings are evaluated on three different SA tasks, word-level SA for sentiment lexicons and sentence level SA classifying tweets and reviews.

Research conducted by Ren et al. (2016) proposes a method to enrich embeddings with topic and sentiment information in order to overcome the issue of traditional word embeddings not taking into account sentiment-bearing words and the context or topic they are used in. Therefore two learning models are introduced to generate Topic Sentiment Word Embeddings (TSWE) and Topic-Enriched Word Embeddings (TEWE). Both models are using a n-gram based neural network (C&W model) that is capable of learning local context and semantic relations. The TEWE learns the topic distributions of the input text based on n-grams, where the topic distribution is learned using Latent Dirichlet Allocation (LDA). The final softmax then outputs both the topic distribution and n-grams. The TSWE model on the other hand has two softmax output layers, which output the sentiment and topic distributions respectively. Then for each model Multi-Prototype Word Embeddings are produced, which calculates an additional ‘environment’ vector for high-frequency words with similar meanings. These new vectors are then clustered using k-means clustering using the cosine distance. This leads to a new embedding dictionary that contains cluster centres of the same words taking into account topics and their different sentiments. Finally, the embeddings are tested in a twitter classification task using a CNN to predict tweet polarities.

Research conducted by Li, Lu, Long & Gui (2017) argues that word embeddings do not only carry semantic, but also sentiment meaning that can be obtained through training. In their work a framework is proposed based on affective lexicons and seed words, which achieves competitive results in various downstream SA tasks. Dragoni & Petrucci (2017) generate skip-gram based word embeddings from a corpus of opinionated documents and apply the task to polarity prediction for multi-domain SA. Work by Picasso et al. (2019) used a finance-specific dictionary and Affective Space (Cambria 2016) to extract sentiment embeddings from news. These were then incorporated in both machine learning (e.g.: SVM) and a feed forward neural network for market trend predictions. The resulting learning model was then able to accurately classify both positive and negative market trends. More recent research conducted by Xu et al. (2020) introduces LMs that are specifically developed for the end-task (in this instance aspect-based SA) in mind, where an extension of BERT is proposed. The model is evaluated on the ABSA task classifying aspects of laptops and restaurants into three polarity categories. At the same time Yin et al. (2020) proposed a modification of BERT for phrase-level sentiment classification and test it on the emotion intensity classification task using Twitter data. The model ‘SentiBERT’ is trained on the SST (Stanford Sentiment Tree) and achieves competitive results, whilst also demonstrating the transferability of the model. Tian et al. (2020) propose ‘SKEP’ (Sentiment Knowledge Enhanced Pre-training), which embeds sentiment words, polarity and aspect-sentiment pairs into the representations during training. This is done by sentiment masking, which is not random but through the use of previously learned sentiment knowledge. A Transformer architecture is used in this work and the model is tested on three tasks, sentence polarity classification, aspect-level sentiment classification and opinion role labelling. Results indicate that on these tasks SKEP outperforms other transformer networks.

Although these aforementioned sentiment embedding representations solve the

problem of integrating sentiment information into word representations, there are still a number of limitations. Firstly, previous work has mostly focused on polarities where commonly positive and negative labels are used to indicate the sentiment. At the point of writing this thesis, there was no work found that incorporates fine-grained emotions using neural network based approaches. Secondly, more recent approaches have focused on generating sentiment embeddings for specific tasks only, such as TABSA or ABSA. Traditionally these tasks use a large amount of labelled data - sometimes labelled by humans- which arguably makes the resulting embeddings less scaleable or transferable to other tasks. Finally, sentiment refinement has been limited to static word embeddings, which leaves room for further experiments using more advanced methods such as BERT or ELMO.

## 2.5 Knowledge Representation

Machine and Deep Learning algorithms have been at the forefront of recent advancements in various NLP tasks, including SA. However, there are three unresolved issues concerning Machine and Deep Learning algorithms, which include the need for large amounts of data for training, variety in training approaches that lead to varying results (reproducibility) and no transparency over an algorithm's decision making process (explainability) (Cambria et al. 2018). These key issues have greatly impacted on the advances of NLP to achieve human-like performance and it has therefore been argued that there is an increased need to merge and utilise a number of disciplines (e.g.: linguistics, affective sciences, computer science and/or common sense reasoning) to achieve human-like performance in NLP (Cambria et al. 2018). In order to incorporate knowledge or common sense knowledge into machine and deep learning methods, researchers have focused on building a number of different resources. Within the field of NLP there have been three dominant ways to store knowledge in knowledge bases, namely lexicons, ontologies and Knowledge Graph (KG). Over recent years particularly lexicons and KGs have been successfully created and applied to a number of different tasks, including but not limited to Topic Modelling (Schoene & de Mel 2019) and Sentiment Analysis (Kiritchenko et al. 2014). Most notably the creation of lexicons such as WordNet (Miller 1995) has influenced NLP tasks such as dependency parsing (Herrera et al. 2005), measuring word similarities (Pedersen et al. 2004) and document clustering (Sedding & Kazakov 2004). This has also led to the creation of lexicons specific for SA, such as WordNet-Affect (Strapparava et al. 2004) or the NRC emotion lexicon (Mohammad & Turney 2013). Other methods include KGs that either store commonsense knowledge, such as ConceptNet (Liu & Singh 2004) or are more domain specific such as SenticNet (Cambria et al. 2018) and store both common sense knowledge and sentiment information. Ji, Pan, Cambria, Marttinen & Yu

(2020) have defined that the main difference between KGs and KBs is that KGs have an innate semantic structure whilst other KBs may lack this.

In the following section different flavours of KBs will be briefly introduced for both commonsense reasoning and SA. Furthermore it will be outlined how these KBs are commonly created. Then an overview will be given of common methods to create embedding representations from KGs and how sentiment has been incorporated into knowledge representations. Furthermore, an overview will be given of the intersection of language modelling and knowledge representation.

Many knowledge bases contain a range of either domain specific or commonsense knowledge, where the latter aims to capture a range of different aspects of the human experience, such as social or physical concepts of everyday life (Liu & Singh 2004). One important aspect of the human experience is the way emotions influence actions or thoughts (Hwang & Matsumoto 2013). It has been commonly acknowledged that words are not only associated to emotions (Mohammad 2012), but also that emotion properties can be inferred from emotion words (Strapparava et al. 2004). Thus, it is intuitive to represent words associated with emotions in a knowledge base, which can help us gain better insight into how humans experience the world and the associated concepts.

Knowledge graphs are a popular means to represent KBs as they provide effective semantics to data through relationships and efficient means to traverse the captured knowledge (Ehrlinger & Wöß 2016, Bordes & Gabrilovich 2014). Some of the most successful and widely utilised KGs are build and maintained by organisations such as Facebook (Facebook 2020), Google (Inc 2020) or IBM (Rajshree 2017) for a variety of different tasks such as finding, extracting or organising entities (e.g., books, products, people).

The use cases of these KGs are manifold and span from social relationship analysis (Ugander et al. 2011) to improve web search engine results (Steiner et al. 2012).

Other KGs have been constructed from multiple data sources, including remote sensing data (Bückner et al. 2002) or natural language, such as news story lines (Vossen et al. 2015). KGs have been successfully applied to a wide variety of different natural language understanding tasks, such as biomedical event extraction (Liu et al. 2014), stock price prediction (Liu et al. 2019) and Sentiment Analysis (Ma, Peng & Cambria 2018). KGs created from natural language often use sources such as procedural texts (Das et al. 2018), Wikipedia (Exner & Nugues 2012) and other webpages (Fan, Gardent, Braud & Bordes 2019) or are event-centric using news articles or headlines to extract triples (Vossen et al. 2015, Rospocher et al. 2016). However, whilst this type of language data is raw and provides its own unique challenges, it is syntactically well structured when compared to the terminology and language used on social media data. Textual data taken from social media platforms also often contains colloquialisms, emojis or hashtags that are often not found in normal text, but still carry important meaning and relations that are useful (Novak et al. 2015) and can in some cases contain more affective content compared to other textual data (Champoux et al. 2012). Research conducted by Mohammad (2012) has found that tweets that contain hashtags consisting of emotion keywords such as ‘*sadness*’ are good indicators of the overall sentiment of a tweet. Many efforts in the creation of KBs have focused on automatic creation of commonsense KBs such as ConceptNet (Liu & Singh 2004) or WordNet (Miller 1995). Whilst a KB will contain words that may directly refer to an affect, e.g., ‘*happy*’ or describe some commonsense concepts that are associated to what an emotion is or how it is expressed, it arguably fails to give insight into how humans feel about the world or its associated concepts.

### 2.5.1 Emotion Knowledge Bases

There are a number of knowledge bases which have been created with the specific intention to capture human emotions, through both fine-grained emotions and polarities that are associated to words. These often include lexicons, ontologies or KGs, where some of the more advanced KBs also rely on linguistic rules in order to improve accuracy of the KB.

Research conducted by Cambria et al. (2010) created a new KB resource for opinion mining, where a collection of polarity concepts was created. There have been many iterations of this work leading to the latest release of SenticNet 5 (Cambria et al. 2018), which uses Recurrent Neural Networks to discover concept primitives. OntoSenticNet (Dragoni et al. 2018) was build on top of SenticNet as an ontology for SA tasks. Further work on SenticNet has seen new techniques developed that learn polarities of new concepts and therefore increase SenticNet's commonsense affective concepts (Ofek et al. 2016). Another commonly used resource is the NRC lexicon (Mohammad & Turney 2013) which is a KB created through human annotation using Amazon Mechanical Turk. In this work around 14,000 words are annotated for both polarities and fine-grained emotions based on the emotion theory proposed by Plutchik (1984). The NRC lexicon also contains a suite of different resources that include an emotion hashtag lexicon (Mohammad & Kiritchenko 2015) and support in different languages (Kiritchenko et al. 2016). Further work by Mohammad et al. (2013) described the creation of a lexicon based on tweets containing positive and negative emoticons, called Sentiment140 lexicon.

WordNet Affect (Strapparava et al. 2004) is an extension of the already existing KB WordNet Domains, where synsets were labelled with affective concepts. In this work, a resource referred to as AFFECT that contains words associated to emotions, feelings and attitudes amongst other categories is used. The content of this lexicon is then projected onto WordNet, but required manual annotation to complete.



Research conducted by Cambria et al. (2015) proposes AffectiveSpace which is a resource created by aligning both ConceptNet and WordNet Affect resulting in an n-dimensional vector space such that reasoning by analogy is enabled. The Human Emotion Ontology was proposed by Grassi (2009) for the annotation of multimedia data and is based on the use of Ekman’s six basic emotions (Ekman et al. 1987) where concepts such as arousal, valence and dominance have been added. Work by Baccianella et al. (2010) introduces SentiWordNet, which is a lexicon based on WordNet synsets that contains three numerical scores to depict positive, negative or neutral stance of an entry. Oneto et al. (2017) introduce AffectNet, which is a common sense KB that contains both concept-level features and semantic links to aspects and their sentiment polarity. Thakor & Sasi (2015) have presented a model that looks at retrieving tweets that contain negative feedback and comments on postal services in the UK, USA and Canada with the goal to extract the reasons for customer’s dissatisfaction. As part of this study an ontology based on Twitter data was built and then used to define the problem from the negative tweet. More recently, work by Buechel et al. (2020) has proposed a methodology to create arbitrarily large emotion lexicons for any given language in order to overcome the issues of manual annotation and scalability. Finally, it is important to note that there are a number of ethical considerations to be taken into account, when working with such resources or creating new ones (Mohammad 2020).

### 2.5.2 Creating Knowledge Graphs

Knowledge base is a term used to describe a number of different computational resources that can include lexicons, ontologies and knowledge graphs. Knowledge graphs typically consist of a mapping between relations and entities (Paulheim 2017), where in KGs created from natural language these mappings are captured through the *Subject-Predicate-Object* structure, which is commonly used in the

English language (Crystal & McLachlan 2004). This structure is often referred to as a ‘*relation triple*’, which describes the relations between arguments in any given sentence (Križ et al. 2014).

The creation of KGs from raw data has been broadly defined as utilising a number of operations and rules to form either one or multiple resources to create a new KG (Paulheim 2017). This is commonly done automatically with varying approaches from statistical (Dong, Gabrilovich, Heitz, Horn, Lao, Murphy, Strohmann, Sun & Zhang 2014, Niu 1912), to neural (Bosselut et al. 2019), different existing tools (Exner & Nugues 2012), and data resources (Vossen et al. 2015, Rospocher et al. 2016). Knowledge graphs created from natural language have been used in a number of NLP tasks such as co-reference resolution (Ponzetto & Strube 2006), Open Information Extraction (Fader et al. 2011) or Question-Answering (Augenstein et al. 2012). One of the most popular KGs for natural language tasks is ConceptNet (Liu & Singh 2004), which is a commonsense KB that is used for real-world applications. Other commonsense KBs include Freebase (Bollacker et al. 2008) or DBpedia (Auer et al. 2007), which both extract information from Wikipedia to represent world knowledge. The remaining class of KBs focuses on specific domains to represent knowledge such as KBs that contain knowledge about legal texts (Montiel-Ponsoda et al. 2018), medical records (Rotmensch et al. 2017) or life sciences (Ernst et al. 2014). Other work by Distiawan et al. (2019) has focused on completing Wikidata utilising Wikipedia sentences by using a novel n-gram based attention mechanism.

There is some commonality in the approaches used when creating new KBs from natural language where there are a number of different methods and tools utilised. Obtaining triples from natural language is commonly done by extracting the main verb, subject and object of the sentence (Fader et al. 2011, Cattoni et al. 2012). Work by Schmitz et al. (2012) incorporates information through verbs, nouns and other syntactic structures such as adjectives that carry important relational

information. Common methods to refine this process include co-reference resolution, named-entity recognition, Part-Of-Speech tagging or dependency parsing (Nakashole et al. 2011). Often, existing NLP toolkits are used to complete this task (Carlson et al. 2010, Exner & Nugues 2012, Niu 1912), where existing tools exist as part of a new framework that creates the new KBs (Kertkeidkachorn & Ichise 2018). Resources to create new KBs and triples also vary, where web pages (Fan, Gardent, Braud & Bordes 2019, Nakashole et al. 2011), Wikipedia (Exner & Nugues 2012, Kertkeidkachorn & Ichise 2018) or newspapers (Augenstein et al. 2012) are often used. Another common step that is taken after the creation of a new KB is to map it onto an existing larger KB (Augenstein et al. 2012). One of the most commonly used KBs in NLP tasks is WordNet which is described as a lexical database and as such is not linked through concepts and entities, but through 166,000 word-sense pairs that show semantic relations (Miller 1995). Another area of research is the completion of existing KBs where new nodes are being created. Most recently, research by Bosselut et al. (2019) introduced COMET, a method to construct a KB from ConceptNet using a Transformer Network to generate richer representations from natural language that are able to complete the existing KB. Overall, it can be seen that these KBs are often created at either end of an extreme scale that is human versus automatic creation of resources. Also, some resources in the field of SA have been created by extending existing work (e.g.: WordNet and WordNet Affect) or aligning new resources with existing KBs (e.g.: AffectiveSpace and ConceptNet). Finally, work by Mohammad (2020) gives a comprehensive overview over the ethical considerations that have to be taken into account when creating a resource. These include but are not limited to, coverage of words in a domain, socio-cultural biases and source errors.

### 2.5.3 Knowledge Graph Embeddings

Due to the highly structured nature of KGs it has traditionally been difficult to manipulate KGs and represent its triples without losing any underlying meaning. This has led to the proposal of a new research direction (Ji, Pan, Cambria, Marttinen & Yu 2020), namely KG embeddings that has the goal to generate high quality embeddings from KG information. Wang, Mao, Wang & Guo (2017) have described the aim of Knowledge Graph Embedding (KGE) as a way to embed entities and relations of a KG into a vector space which captures and maintains the structure of the KG.

Over recent years KG embeddings have commonly been used for tasks to improve KGs, where relation or link prediction (Chami et al. 2020, Xiao, Huang & Zhu 2015), KG completion (Bosselut et al. 2019, Lin et al. 2015) or event extraction (Liu et al. 2014) are common tasks. However, KG embeddings have been used outside of the space of KG applications. In the space of NLP, tasks such as co-reference resolution (Rahman & Ng 2011), question-answering (Bordes, Weston & Usunier 2014, Bordes, Chopra & Weston 2014) or information extraction (Wang, Mao, Wang & Guo 2017) have achieved state-of-the-art results through the use of KG embeddings. KGE has also been referred to as Knowledge Representation Learning (KRL) (Ji, Pan, Cambria, Marttinen & Yu 2020), however for the purpose of this work only the term KGE will be used. KGEs can be broadly split into two categories, where either KG embeddings are build on facts alone or using additional information (Wang, Mao, Wang & Guo 2017).

#### **Facts Alone**

Using facts alone to generate KG embeddings is a common technique and can be further grouped into translational distance, semantic matching and other

approaches.

A common approach is to firstly specify in what way entities  $e$  and relations  $r$  are represented in the vector space, where an entity is usually represented as a vector whilst a relation can be either a vector, matrix, tensor or Gaussian distribution. Secondly, a scoring function is used for each triple to determine the likelihood or plausibility of a triple (Ji, Pan, Cambria, Marttinen & Yu 2020), where triples existent in the KG score higher compared to triples that have not been observed. The final and third step is then to learn the embedding representation, which can be seen as an optimization problem. Two of the main differences of the different types of KG embeddings are in the nature of the scoring functions used to estimate the likelihood of a triple.

**Translational Distance Approaches** In Translational Distance Model Approaches (TDA), the scoring function is distance-based, where it measures the distance between two entities after a relation carried out a translation. Within this approach there are further subcategories, where the most popular approach is based on a learning model called *Trans E* and its later iterations. Other approaches include Gaussian embeddings and other unstructured models.

*Trans E* In order to gain an understanding of translational distance model, the TransE (Bordes et al. 2013) will be used to outline the basic intuition behind the approach. In TransE both entities and relations are represented in a vector space  $R_d$ , where given a triple  $(h, r, t)$  the relation  $r$  is used a translation vector. Therefore both embedded entities  $h$  and  $t$  can be connected to  $r$  with low error, where  $h+r \approx t$  and as long as  $(h, r, t)$  is true. Here the scoring function is then a negative distance between  $h+r$  and  $t$ , where if the triple is true the scoring function is larger. However, there are a number of limitations to this approach, including TransE not being able to learn 1-to-N relations which means that triples with a shared relation but separate

entities may be embedded in a similar vector space.

There have been a number of iterations of this model, such as TransH (Wang et al. 2014) to tackle such issues by introducing relation-specific entity embeddings, where the goal is to learn specific and distinct entity representations given a relation. This has led to introducing models that improved the aforementioned disadvantages, but has also led to an increase of complexity and decrease of efficiency (Lin et al. 2015). Other work such as TransD (Ji et al. 2015) and TransSparse (Fan et al. 2014) have reused scoring functions of TransR (Lin et al. 2017), but improved on efficiency by reducing the number of parameters required. In TransD this was done by decomposing the projection matrix into a product of two vectors and the introduction of an additional mapping vector. In TransSparse this was achieved by utilising two versions called ‘share’ and ‘separate’, that both utilise one and two sparse projection matrices respectively. Another major shift was then in learning models that followed, where TransM (Fan et al. 2014), ManifoldE (Xiao, Huang & Zhu 2015), TransF (Feng et al. 2016) and TransA (Xiao, Huang, Hao & Zhu 2015a) all introduced a separate scoring function.

*Gaussian Embeddings* All models introduced so far have treated triples as non-random points in vector space and uncertainty has not been taken into account. Therefore work proposed by He et al. (2015) has used multi-variate Gaussian distributions to represent triples as random vectors, where the distances of triples are scored by using two random vectors of  $t-h$  and  $r$ . Here distance is measured through the use of the Kullback-Leibler divergence and the probability of the inner product. This enabled the proposed model, called KG2E to learn and represent uncertainties of triple representations effectively. Other work using Gaussian distributions, was introduced by Xiao, Huang, Hao & Zhu (2015b), where a mixture of Gaussian distributions is used to take into account relations potentially having more than one semantic meaning.

*Other unstructured models* There are a few learning models that have looked at using unstructured approaches to modelling triples, where for example one approach set all relations  $r = 0$  (Bordes et al. 2012). One of the major disadvantages of such an approach is that the learning model is not able to distinguish between different types of relations.

**Semantic Matching Approaches** In Semantic Matching Approaches (SMA), the scoring function is similarity based, where it measures entities and relations in the vector space by matching latent semantics.

*Bilinear Scoring Functions* A popular model in the area of Semantic Matching Approaches called RESCAL was introduced by Nickel et al. (2011) and has henceforth been extended and improved for efficiency. RESCAL is a bilinear model where each entity is a vector and each relation is a matrix that learns pairwise interactions between latent factors, so that the score of a triple captures the interactions between entities  $h$  and  $t$ . This model's efficiency was improved by Yang et al. (2014), where each relation  $r$  has a vector embedding and the scoring function captures pairwise interactions between entities only along the same dimension. However, this also means that the model is only able to learn symmetric relations and cannot capture the full complexity of a KG. Research conducted by Nickel et al. (2016) represents both entities and relations as vectors, where for each entity in a triple a representation is computed by using the circular correlation operation. This compositional vector is then matched with a relation representation and a scoring function is used to compute the triple output. Due to circular correlations not being commutative the model can capture asymmetric relations. Complex Embeddings as introduced by Trouillon et al. (2016) also aims to learn asymmetric relations of triples, where a relation embedding is not represented in a real-valued space but a complex valued space. For this the scoring function has also changed to be able to compute scores for asymmetric relations depending on the order of

entities. Analogy is another extension of RESCAL and used to learn analogical features of triples.

*Neural Networks* A number of different earlier neural network based approaches have been used to generate KG embeddings, where many early approaches have used architectures such as Neural Tensor Networks (NTN) (Socher, Chen, Manning & Ng 2013) or a MLP (Dong, Gabrilovich, Heitz, Horn, Lao, Murphy, Strohmann, Sun & Zhang 2014). The approach uses embeddings generated by a NTN for completing a KG, where the network’s input layer projects entities to a vector embedding space (Socher, Chen, Manning & Ng 2013). Then the two entities are combined by a relation tensor and propagated to a non-linear hidden layer, where at the last step a relation-specific linear output layer assigns a score. In the approach using an MLP, each entity and relation in a triple are a single vector, which are concatenated in the input layer, mapped to a non-linear hidden layer and afterwards an output is computed by a linear output layer (Dong, Wei, Tan, Tang, Zhou & Xu 2014).

Since the popularisation of neural network architectures, such as CNNs, Transformers and GNNs in a range of different NLP tasks, they have also been used for learning KGEs. More specifically, Dettmers et al. (2018) have used CNNs to learn interactions between entities and relations by using a 2D convolutional model to reshape head and relation into a 2D matrix. This work was subsequently simplified by Nguyen et al. (2018), which used a 1D relation-specific convolution. Inspired by the idea of recurrence made successful in RNNs Guo et al. (2019) developed a recurrent skip mechanism to differentiate between entities and relations in representations. Similarly, Transformer based methods have been used to build KG-Bert (Yao et al. 2019b) and CoKE (Wang et al. 2019) in order to take advantage of learning contextualised representations. Research proposed by Schlichtkrull et al. (2018),



Kipf & Welling (2016), Nathani et al. (2019) utilise GNNs, introducing R-GCN for relation-specific transformations, GCN for graph encoding and Graph Attention Network for modelling multi-hop neighbourhood features respectively.

### **Additional Information**

Some methods use other information in addition to the triples present in the KG to generate embeddings. These can range from entity types, relation paths, logic rules or textual descriptions.

Entity types are often already present in a KG as it is encoded in a relation, where Xie et al. (2016) have used entity types as constraints during the training process. Relation paths are usually denoted as a sequence of relations that connect two or more entities in a KG, where one of the biggest challenges lies in how to represent a relation path in the same vector space as entities and relations (Wang, Mao, Wang & Guo 2017). In order to take advantage of learning long-term relational dependencies in KGs, RNNs have been used to learn vector representations via relation paths (Gardner et al. 2014, Neelakantan et al. 2015). Textual descriptions are also commonly used, especially because entities in KGs are often described through text and therefore contain additional semantic information. However, there have also been attempts to add other information through sources such as news articles. Work by Liu et al. (2019) uses a GRU network, a KB and sentiment of news to forecast stock prices. The work is carried out using Chinese news sources to extract relationships between large companies and their customers. More recent work has also focused on using GCNs to generate embeddings from logic rules and apply it to the task of visual relation prediction (Yaqi et al. 2019).

Other forms of information that have been included are, for example, labels where research conducted by Miyazaki et al. (2019) proposes a label embedding method for social media-based emergency response using twitter data. Work by Ma et al.

(2019) proposes a new method for news embeddings based on a news network, where a subnode model is used to enable unseen news nodes outside of existing network enrich a news vector with associated features. Weston et al. (2013) uses an energy-based model to predict relationships based on mentions and encodes the interactions amongst entities and their relationships. For this two different scoring functions are used to learn low-dimensional embeddings of words, entities. Research proposed by Fan, Gardent, Braud & Bordes (2019) argues that in order to accurately extract and process long and diverse textual web results, it is useful to restructure free text into individual KGs that can be removed, merged or reduced. For this Graph Attribute Embeddings are introduced, which are then fed to a learning model.

### **Training KG Embeddings**

There are two main assumptions when training KG embedding models (Wang, Mao, Wang & Guo 2017), which either work on the Open World Assumption (OWA) or Closed World Assumption (CWA). The first assumption states that a KG contains true facts (triples) and all missing facts can be either false or missing, whilst the second assumption states that all facts (triples) that are not contained in a KG are simply false and new entities cannot be easily added.

*OWA* In this approach training KG embeddings is done by using either logistic loss or pairwise loss, where often individual embedding models introduce additional constraints and/ or regulations (Wang, Mao, Wang & Guo 2017). Furthermore, it has been found that logistic loss achieves better results in semantic matching approaches, whilst pairwise loss is often more appropriate for translation distance approaches. A common first step then is to initialise embeddings by using either random uniform distribution, Gaussian distributions, use a corpus to pretrain embeddings or use a simpler model to generate embeddings first. The next step is then to generate negative samples, where a naive approach is to replace entities

of a triple  $(h, t, r)$  randomly or corrupting the relation. However, this can lead to an increased amount of false-negative samples and in order to overcome this it has been proposed to use set probabilities for replacing  $h$  or  $t$  (Wang, Mao, Wang & Guo 2017).

*CWA* Under the closed world assumption, entity and relation embeddings can be learned through using squared loss, but also logistic loss or absolute loss. This often means that for true triples the score is often close to 1 and for false triples the score is closer to 0. There are two main disadvantages to training under CWA, where it cannot be applied on incomplete KGs and the majority of commercial and popular KGs are incomplete. Furthermore it has often been found that using KG embeddings trained under CWA perform worse in downstream tasks and also introduce more negative examples (Wang, Mao, Wang & Guo 2017).

### **Application of KG embeddings in NLP downstream tasks**

Research conducted by Peters et al. (2019) proposed a new methodology to embed multiple KBs into language models (LM) such as BERT. More specifically, they use both WordNet and Wikipedia data to extract entity spans from input text and then link it to a relevant embedding from the KB using a Knowledge Attention and Recontextualization (KAR) mechanism. This is done to enable ‘long-range interactions between contextual word embeddings’ and entities in the context. As a result of this KAR is inserted into BERT to generate knowledge representations that contain general knowledge, which are useful for a number of downstream tasks. Other areas that have benefited from using KGEs are spoken language understanding (Chekol et al. 2017) or short text representation learning (Wang, Wang, Zhang & Yan 2017).

**Language Modelling with Knowledge Graphs** On the one hand recent success in LMs has been largely based on the development of new neural network architectures, such as Transformers. On the other hand, recent trends in KGEs are also based on neural network architectures, such as GNNs. However, there are still many challenges when it comes to incorporating structured knowledge into a LM, where the KGE capturing the knowledge needs to be encoded and extracted effectively and both a resulting knowledge representation and language representation needs to be aligned in the same vector space (Zhang, Han, Liu, Jiang, Sun & Liu 2019).

The Enhanced Language Representation with Informative Entities (ERNIE) was first introduced by Zhang, Han, Liu, Jiang, Sun & Liu (2019) to incorporate knowledge into LMs. In their work entity mentions are first extracted from a given text and then matched with mentions in a KG. For this TransE is used to generate a KGE, which is then used as input to ERNIE. ERNIE also utilises a masked language model and uses next sentence prediction as a training objective, but introduces an additional new training objective that randomly masks named entity alignments and the model then selects entities itself from the KG to complete alignments. The model then has the following components (see Figure 2.19), where there is a T-Encoder capturing semantic and lexical features of input text and is identical to a BERT in its implementation. Then a K-Encoder integrates the knowledge information through the output of TransE's KGEs. The final output of the K-encoder is an information fusion layer, which generates an embedding containing both token and entity knowledge. Since the ERNIE was introduced there have been new iterations of it (Sun, Wang, Li, Feng, Chen, Zhang, Tian, Zhu, Tian & Wu 2019), which improve the framework by introduction continual multi-task learning (Sun et al. 2020). Work by Logan et al. (2019) introduces a new model called the knowledge graph language model (KGLM), where the main aim is to generate entities and facts from a KG. This architecture is a LSTM based LM and has a dynamically growing

local KG that contains already existing entities and their related entities. KGEs are generated based on the idea of TransE to extract entity and relation embeddings. He et al. (2019) propose BERT-MK, a LM that incorporates knowledge from a medical KG.

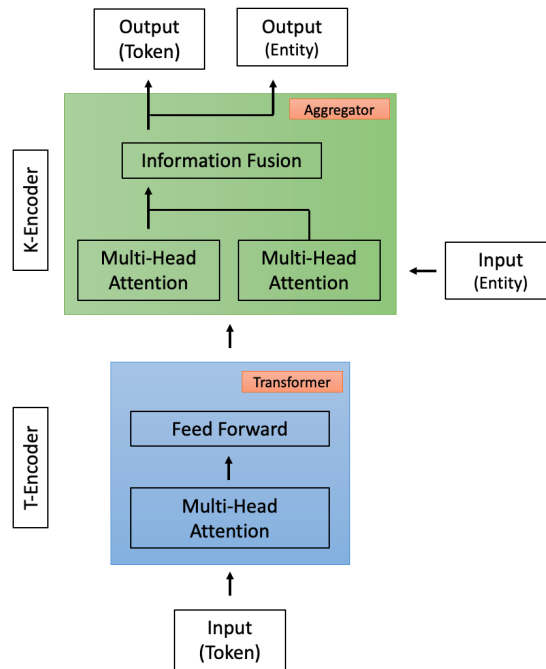


Figure 2.19: Architecture of Language Model ‘ERNIE’ adapted from Zhang, Han, Liu, Jiang, Sun & Liu (2019)

Similarly to the aforementioned argument in language representation, the research mentioned in this section has not incorporated or considered to which extent sentiment is presented or even excluded in the learning process. Again, this may lead to similar representation of words in LMs that carry an opposite sentiment meaning.

#### 2.5.4 Knowledge Embeddings in SA

The usefulness and importance of KBs for Natural Language Understanding (NLU) tasks has long been acknowledged (Minsky 1988), however most KGs have been developed with the aim to represent common sense or domain specific knowledge

in a structured manner. However, the success of KGEs in a range of different NLP tasks, has also generated interest in the area of SA. However, there are not many KGs that were developed with the view of being used specifically for fine-grained emotion detection.

Therefore in this section a range of different KBs will be introduced that have been used to inject knowledge into embeddings.

In the SA subtask called ASBSA, the main goal is to identify a polarity that is associated to a specific aspect. Research conducted by Ma, Peng, Khan, Cambria & Hussain (2018) has proposed a knowledge-enriched network called SenticLSTM, which is a LSTM architecture that has an augmented cell and additional attention mechanisms (see Figure 2.20).

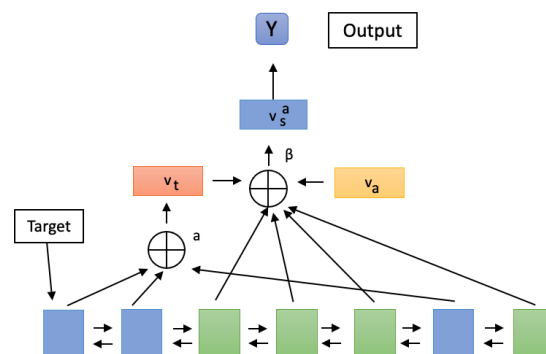


Figure 2.20: Architecture of SenticLSTM adapted from Ma, Peng, Khan, Cambria & Hussain (2018).

For this AffectiveSpace (Cambria 2016) is used to integrate concept level information into the SenticLSTM, because it is assumed that each concept has additional information that relates to the input word sequences. Therefore at each timestep embeddings of the concepts can be mapped to input sequences, where the average vector of all associated concepts is used as input. Furthermore, knowledge concepts are added to the cells input, output and forget gate in order to filter information as it propagates through the cell.

Another area of research that has garnered attention for integrating sentiment

knowledge is the prediction of stock market prices. Work proposed by Giatsoglou et al. (2017) uses the NRC emotion association lexicon in combination with Word2Vec to create new hybrid vectors. The hybrid vector is a concatenation of word embedding vectors and lexicon based vectors, where words with little or no information are being excluded to reduce the vector size. Research conducted by Kumara et al. (2018) introduces a two layer bidirectional LSTM with attention that is combined with a Support Vector Regressions model for stock market price prediction based on sentiment. To integrate knowledge into the network WordNet is used to learn KGEs and a number of sentiment lexicons (Gilbert & Hutto 2014, Baccianella et al. 2010) are employed in order to extract positive and negative words in the given text. Work by Liu et al. (2019) also leverages sentiment knowledge to predict stock market prices based on news articles and uses a GRU for prediction. In this work two lists of words are created that are associated with either a positive or negative sentiment, which then in turn generates a sentiment vector for each news item. Then KGEs are generated from a Enterprise KG based on TransR (Lin et al. 2017), where in the final step both KGEs and sentiment news vectors are modelled as a time series prediction problem. Research conducted by Fares et al. (2019) proposes LISA, an unsupervised word-level KG based LSA (Lexical Sentiment Analysis) framework, where a Lexical Affective Graph (LAG) is created based on WordNet and WordNet Affect. LISA has two main parts, where one allows for affective navigation and the other for affect propagation and look up.

Overall, it can be seen that a variety of approaches have been used in order to learn KGE for SA using existing sentiment and emotion KBs. However, up until this point much focus and effort has been on either using limited polarities, such as negative and positive. Furthermore, mostly lexical resources or resources that have been modified have been used in this kind of research.

## 2.6 Conclusion

In this chapter, it was firstly defined in section 2.1 what an emotion is and what kind of emotion theories exists. Then an overview of the two most popular emotion theories was given that are used in fine-grained emotion classification for NLP. Section 2.2 gave a broad overview of the field of sentiment analysis, its tasks and different approaches. It was noted how previously Twitter was only considered as sentence-level or short-sequence SA task, because of its length. Furthermore, this section has shown the disadvantages of only using either statistical or knowledge-based approaches to SA. Then it was outlined what hybrid approaches are and how they have been applied in other SA tasks. In section 2.3 it was first described how basic neural network architecture work and how they have been applied to SA tasks. Then an in-depth description of RNNs architectures was given, where the three main approaches to overcoming the vanishing and exploding gradient descent problem were described (see section 2.3.4). Furthermore, this section has shown how most existing methods have often not been tested on real world tasks or datasets. This means that oftentimes it is not clear or established if a new method or architecture does work for other tasks, especially in SA where often only reviews or the SST dataset are used as benchmarks. Therefore there is a need to (i) test this type of network on a variety of tasks (e.g.: fine-grained emotion detection) (ii) evaluate where these networks may fall short and need improving (see chapter 3.3). Section 2.4 introduced the field of language modelling and the most successful LMs were introduced. Then it was shown how they are used in SA tasks and it was shown that whilst most LMs consider context for a learned sequence, they never consider any form of emotion or sentiment. Therefore, the aforementioned LMs often fall short in achieving state-of-the-art results in a variety of SA tasks. Also, an overview is given of sentiment embeddings and approaches that have been used so far to generate them. It was found that whilst the existing approaches work, they



mostly work on detecting polarities or very specific SA tasks (e.g.: TABSA/ABSA). This section also outlined existing resources for SA tasks and where they fall short. Common disadvantages of existing KBs are outlined, which include the trade-off between having scalable or more fine-grained resources. Another disadvantage of these LMs and embeddings is that they often have similar vector representations for words that carry opposing sentiment or emotion meaning. Finally, this section looked at existing approaches to embed knowledge into neural networks via word embeddings. Similar, to previous findings it was noted that sentiment embeddings either focus on a small set of polarities that are scaleable or they focus on fine-grained emotions, but are smaller in size. Therefore, in chapter 4 popular LMs will be applied to the task of fine-grained emotion detection, where models such as BERT are applied to an SA task other than polarity detection with movie reviews. Then a new KG will be introduced that is capable of overcoming the aforementioned issues through utilising linguistic knowledge that is also scalable. Finally, chapter 4 will introduce a new method to learn sentiment embeddings that are then benchmarked on the task of fine-grained emotion detection.

## Chapter 3

# Fine-grained Emotion Classification in Tweets

There are several challenges that have to be taken into account when using recurrent neural networks to learn longer sequences, which include but are not limited to: (1) maintaining mid and short term memory, which is problematic when memorising long-term dependencies (Hochreiter et al. 2001) and (2) the vanishing and exploding gradient descent problem (Pascanu et al. 2012). In particular, it has been established previously (Hochreiter & Schmidhuber 1997*a*, Hochreiter et al. 2001) that any vanilla recurrent neural network trained with stochastic gradient descent on a sequence of more than *ten* time-steps will struggle to learn long-term dependencies. Therefore it could be argued that there is a need for more specialised learning models which can overcome these challenges, especially for tasks in Sentiment Analysis where there are often long sequences such as reviews (Mesnil et al. 2014) used. Tweets have previously been treated as shorter sequences or sentence-level tasks in sentiment analysis (Kouloumpis et al. 2011). However, it could be argued that this should no longer be the case especially since Twitter increased its allowed character limit from 140 to 280 (Twitter 2018*a*). Subsequently, many tweets could also be seen as

longer sequences, that face problems associated to other long sequence classification tasks in NLP. As a result of this it is even more important to classify long sequences accurately, because often important information that indicates a person's emotions, attitude or opinions are expressed through the use of emojis and often appear towards the end of a tweet (Novak et al. 2015). Therefore the use of multiscale RNNs to overcome the issue of learning more frequently occurring short-term information whilst retaining long-term information is investigated in this chapter.

In this chapter, it will firstly be outlined what ethical concerns have to be considered when conducting large scale data collection on social media platforms. Then the data collection process is outlined an overview is given over the amount of data that has been collected. The first experiment series in this chapter tests the hypothesis of whether tweets should be treated as long sequences and investigates the use of a hierarchical RNNs for this task. The second experiment series proposes a new learning model for the fine-grained emotion detection task. The learning model is then tested in two different experimental settings and three different evaluations are conducted. Finally, it is outlined how this work can progress in the future.

### **3.1 Ethical considerations in Data Collection**

Twitter is one of the largest, most popular and fastest-growing social media platforms (Chaffey 2016) that offers users to micro-blog about their opinions, emotions and comments towards events, products and other entities. Using micro-blogs for SA has many advantages as there is a restriction of characters (280 characters (Twitter 2017c)) which means that the number of sentences in a document is limited by default. Therefore it is argued that this enables researchers to get more concise tweets that may contain emotions/polarities from a micro-blog as the ability of the author to convey their message is limited (Kontopoulos et al. 2013).

It is also one of the most popular micro-blogging platforms in the research community (Kontopoulos et al. 2013) as it is one of the only social media platforms that still allows members of the public to stream public data through their API (Twitter 2017a). Although it was previously possible for researchers to access data from other social media platforms such as Instagram (Instagram 2017) or Facebook (Facebook 2017), it is now harder for the public and the academic community to get access. This is largely due to the fact that the commercial value of such data has increased rapidly over recent years, and data protection has become a focus-point for companies and the media (Chen et al. 2014). Furthermore, this is also partly due to new legislation and debates on personal privacy online, where laws such as the ‘General Data Protection Regulation’ (Office 2020) have come into force in Europe. The picture-sharing platform Instagram now only allows companies with clearly evidenced data protection rules and regulations to access some of their data provided it is for business purposes (Instagram 2017). However, there are several datasets which have been published by companies and research institutions alike in order to support the ongoing efforts of researchers and interested individuals (Stanford 2017). One of the disadvantages of this situation is that ongoing knowledge and information discovery is monopolised and somewhat limits the opportunities for open-source research. However, it also needs to be considered how the potential ‘free’ and ‘unlimited’ use of such data can be harmful when used for malicious purposes. More specifically, the uncontrolled use can lead to the development of technologies that increase social inequality as we all lead to findings that are not appropriate for multiple different communities (e.g.: further increase of racial inequality and poverty) (O’neil 2016). It can also lead to the violation of citizen’s privacy (Zhang 2018) and raises questions of consent in research. In a worst case scenario this type of social media data and the analysis of it can lead to threaten democracy by influencing citizens through tools such as fake news (Levy 2017). Therefore, a full ethics application has been made to help ensure that this project adheres to ethical

guidelines as much as possible. Furthermore, this project has also been reviewed at halfway point, when new legislation (e.g., GDPR) came into force during the duration of the project.

**Ethics** A full application for ethical approval has been submitted to the Research Ethics committee at the University of Hull and has been approved (Case no.: FEC\_2018\_17).

All of the current experiments are conducted using self-collected data from a social media platform called Twitter (Twitter 2017*b*). The data is collected on a weekly basis for several different categories. Tweets are collected through Twitter’s Public API (Twitter 2017*a*). Therefore some restrictions apply which include how much data can be collected, how much historical data (maximum of seven days) can be accessed, and only data from public Twitter profiles can be streamed for collection. Before starting the weekly collection of Twitter data, the ‘Social Media Research: A Guide to Ethics’ published by the University of Aberdeen (Townsend & Wallace 2016) was consulted. This comprehensive guide outlines several ethical considerations that have to be taken into account for social media research. It also includes discussions over private and public content, how consent can be given, anonymity and the risk of harm. Furthermore, it also includes a ‘Social Media Ethics Framework’ which guides researchers through a number of questions. The questions are tailored to any research project so that depending on the answer, it advises to investigate certain aspects of the research project further. This framework was completed for this research project, and some of the applicable considerations will be addressed in the following section.

In order to have enough data for Deep Learning experiments, it is essential to collect a high volume of data (around 1,000,000 Tweets or more). Due to the various restriction of Twitter’s Public API, this means that this will take a substantial

amount of time. In order to succeed in this goal without compromising research integrity or best practice due to time pressure, a data management process has been created. This process includes a Python script where only the keyword has to be changed for the different categories as well as a filing and recording system for any collected data. This allows for a time-efficient and accurate record of any collected data.

**Ethical Concerns** One ethical concern for this research project could be that the data which legally can be collected is not anonymised and therefore, individuals can be identified by features such as their username, profile pictures or location. In order to avoid this, a number of steps are taken to anonymise the data once it has been collected. This involves firstly stripping the data of any identifying extra information such as geo-location, time, nationality, profile pictures and usernames so that only the raw text data of a full tweet is leftover. This is especially important as it could be argued that some users may not be fully aware that this information can be accessed by anyone through the API as the profile is set to public by default. However, this regulation has been made clear in the governance documents of Twitter (Twitter 2017c). Then the tweet is searched for any tagged usernames, which can be identified by the '@' sign. If any tagged names are found the name is then replaced by the word 'name' in order to protect the anonymity of users.

The data for this work is stored and encrypted on the University of Hull's server under the University's security guidelines.

## 3.2 Data

Due to the advantages and changes mentioned in the previous section 3.1, it has been decided that for this PhD project, it is most useful to collect data from Twitter.

The data is collected every week for several different categories, which can include ‘affective’ words as the keyword that is streamed (i.e., ‘love’). The data collection was started in September 2017 and has finished in September 2019.

A number of different preprocessing steps are taken in order to reduce some of the noise in the data and manipulate it, so it is in the right input format for future experiments (Ravi & Ravi 2015). This is especially important due to the nature of the data collected as there is no restriction put upon Twitter users, except the limitation of characters per tweet. Therefore people are free to use any form of language in order to communicate their message. This can include colloquialism, well-known acronyms (e.g., BRB = Be Right Back) or Emojis (Agarwal et al. 2011).

- Firstly, the streamed data is checked for any duplicates or re-tweets, and if any are found, these tweets will be removed. The reason for removing the re-tweets are twofold: (i) collecting re-tweets would introduce duplicates as the original message is always included in the stream and (ii) it is hypothesised that re-tweets always involve some form of conversation which might be incomplete due to the way the API works and any emotions or polarities might be lost.
- Next the data is trimmed down so that only the text of the original tweets remains, this is important as information can be associated to a tweet such as information on age, gender or profile pictures. Although this information might be useful for other studies, it is for ethical reasons that it is removed. This also includes the anonymisation of all tagged people and user-names for more detailed information.
- Another important step of preprocessing the data is the manipulation of unicode that is automatically streamed with tweets that use emotions or other special characters. As part of this project, three methods will be used in order to see to which extent emoticons and special characters are contributing to

the accurate classification of more fine-grained emotions. One method involves using a dictionary to replace all unicode in the data with a placeholder value that corresponds to this Unicode (i.e.,  $u2026 = \dots$ ). The other method relies on removing all kind of unicode from the text data, but any trending hash-tags will be kept as they might contain the streamed keyword (i.e.: ‘#love’) or provide more information about the topic within the tweet. The final method involves the use of already existing tools such as ‘Ekphrasis’ (Baziotis et al. 2017b) to ensure tweets are preprocessed correctly.

The final number of unique tweets per category can be found in Table 3.1. Furthermore, it can be seen that the data collection of tweets containing keywords is currently closely monitored, based on two emotion theories by Plutchik (1984) and Ekman et al. (1987). Each tweet is assigned a label based on the emotion category its keyword belonged to (see Figure 3.1). Furthermore, it can be seen that for the category ‘Sadness’ only one emotion keyword was used instead of multiple. Therefore it could be argued that this may impact on the results as there is less variety in the data compared to other emotion categories.

Figures 3.1 and 3.2 show a two examples of tweets.

listening to everglow still makes me  
all soft and sad inside and makes  
me want to just curl up and cry for  
days 🥺🥹🥲

Figure 3.1: Example of a tweet

Argentinian steakhouse 🍖 surprise  
dinner & gift 🎁 thank you <user>  
❤️🧡💙 ( <location><number> in  
<location> ) <url>

Figure 3.2: Example of a tweet

More recently, similar approaches have been taken for this type of data collection. During the WASSA shared task on Implicit Emotion detection, a dataset of 155000 tweets was collected using (Twitter 2018b) and a number of keywords as synonyms (Klinger et al. 2018). These keywords were queried with the following pattern: ‘EMOTION KEYWORD’ + term (e.g.: that, because or when). One of the main challenges researchers face is the annotation of self-collected data (Glorot



Emotion	Tweets	Keywords	Length
Anger	44320	anger,angry, furious	17.42
Fear	76718	fear, scared, fearful	17.58
Disgust	41742	disgust, disgusting	16.06
Surprise	41647	surprise, surprising	16.20
Joy	184507	joy, happy	14.47
Sadness	48909	sad	16.78
Trust	69066	trust, trusting	-
Anticipation	52540	predict, anticipate, predicting	-
Total	559,449	-	15.75

Table 3.1: All collected EEK data over time

et al. 2011). There are many professional services available for annotation such as Amazon’s Mechanical Turk (Amazon 2018b) or Crowdfunder (CrowdFlower 2018). Although it is planned to use an annotation service for this dataset, it is important to note that only a subsection will be annotated as these services can be either costly as their quality is quite high or cheaper, but the quality is suffering.

### Ekman Emotion Keyword dataset

A subset of the collected tweets was used (Ekman Emotion Keyword (EEK) (see Table 3.2)), where a list of synonymous keywords for emotions can be found in Table 3.1. After the initial data collection tweets were filtered by those marked in the language tab as "English" and performed and checked for duplicates. All data was then anonymised including replacing all user-names with ‘@placeholder’ and masking URLs. Afterwards, a dictionary was used that contained all emotion keywords listed in 3.1 and replaced existing keywords in all tweets with the term *[keyword]*. This was done in order to prepare the data for the classification task outlined in section 3.3.

<b>Emotion</b>	<b>Tweets</b>
Anger	40,000
Fear	40,000
Disgust	40,000
Surprise	40,000
Joy	40,000
Sadness	40,000

Table 3.2: EEK Dataset

### 3.3 Dilated LSTM

The use of deep neural networks for sentiment analysis tasks has increased, but there are still some open challenges in classifying long sequences accurately. These challenges include but are not limited to (1) the fact that sequences can contain a varied vocabulary of input symbols and therefore models need to learn the long-term context or dependencies between symbols or (2) there is often limited data available that impacts on the performance of learning algorithms (Caragea et al. 2011).

In this section, the Dilated RNNs (DRNN) for emotion classification from tweets is proposed. DRNNs introduce skip connections into a standard RNN to increase the range of temporal dependencies that can be modelled. Experiments on sequence classification for language modelling on the Penn Treebank, pixel-by-pixel MNIST classification and speaker identification from audio (Chang et al. 2017) have shown to outperform competitive baselines such as standard LSTM/GRU architectures as well as more specialised models.

The following section outlines initial experiments using the Dilated LSTM as proposed by Chang et al. (2017) and as described in section 2.3.4. This is done to test the hypothesis that treating tweets as longer sequences can help obtain more accurate classification results in the fine-grained emotion classification task. Furthermore, the proposed learning model will be compared against the results achieved in the shared task held by Klinger et al. (2018).

### 3.3.1 Experimental setup

In these experiments, a standard LSTM and dilated Recurrent Neural Network (dRNN) with a dilated LSTM Neural Network (dLSTM) are compared with each other. For the final model, an additional embedding layer is added to the dilated LSTM, where an example of a part unrolled LSTM with three dilations is depicted in Figure 3.3 and a description can be found in section 2.3.4.

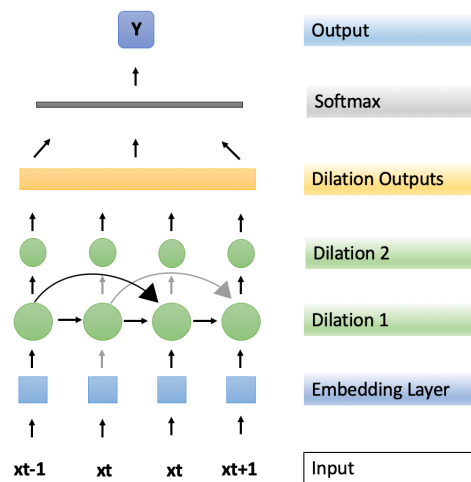


Figure 3.3: Dilated LSTM with an additional embedding layer.

Two different datasets are used for these experiments. The first dataset was released for the WASSA 2018 Implicit Emotion Shared Task (hereafter referred to as IEST dataset) (Klinger et al. 2018) and originally a baseline has been established using a maximum entropy classifier with L2 regularisation. The second dataset is the EEK dataset, where a detailed description is provided in section 3.2. A Vanilla LSTM with 2 hidden layers is used to establish the baseline for this dataset. All of the experiments were conducted using Tensorflow (Abadi et al. 2016), where four sets of experiments are performed on each learning model as outlined in Table 3.3.

For the experiments, 80% of the data is used as the training set, 10% as the validation set and the remaining 10% as test set. The hyperparameters of the model are set as follows; the learning rate was set to 0.001, and the batch size was set to 128

Number	Experiment type
1	Integer representation
2	Added Embedding Layer
3	Integer representation with Part-Of-Speech Tagging
4	Added Embedding Layer with Part-Of-Speech Tagging

Table 3.3: Description of experiment series for the Dilated LSTM

unless stated otherwise. For all experiments, a standard AdamOptimizer (Kingma & Ba 2014) was used. As in Chang et al. (2017) dilations are stacked hierarchically. There are three dilated layers with the dilations increasing exponentially starting at 1  $[1,2,4]$ . This means that each sub-LSTM has the following sequence length  $Dilation\ 1 = 200$ ,  $Dilation\ 2 = 100$ ,  $Dilation\ 3 = 50$  with a total of 20 hidden units per layer. The number of our dilations is established empirically, where the shortest input sequence to a sub RNN is no less than 40. This was partially motivated by Rozental & Fleischer (2018), who restricted the sequence length of each tweet to 40. A 100-dimensional Word2Vec (Mikolov, Sutskever, Chen, Corrado & Dean 2013) model was used to generate word embeddings for all experiments.

### 3.3.2 Results

Table 3.4 shows the results of all experiments for each model. The results using the dilated RNN for the IEST dataset only marginally outperforms the baseline when using an additional embedding layer 3.4. Results that only use integer representation of the input are performing worse, but this is not surprising seeing as it has long been known that word embeddings improve learning in models (Mikolov, Sutskever, Chen, Corrado & Dean 2013). Furthermore, it can be seen that the dLSTM is performing far better than the baseline. However, this is not true for the experiments where there is only an integer representation used as input. There are also slightly lower results for experiments where Part-Of-Speech-Tagging (POS) was used, and it could be argued that these results are due to the additional noise (e.g: not appropriately

annotating colloquial or non-standard language in a tweet) the POS tags created. Classification accuracies for the EEK dataset are higher compared to the IEST dataset. This difference is attributed to the difference in the size of both datasets. Furthermore, it was found that, similarly to the results of the IEST dataset, learning models that have word embeddings as input outperform learning models with integer representations as inputs. Moreover, it can be seen that using POS does not make a significant difference to the results.

<b>IEST Dataset</b>	<b>Max Ent Baseline</b>	<b>dRNN</b>	<b>dLSTM</b>
Integer Representation	59.88	41.40	37.5
Embedding Representation	59.88	61.17	78.78
Integer Representation + Part-Of-Speech-Tagging	59.88	41.40	36.71
Embedding Representation + Part-Of-Speech-Tagging	59.88	68.75	78.78
Amobee	-	-	71.45
<b>EEK Dataset</b>	<b>LSTM Baseline</b>	<b>dRNN</b>	<b>dLSTM</b>
Integer Representation	39.84	49.21	52.34
Embedding Representation	74.21	76.56	80.46
Integer Representation + Part-Of-Speech-Tagging	46.87	47.65	49.21
Embedding Representation + Part-Of-Speech-Tagging	74.21	75.78	80.46

Table 3.4: Results of experiments using a dilated LSTM using test set accuracy %

### 3.3.3 Conclusion

In this experiment series it was found that by both increasing both dataset and treating tweets as longer sequences, more accurate classification results can be obtained. Four initial experiments were conducted to test the aforementioned hypothesis of treating tweets as long sequences. It was found that the dilated LSTM with an additional embedding layer, performs above the baseline of 59.88% by over 7% on the IEST dataset. Furthermore, the model performs also best on the EEK dataset achieving an accuracy of 80.46%. In addition to this, it has been noted that adding Part-Of-Speech-Tagging does not significantly affect the learning model's results. Also, it was found that a vanilla Dilated RNN outperforms a simple LSTM on the EEK dataset. Therefore results indicate that whilst tweets cannot be considered as very long sequences, but that there is an argument to be made to not

treat them solely as short sequences. Finally, it could be argued that there is a need for a more specialised learning model which can overcome these challenges.

## 3.4 Bidirectional Dilated LSTM with Attention

In this section, the use of a bidirectional Dilated RNNs (DRNN) with attention for emotion classification from tweets is proposed. However, this type of learning model has as of now not been applied to a real-world task such as classifying fine-grained emotions in tweets, and it is hypothesised that the same advantages of this model apply to this. Furthermore, the proposed learning model is extended with an embedding layer, a bidirectional layer and a custom attention layer that focuses on the output of the networks dilations.

### 3.4.1 Bidirectional Dilated LSTM with Attention

For the Dilated LSTM, the implementation of recurrent skip connections with exponentially increasing dilations in a multi-layered learning model - as proposed by Chang et al. (2017) was followed - as it allows LSTMs to learn input sequences and their dependencies better. This means that temporal and complex data dependencies are learned on different layers. The dilated RNN has been developed for specifically learning longer sequences, where some of the shortcomings of RNNs are addressed.

The most important part of this architecture is the dilated recurrent skip connection, where  $c_t^{(l)}$  is the cell in layer  $l$  at time  $t$ :

$$c_t^{(l)} = f(x_t^{(l)}, c_{t-sl}^{(l)}). \quad (3.1)$$

$s^{(l)}$  is the skip length; or dilation of layer  $l$ ;  $x_t^{(l)}$  is the input to layer  $l$  at time  $t$ ; and  $f(\cdot)$  denotes a LSTM cell.

The exponentially increasing dilations across layers have been inspired by Van Den Oord et al. (2016);  $s^{(l)}$  denotes the dilation of the  $l$ -th layer, where  $M$  and  $L$  denote dilations at different layers:

$$s^{(l)} = M^{(l-1)}, l = 1, \dots, L. \quad (3.2)$$

As outlined by Chang et al. (2017) there are two main benefits to stacking exponentially dilated recurrent layers: (1) it enables different layers to focus on different temporal resolutions and (2) it reduces the length of paths between nodes at different time steps, which enables the network to learn more complex long-term dependencies. Therefore exponentially increasing dilations shortens any given sequence length at different layers.

The dilated RNN alleviates the problem of learning long sequences; however, not every word in a sequence has the same meaning or importance. Therefore we extend this network by (1) an embedding layer, (2) a bi-directional layer and (3) attention mechanism to summarise information from both directions of each word in a tweet and to be able to incorporate meaningful information into the learning model. The full architecture of the bidirectional Dilated LSTM (BiDLSTM) with attention is show in 3.4.

Each tweet contains  $t_i$  words where  $w_i t, t \in [0, T]$  represents the  $i$ th word in each tweet. First, the words are embedded to vectors through an embedding matrix  $W_e, x_{ij} = W_e w_{ij}$  and then a bidirectional LSTM is used to obtain information from both directions of each word. The bidirectional LSTM incorporates the forward LSTM  $\vec{h}_{t(i)}$  which reads each tweet from  $w_i 1$  to  $w_i T$  and a backward LSTM  $\overleftarrow{h}_{t(i)}$

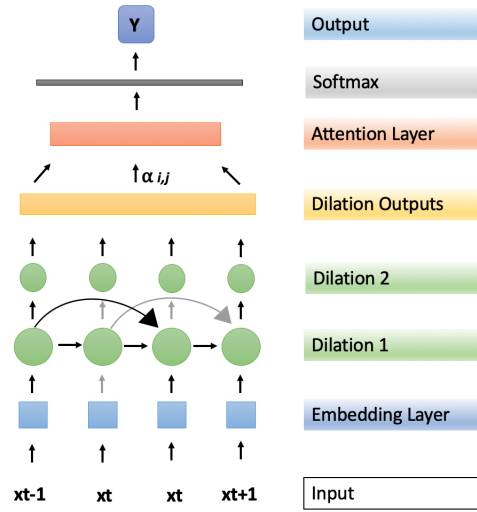


Figure 3.4: Bidirectional DLSTM with attention.

which reads words in each tweet from  $w_i T$  to  $w_i 1$ , where  $x_i t$  represents word vectors in an embedding matrix:

$$x_i t = W_e w_i t, t \in [1, T] \quad (3.3)$$

$$\vec{h}_{t(i)} = \overrightarrow{LSTM}(x_i t), t \in [1, T] \quad (3.4)$$

$$\overleftarrow{h}_{t(i)} = \overleftarrow{LSTM}(x_i t), t \in [1, T] \quad (3.5)$$

Then all outputs of the forward hidden state  $\vec{h}_t$  and backward hidden state  $\overleftarrow{h}_t$  are concatenated, where the output  $o$  utilises all the information in each tweet. The output  $o$  is then fed into the Dilated LSTM.

**Attention layer** The attention mechanism was first introduced by Bahdanau et al. (2015), but has since been used in a number of different tasks including machine translation (Luong et al. 2015), sentence pairs detection (Yin et al. 2016), neural



image captioning (Xu et al. 2015) and action recognition (Sharma et al. 2015).

The implementation of the attention mechanism is inspired by Yang et al. (2016), using attention to find words that are most important to the meaning of a sentence at document level. The output of the dilated LSTM is used as direct input into the attention layer, where  $O$  denotes the output of final layer  $L$  of the Dilated LSTM at time  $t_{+1}$ .

The *attention* for each word  $w$  in a sentence  $s$  is computed as follows, where  $u_{it}$  is the hidden representation of the dilated LSTM output,  $\alpha_{it}$  represents normalised alpha weights measuring the importance of each word and  $s_i$  is the sentence vector:

$$u_{it} = \tanh(O + b_w) \quad (3.6)$$

$$\alpha_{it} = \frac{\exp(u_{it}^T u_w)}{\sum_t \exp(u_{it}^T u_w)} \quad (3.7)$$

$$s_i = \sum_t \alpha_{it} O. \quad (3.8)$$

### 3.4.2 Experimental setup

There are two different datasets used in this experiment series, the IEST (Klinger et al. 2018) and EEK dataset (see section 3.2), where Table 3.5 shows a comparison of the two datasets in terms of their size and basic distribution of emotion categories represented in them. Furthermore the proposed learning model will be tested in two different settings to show its full effectiveness. This is due the winning model of the WASSA shared task (Rozenal & Fleischer 2018) having restricted the sequence length of each tweet to 40. Therefore the learning model will be tested in two

different sequence length settings, capped sequence length and full sequence length.

<b>Emotion</b>	<b>IEST</b>	<b>EEK</b>
Anger	25,384	40,000
Fear	25,387	40,000
Disgust	25,396	40,000
Surprise	25,402	40,000
Joy	25,377	40,000
Sadness	25,396	40,000

Table 3.5: Comparison of dataset distributions.

**Baselines** BiDLSTM with attention is evaluated against the following baselines:

- **DLSTM** – a dilated LSTM with hierarchically stacked dilations and hyper-parameters: learning rate: 0.001, batch size: 128, optimiser: Adam, dropout: 0.5
- **BiDLSTM** – a two-layer bidirectional dilated LSTM with a three-layer LSTM, hierarchically stacked dilations and the same hyper-parameters as the DLSTM.
- **BiLSTM** – a biLSTM with 2 layers and the following hyper-parameters: learning rate: 0.001, batch size: 128, optimiser: Adam, dropout: 0.5. This model is similar to recent work by Sachan et al. (2018) who used a single layer biLSTM to classify the Imdb movie review dataset into positive and negative reviews.
- **BiLSTM with attention** – a biLSTM with the following hyper-parameters: learning rate: 0.001, batch size: 128, optimiser: Adam, dropout: 0.5. This model is similar to recent work by Baziotis et al. (2017a) and Schoene & Dethlefs (2018).

- **CNN** – a CNN with 2-D convolution, a filter size of  $1,2$  and  $102$  filters, and a ReLU function. This learning model is similar to recent work by Dos Santos & Gatti (2014).
- **CNN-LSTM** – we follow the implementation of the learning model by Wang et al. (2016), using a CNN that is feeding into an LSTM. This model was used to predict the valence/arousal of ratings in textual data.
- **MaxEnt** – a maximum entropy classifier with L2 regularisation

For the proposed model, the number of dilations are established empirically. There are three dilated layers with the dilations increasing exponentially starting at  $1$   $[1,2,4]$ . This means that each sub-LSTM for the pruned sequence has the following sequence length  $[Dilation\ 1 = 40, Dilation\ 2 = 20, Dilation\ 3 = 10]$  with a total of 20 hidden units per layer. Each sub-LSTM for the longer sequence has the following sequence length:  $[Dilation\ 1 = 102, Dilation\ 2 = 51, Dilation\ 3 = 25]$ . 200-dimensional Word2Vec (Mikolov, Sutskever, Chen, Corrado & Dean 2013) word embeddings are used for all experiments.

In addition to the above neural networks, the learning model is compared against the winner of the 2019 WASSA IEST dataset. Rozental & Fleischer (2018) use a bidirectional GRU with an additional attention mechanism inspired by Bahdanau et al. (2014) and extra hidden layer and transfer learning, achieving an accuracy of 71.45%. All baselines will be evaluated in two conditions:

- **Capped length** – where the length of any sequence is capped to 40 in accordance with the WASSA IEST challenge winners.
- **Full length** – where the average full uncapped length of a sequence (maximum 103) is used. The intuition is that this condition will particularly reveal the advantages of the proposed learning model.

### 3.4.3 Results

The BiDLSTM with attention is compared to a number of different neural networks, using both vanilla neural networks and more specialised neural networks that have been used in sentiment analysis tasks. Then the results are also compared by two different sequence lengths. For the evaluation of the results four different metrics are used; test set accuracy, precision, recall and F1-score are used.

**Capped Sequences** Tables 3.6 and 3.7 show the results for capped sequence lengths for both the IEST and EEK dataset respectively. It can be seen that vanilla CNN and BiLSTM fall just short of the baselines established for this task. The CNN-LSTM and DLSTM architecture, both outperform their vanilla predecessors. The BiLSTM with attention and BiDLSTM surpass the baselines but fall short of the model proposed in the IEST task for both datasets. It can be seen that BiDLSTM with attention outperforms all previous models on the capped sequence length by over 14.43% for capped sequences and the IEST baseline by 11.24%. The results for capped sequence length using the IEST dataset (Table 3.6) show that our proposed model surpasses the ‘*Amobee*’ model’s result, however, this is only marginally. It is hypothesised that the reason the DLSTM, BiDLSTM and BiDLSTM with attention either fall short of the baselines or only marginally surpass them is due the model not being able to take full advantage of the full sequence length.

Learning Model	Test Acc.	Precision	Recall	F1-score
Max Entropy	58.4	0.59	0.57	0.58
CNN	43.17	0.44	0.42	0.43
CNN LSTM	55.42	0.56	0.54	0.55
BI LSTM	49.47	0.50	0.48	0.49
BI LSTM attention	58.60	0.60	0.56	0.58
DLSTM	56.44	0.57	0.55	0.56
BiDLSTM	67.96	0.68	0.67	0.67
Amobee	-	-	-	<b>71.45</b>
BiDLSTM attention	<b>72.83</b>	<b>0.74</b>	<b>0.71</b>	<b>0.72</b>

Table 3.6: Results of test accuracy in %, precision, recall and F-1 score for capped sequences (IEST Dataset).

Learning Model	Test Acc.	Precision	Recall	F1-score
Max Entropy	62.50	0.63	0.62	0.62
CNN	55.33	0.56	0.54	0.55
CNN LSTM	59.79	0.60	0.59	0.59
BI LSTM	60.19	0.61	0.59	0.60
BI LSTM attention	63.62	0.64	0.62	0.63
DLSTM	66.80	0.67	0.65	0.66
BiDLSTM	69.71	0.70	0.69	0.69
BiDLSTM attention	<b>73.74</b>	<b>0.75</b>	<b>0.72</b>	<b>0.73</b>

Table 3.7: Results of test accuracy in %, precision, recall and F-1 score for capped sequences (EEK dataset).

**Long sequences** Table 3.8 shows the results for the IEST dataset using full-length sequences, and Table 3.9 also shows the results for the full length for the EEK dataset. Similar to the results for the capped sequence length, the CNN and BiLSTM fall short of the established baselines. Only the CNN-LSTM improves the performance of the results, whereas for the long sequences the DLSTM, BiLSTM with attention and BiDSLTM surpasses the baselines of both datasets. The BiDLSTM with attention outperforms all models on the full-length sequences by over 20.36% on the EEK dataset and the IEST baseline by 18.47%. These results show that incorporating contextual information through the bidirectional layer and using attention to focus on the most important words in a tweet enhances the dilated LSTM’s ability to cope with longer sequences. This confirms that using more specialised learning models such as the DLSTM, BiDLSTM and BiDLSTM with attention allows us to capture information in longer sequences better.

Learning Model	Test Acc.	Precision	Recall	F1-score
Max Entropy	58.4	0.59	0.57	0.58
CNN	43.95	0.44	0.43	0.43
CNN LSTM	56.15	0.57	0.55	0.56
BI LSTM	51.73	0.52	0.51	0.51
BI LSTM attention	58.79	0.59	0.58	0.58
DLSTM	60.27	0.61	0.59	0.60
BiDLSTM	69.01	0.71	0.67	0.69
BiDLSTM attention	<b>78.76</b>	<b>0.79</b>	<b>0.78</b>	<b>0.78</b>

Table 3.8: Results of test accuracy in %, precision, recall and F-1 score full length (IEST dataset).

Learning Model	Test Acc.	Precision	Recall	F1-score
Max Entropy	62.50	0.63	0.62	0.62
CNN	55.12	0.56	0.54	0.55
CNN LSTM	60.11	0.61	0.59	0.60
BI LSTM	60.88	0.61	0.60	0.60
BI LSTM attention	62.70	0.63	0.62	0.62
DLSTM	67.18	0.68	0.66	0.67
BiDLSTM	69.53	0.71	0.68	0.69
BiDLSTM attention	<b>80.97</b>	<b>0.82</b>	<b>0.79</b>	<b>0.80</b>

Table 3.9: Results of test accuracy in %, precision, recall and F-1 score long (EEK dataset).

**Evaluation of Prediction Labels** In order to evaluate the performance of each model, 5,000 tweets per dataset were set aside, that have not been used during training or testing previously. The pretrained models were used to establish, which labels are hardest to predict for each network. Then the best performing learning model was compared with human performance on the same task. For this Amazon Mechanical Turk (Turk 2012) was used, where each tweet was annotated by three different annotators for the six emotion categories, yielding 15,000 annotations per dataset. All emotion words were replaced with the term ‘[Keyword]’, a sample tweet can be seen in Figure 3.5.

<user> <user> damn i just got [keyword] that i  
have no love life 😞 others : 🍷🍷🍷 me : 🍷🍷🍷

Figure 3.5: Example of a tweet shown to annotators

Confusion matrices are used to visualise the results for both datasets. Figures 3.6 and 3.7 both show the confusion matrices for the BiDLSTM with attention. Figures 3.6 and 3.7 shows that for the both datasets *Joy* was most accurately predicted emotion, whilst *Anger* (61.96 %) was often misclassified. Furthermore, it shows that *Anger* is more often confused with *Disgust* in both datasets.

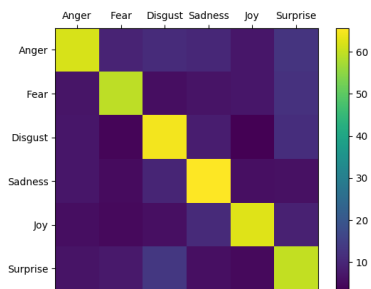


Figure 3.6: BiDLSTM attention (IEST)

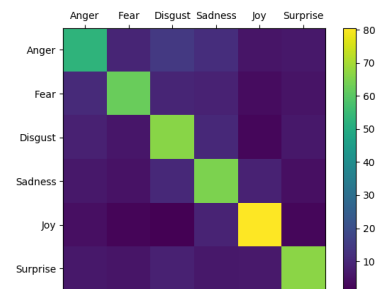


Figure 3.7: BiDLSTM attention (EEK)

Furthermore, each emotion was compared in both datasets in order to gain a better insight into how well each emotion is classified by the proposed learning model. We

use Precision, Recall and F-1 score as our evaluation metrics for both of the test datasets. Table 3.10 shows the emotion labels in the IEST dataset using the full sequence length, where the best performing emotion is *Joy* and the emotion *Anger* is most often misclassified. Table 3.11 also shows the label classification for the EEK dataset using the full sequence length, confirming that the same emotions, *Joy* and *Anger*, are also the most and least likely to be accurately classified.

Type	Precision	Recall	F1-score
<b>Anger</b>	<b>0.69</b>	<b>0.76</b>	<b>0.72</b>
Fear	0.69	0.83	0.75
Disgust	0.83	0.75	0.79
Sadness	0.76	0.78	0.77
<b>Joy</b>	<b>0.90</b>	<b>0.75</b>	<b>0.82</b>
Surprise	0.84	0.78	0.81
Average	0.79	0.78	0.78

Table 3.10: Evaluation metrics per emotion label - BiDLSMT with attention in % (IEST dataset).

Type	Precision	Recall	F1-score
<b>Anger</b>	<b>0.71</b>	<b>0.79</b>	<b>0.75</b>
Fear	0.74	0.86	0.80
Disgust	0.84	0.78	0.81
Sadness	0.78	0.81	0.79
<b>Joy</b>	<b>0.93</b>	<b>0.77</b>	<b>0.85</b>
Surprise	0.84	0.79	0.81
Average	0.81	0.80	0.80

Table 3.11: Evaluation metrics per emotion label - BiDLSMT with attention in % (EEK dataset).

**Evaluation of human annotation task** Afterwards the results for the human annotation task were evaluated, for the same test datasets. Figures 3.8 and 3.9 show the confusion matrices for the human annotators. Each confusion matrix shows the number of correctly and false predicted labels in percentages. It was found that for both datasets evaluated by humans that the most commonly correctly annotated emotion was *Joy* with 37.70% in the IEST and 41.80% in the EEK dataset. The emotion *Disgust* was least likely to be accurately annotated in both datasets.



Furthermore *Disgust* was most often mistaken for the emotion *Sadness* in both datasets and overall there were far fewer accurately predicted labels by the human annotators compared to the proposed learning model.

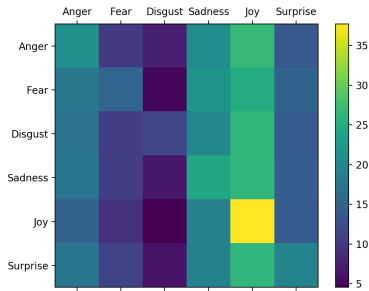


Figure 3.8: BiDLSTM attention (IEST)

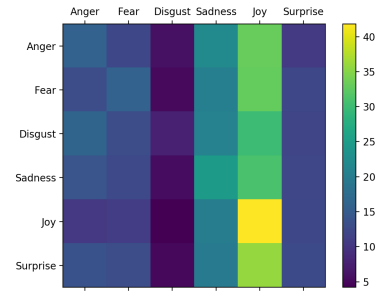


Figure 3.9: BiDLSTM attention (EEK)

Figure 3.10 shows an example of a tweet with its true label and the labels predicted by human annotators. It can be seen that for all three people annotating this tweet there was no agreement on the emotion label and no annotator picked the correct label. This illustrates how hard this task may be for humans as the keyword could have been replaced with a number of different emotion keywords and made sense.

***“that one girl from my art class said she feels [keyword] when she sees children and pregnant women”***

***True Label: Disgust***  
***Predicted Label: Surprise, Fear and Sadness***

Figure 3.10: A tweet illustrating the difficulty of the task for a human annotator to choose one emotion keyword.

**Probabilities of labels** Furthermore, 100 random test samples were evaluated to see the probability distribution of the output labels (see Figures 3.11 and 3.12). It could be argued that there might be some larger pattern that is detected by learning models when humans write about emotion that may not be detected by

humans on a qualitative basis. This might be due to the difficulty in the task where many emotions are closely related or overlapping such as *Disgust* and *Anger*, where humans were not able to interpret them correctly (Widen et al. 2004). Other studies have previously found that humans struggle to identify emotions in textual data due to the lack of extra information provided (e.g.: tone of voice or facial expression) and therefore often projecting their own emotional state and information (Riordan & Trichtinger 2017). However, this is not possible for any learning model and therefore might be the reason why they are better at detecting underlying patterns in this type of data.

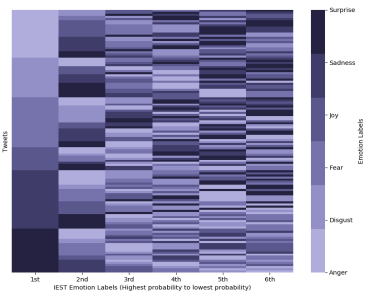


Figure 3.11: Visualisation of IEST Emotion labels based on the probability of accurate prediction - BiDLSTM with attention

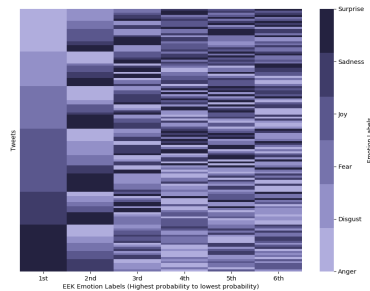


Figure 3.12: Visualisation of EEK Emotion labels based on the probability of accurate prediction - BiDLSTM with attention

### 3.4.4 Conclusion

In this chapter it was found that the proposed learning model, the bidirectional dilated LSTM with attention, performs above the baseline of 58.4% by over 14.43% on the WASSA shared task dataset. Furthermore, the learning model performs also best on our own dataset achieving an accuracy of 80.97%. It was also found that when using longer sequences we achieve better results with models that are more specialised compared to vanilla neural networks. Additionally, it was shown that when pruning our model to use a shorter input sequence it still outperforms

state-of-the art results. Also, it could be argued that treating tweets as longer sequences we can utilise more information in a tweet. Furthermore it was evaluated which labels are most likely predicted correctly by both humans and the BiDLSTM with attention. Finally, it was demonstrated that the task of accurately identifying the six emotion categories in tweets is considerably harder for humans compared to the learning model. This could largely be due to the amount of emotions projected by humans on an individual tweet which does not enable them to identify overall patterns on a qualitative basis.

## 3.5 Conclusion

In this chapter, the EEK dataset was introduced and ethical considerations of data collection were outlined. Then two different experiment series were described, where at the centre of it was the use of a dilated LSTM for fine-grained emotion classification. Furthermore, it was used to test the hypothesis that tweets should no longer be treated as a short sequence or sentence-level SA task. Then a new learning model was proposed to overcome issues the dilated LSTM struggles with on real-world tasks. This includes the addition of a bidirectional layer and a custom attention mechanism that focuses on the output of the dilated layer. It was shown that the proposed model improved classification accuracy and further proved how important it is to take advantage of the full sequence length of a tweet. However, it is important to note that the results of these algorithms may vary based on the data they are being fed. Additionally, a human evaluation of the same task was conducted using Amazon Mechanical Turk (Turk 2012). This showed how hard this task is on a human level and that the neural networks may pick up overarching patterns not accessible to humans. There are a number of future considerations and avenues of work considered for this type of research. Firstly, it has to be acknowledged that ethics and data protection are rapidly evolving and that for

future projects these changes need to be examined closely. Furthermore, it has to be more thoroughly considered how the use of such data and the subsequent development of any technologies or algorithms impact wider society. Concepts such as bias and fairness in deep learning need be further explored in order to gain substantial insight into the data and draw the right conclusions. On the other hand, it should also be considered how this kind of work can further the area of Sentiment Analysis, where it would be useful to scale up this work to other emotion theories. One of those theories would be the work by Plutchik (Plutchik 1984), which directly builds upon the emotion theory used in this work. In doing so more interesting and fine-grained insights could be generated from such data and also be used for purposes closely aligned to AI for social good research. Finally, this work should not only consider the use of other deep learning methods, but also the use of knowledge-based methodologies, which could help to overcome one of the key issues in SA research of achieving Natural Language Understanding (as outlined in section 2.2).

## Chapter 4

# Learning SSE representations using RELATE

A variety of neural network models have been at the core of ground-breaking results in several different research areas in natural language processing, which includes Machine Translation (Sutskever et al. 2014), Natural Language Generation (Konstas et al. 2017) and Sentiment Analysis (Dos Santos & Gatti 2014). However, one of the major disadvantages of these models is that they generally require large amounts of labelled data, whilst the knowledge contained in this data could be described in smaller more efficient knowledge representations (Yaqi et al. 2019). Therefore, recent research efforts have increasingly focused on developing methodologies that make prior knowledge accessible for neural networks, where one of the most flourishing approaches include embedding representations (Liang et al. 2019). Several existing resources are available for SA tasks that are used for learning Sentiment Specific Embedding (SSE) representations. These resources are either large, common-sense KGs that cover a limited amount of polarities/emotions or they are smaller in size, such as lexicons, which require costly human annotation and cover fine-grained emotions. Therefore using knowledge resources to learn SSE representations is either

limited by the low coverage of polarities/emotions or the overall size of a resource.

In order to incorporate knowledge into deep learning methods, researchers have focused on building many different resources. There have been three dominant ways to store knowledge in KBs, namely lexicons, ontologies and KGs. Over recent years particularly lexicons and KGs have been successfully created and applied to different NLP tasks. Most notably, the creation of lexicons such as WordNet (Miller 1995) has influenced tasks such as dependency parsing (Herrera et al. 2005). This has also led to the creation of lexicons specific for SA, such as WordNet-Affect (Strapparava et al. 2004) or the NRC emotion lexicon (Mohammad & Turney 2013). At the same time, word embeddings or bigger language models have not been focused on utilising sentiment or emotion information as part of the learning process. LMs, such as BERT (Devlin et al. 2018) or ELMO (Peters et al. 2019) have had considerable breakthroughs in various NLP tasks, but have been focused on incorporating context only. This has often led to words carrying opposing sentiment or emotional meaning having similar vector representations (Zhang, Wu & Dou 2019). Furthermore words carrying sentiment can also have different polarities or emotions when used in different topics or context (Ren et al. 2016). An example of this could be words such as *'good'* and *'bad'* or more subtle and fine-grained *'sad'* and *'upset'*. Overall, this can lead to worse SA performance (Tang, Wei, Qin, Yang, Liu & Zhou 2015), where a common approach to overcome this issue relies on using fixed embeddings or fine-tuning them with external resources. These methods range from post-editing already learned embeddings (Yu et al. 2017) to introducing separate 'sentiment channels' to learn new embeddings (Lan et al. 2016).

In this chapter, a new directed KG called RELATE is introduced. It is built to overcome both the issue of low coverage of emotions and the issue of scalability. RELATE is the first KG of its size to cover Ekman's six basic emotions. Furthermore, it is based on linguistic insights to incorporate the benefit of semantics

without relying on costly human annotation. Section 4.1 will describe which datasets have been used for this work. Then it will be outlined how the directed KG *RELATE* is created using linguistically inspired algorithms. Then several visualisations will be shown to give insight into what the triples of this KG look like and how much emotion information has been retained. The performance of RELATE is evaluated by learning SSE representations using a Graph Convolutional Neural Network (GCN) in section 4.2. The SSE are comprehensively compared against existing language models (LMs), such as BERT and ELMO in the task of fine-grained emotion classification. Finally, embedding visualisations will be compared to see how emotion keywords are represented in the embedding space.

## 4.1 Creating an emotion knowledge graph

In an effort to include emotions, new knowledge bases have been created either from scratch (Cambria et al. 2010, Mohammad & Turney 2013) or built on existing resources (Strapparava et al. 2004). In the following section a new knowledge base is introduced called *RELATE*, which is a knowledge graph build on Ekman’s six basic emotions. For this the Twitter data introduced in 3.2 is used (EEK dataset) to create this new directed knowledge graph. This data is used because it offers users to express their opinions, emotions and comments towards events, products and other entities (Chaffey 2016) freely within Twitter’s Terms of Services Twitter (2017c). Harnessing the relations, entities and sources from such data could provide useful insight and open up further research directions that could benefit the use of a larger emotion knowledge base.

The following section will outline the steps taken to preprocess the collected Twitter data. Then it will be shown how triples were obtained for this resource. Finally, two different methods of initial evaluation will be used to gain some insight into what

different emotion triples look like.

### 4.1.1 Preprocessing

There are several challenges when working with Twitter data because there is no restriction put upon Twitter users, except the limitation of characters per tweet (maximum of 250 characters per tweet). Therefore, people are free to use any form of language in order to communicate their message. This can include colloquialism, well-known acronyms (e.g., BRB = Be Right Back) or emojis (Agarwal et al. 2011) amongst others.

#### Text Preprocessing

Two well-known NLP tools, called *Ekphrasis* (Baziotis et al. 2017b) and *Spacy* (Explosion 2017) were used for anonymising and preprocessing the data, which is a common step in producing a new KG (Exner & Nugues 2012, Cattoni et al. 2012). Several preprocessing techniques that are often utilised in NLP tasks include tokenization, lemmatization and dependency parsing.

Firstly, *Ekphrasis* (Baziotis et al. 2017b) was used to replace and remove all usernames and URLs in each tweet with placeholders (e.g.: ‘< user >’). Whilst this is very effective for ensuring any tagged person is not mentioned, there is still a risk that a person is named by name only (e.g., ‘*Barack Obama*’). Arguably, this effects predominantly persons whose work is carried out in public through politics or the entertainment industry, and therefore it was decided not to remove those names. Furthermore, the functionality to annotate words that are elongated, repeated or are hashtags was used in this work. Also, it was decided to manually replace personal pronouns, such as (*i’m* to *i am*) and helping verbs (*wouldn’t* to *would not*) in order to make dependency parsing easier at a later point.



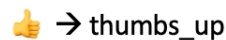


Figure 4.1: Example of a Emoji to textual description representation

**Emoji Preprocessing** Similarly to research conducted by Exner & Nugues (2012) it was chosen to remove all additional references, such as mark-ups or annotations and only keep the running text, which also includes the preprocessing of emojis in the data. In order to get accurate representations of each emoji used in this dataset, *Spacymoji* (Explosion 2017) was used to identify all emojis in the dataset. Then the description provided by Spacymoji was used for each entry to create a new textual representation as shown by Figure 4.1. Then each emoji in the dataset was replaced with its new textual representation. Overall there were over 1,069 different types of emojis found in this dataset. This concludes the initial data cleaning process.

### Sentence Segmentation

In order to obtain more high-quality triples, sentence segmentation was performed to distinguish between minor (irregular) and major (regular) sentences. Minor sentences usually follow an abnormal pattern and often compromise emotional noises, such as ‘*ugh!*’ or proverbs, e.g., ‘*easy come, easy go*’, which means that they often do not follow the rules of English grammar (Crystal & McLachlan 2004). Major sentences largely follow the rules of English grammar and most clauses contain some variation of the Subject-Verb-Object order (Crystal & McLachlan 2004).

Following this approach, minor and major sentences are distinguished by splitting each tweet based on three types of punctuation marks: full stop, question and exclamation mark. Then any sentence fragments that are less than two words long (see Algorithm 1) were filtered out; the author is aware that this approach does not ensure that all fragments containing less than two words are minor sentences and that there are no minor sentences mistaken for major sentences. Using this

approach 617,741 major sentences and 47,123 minor sentences are obtained. To obtain triples from tweets, only major sentences are used.

Major sentences can be split into two main types—simple and multiple—sentences where simple sentences often contain only one clause and multiple sentences often follow a pattern of ‘*clause*’ + ‘*linking word*’ + ‘*clause*’ and therefore contain multiple clauses. Overall, the four main types of clauses are either simple or linked by coordination or subordination (Crystal & McLachlan 2004). We created three different dictionaries that contained both coordinators and/or subordinators as outlined by Crystal & McLachlan (2004) that link clauses in order to get an estimate of how many of each type of sentences are in our dataset (see Table 4.1).

Type of Sentence	Number
Major	617,078
Minor	47,658
Simple	380,079
Compound	86,233
Complex	86,987
Compound Complex	63,779

Table 4.1: Types of Sentences in Dataset

```

input : Preprocessed Tweets
output: Tweets split into individual clauses

/* Each clause in a tweet is tokenized and checked for its
   overall length */
1 major_sentences = []
2 minor_sentences = []
3 for ( clause in preprocessed tweet ) {
4   | if length(clause) < 2 then
5   |   | major_sentences.append(clause) ;
6   | else
7   |   | minor_sentences.append(clause);
8   | end
9 }

```

**Algorithm 1:** Sentence Segmentation

**Coreference Resolution** Spacy’s neural coreference module (Explosion 2017) was used for this task. It was found that the majority of sentences do not contain any coreference, where the highest amount of coreference was around 17% in the compound and complex clauses.

### 4.1.2 Obtaining Triples

Constructing a KG from natural language is traditionally seen as a challenging task, because of the complex structure of language data (Kertkeidkachorn & Ichise 2018). A commonly used technique when creating KGs from text is using linguistic theory in the form of semantic parsing (Exner & Nugues 2012, Carlson et al. 2010, Fader et al. 2011). Therefore, it was decided to use Spacy’s dependency parser (Explosion 2017) to extract triples from the data. There are several different dependency parsers available for NLP tasks (Loper & Bird 2002, Chen & Manning 2014), and there are also parsers for Twitter data such as Tweepo (Kong et al. 2014); however, it was found that these were less effective in identifying an appropriate sentence structure. Furthermore, it was found that when parsing longer tweets, the parser struggled to identify the traditional *Subject-Verb-Object* for triples in a clause. One explanation for this could be due to the noise and complexity that is present in tweets. Therefore, it was decided to take this approach and split each clause of the type compound, complex and compound-complex based on coordinators as outlined by Crystal & McLachlan (2004). The main reason for this is due to clauses that are linked by coordinates can in principle stand as a sentence on their own (Crystal & McLachlan 2004). This is unlike clauses linked by subordinators as the clauses may be depended or embedded, and therefore grammatically dependent (Crystal & McLachlan 2004). Taking this approach yielded a total of 617,078 clauses before using dependency parsing.

Using an approach similar to Schmitz et al. (2012), all syntactic information is given

to extract the main subject, relation and object from the tweets and annotate each clause with syntactic information.

**Triples - no coreference** For clauses containing no coreference, triples were obtained as described in Algorithm 2.

First, it was iterated over the clause and found the main *ROOT* (see line 4) of the sentence. This is in most instances the main *verb* of the clause and is used as the relation linking the *subject* and the *object*. In the English language, there is a strict pattern for grammatically correct sentences that means the *subject* is usually found on the left side of the main *verb* and the *object* is on the right side of the *verb*. Whilst the identification of the main *verb* is more simple, it is more difficult to obtain the correct *subject* or *object*, especially when the main *verb* is at the start of a sentence. A common technique used is using Named Entity Recognition (NER), which is often deployed when extracting triples from natural language. However, in this instance it was decided not to move forward and use an existing NER tool due to the lack of coverage existing tools provided. Therefore, both *subject* and *object* are identified through the algorithm in each clause.

For the *subjects* (see line 10), it was decided to iterate to the left of the main *verb* and use these as the main *subject*. If there was no left side to the root word, then the whole clause was considered to search for the dependency tag '*subj*'. The *object* was identified by iterating over the subtree to the right of the *ROOT* word (see line 10). A subtree is a sequence which contains the syntactic descendants of the main token in this instance. This was done because the main *verb* is usually followed by a longer sequence of words, especially in sentences of the type complex-compound and complex. Getting the syntactic descendants of the root word to the right allowed the identification of dependencies tagged with '*obj*' that were related to the main root. Furthermore, '*modifiers*' and '*compounds*' were added to the identification of

```

input : Single clause with no coreference
output: A triple with the structure Subject-Verb-Object

/* Each word in a clause is annotated with a dependency tag
   */
1 relations = []
2 subjects = []
3 objects = []

/* Find the main ROOT in each clause and use it as the
   relation
   */
4 for ( dependency_tag in clause ) {
5 | if dependency_tag == 'ROOT' then
6 | | relations.append(word)
7 | end
8 }
/* Iterate over the subtree to the left of the ROOT and find
   main subject
   */
9 left_root=ROOT.lefts
10 if len(left_root)==0 then
11 | dependency_tag == 'SUBJ'
12 | subjects.append(word)
13 else
14 | subject.append(left_root)
15 end
/* Iterate over the subtree to the right of the ROOT and find
   main object
   */
16 right_root=ROOT.rights
17 o=""
18 for ( s in right_root.subtree ) {
19 | if dependency_tag == 'COMP' then
20 | | o+= word
21 | end
22 | if dependency_tag == 'MOD' then
23 | | o+= word
24 | end
25 | if dependency_tag == 'OBJ' then
26 | | o+= word
27 | | objects.append(o)
28 | else
29 | | objects.append("")
30 | end
31 }

```

**Algorithm 2:** Obtaining Triples with no coreference

the object on the right of the main *verb*.

An example output of a sentence for a triple with no coreference would look like this:

```

Dependencies:   i - NSUBJ can - AUX not - NEG imagine - ROOT
                   the - DET anger - DOBJ
Triple:         i Subject -- not imagine Relation -- anger Object

```

Figure 4.2: Example of a clause annotated with dependencies and the resulting triple.

**Triples - Coreference** Similarly, for the tweets that contain coreference, the same methodology was used to obtain the main *verb* and *object*. However, in order to identify the main subject, Spacy’s Explosion (2017) neural coreference module was used. For this, it was iterated over each clause and the first item in the returned list of coreferences was used as the main *subject*.

Once this final process was concluded, only triples where there were two or more types of each triple identified were kept, e.g.: *Subject* and *Object* or *Subject* and *Relation*. There were 490,299 remaining triples after completing the whole process. There are some downsides to this approach, where we only identify the main triple in each clause and not account for clauses that have more than one *Root*. However, it could be argued that there is no possibility to obtain a complete KG, where the trade-off between coverage and correctness of a KG is very common (Paulheim 2017).

### 4.1.3 Qualitative evaluation of RELATE

There are a number of ways in which this KG is evaluated. Firstly the distribution of emotion keywords in each triple is looked at. This is done to show how many triples in this directed knowledge graph carry affective meaning. Secondly, parts of the emotion keywords are visualised to show whether there are meaningful triples

present. Finally, in section 4.2 this new knowledge graph will be used as a resource to generate new embedding representations.

### Distribution of triples

Figure 4.2 shows the overall amount of triples and the number of individual entries for each triple. The total number of triples of the whole KG is *490,299*. It can be seen that the number of complete triples, containing all three entries is *323,106*, whilst there is only one triple that does not have an entry for the relation.

Triple Type	Number of Entries
Subject	400,166
Relation	464,295
Object	323,106

Table 4.2: Overall number of triples in the knowledge base.

Figure 4.3 shows the distribution of emotion keywords in the KG, and it can be seen that 42,646 triples contain emotion keywords, which means that there are around 8.69% of the whole KB that contain emotion keywords. It could be argued that when splitting tweets into its sentence segments that there is not necessarily an emotion keyword in each sentence segment. Furthermore, it can be seen that there is a large number of triples in the *happy* emotion category. This is not surprising, because of the initial distribution of keywords in the original dataset. Furthermore, it can be seen that the largest amount of emotion keywords are in the relation category, whilst subjects contain the least amount of emotion keywords.

### Visualisation of triples

In Figures 4.3, 4.4, 4.5, 4.6, 4.7 and 4.8 examples of where emotion keywords are relations are shown for each category. Networkx (Hagberg et al. 2005) is used for

Emotion	Subject	Relation	Object	Total
Anger	641	1143	2312	4096
Fear	701	4019	1421	6141
Disgust	484	2639	1748	4871
Surprise	687	3062	1062	4811
Joy	7632	5813	3539	16,984
Sadness	816	2539	2388	5743
<b>Total</b>	10,961	19,064	12,470	42,646

Table 4.3: Emotion keywords that are part of a triple

visualisation, and a randomly chosen set of examples of triples are used. This means that not all triples are visualised. Furthermore, the relation word itself is not visualised but symbolised through the arrow starting at the subject and pointing at the object. It can be seen in Figures 4.3, 4.4, and 4.5 that there are emoji representations part of a triple such as ‘downcast\_face\_with\_sweat’ or ‘neutral\_face’. Moreover, it can be seen that there are empty circles, which shows that the Subject and Object in this case (indicated by the arrows pointing towards and away from it) are empty.

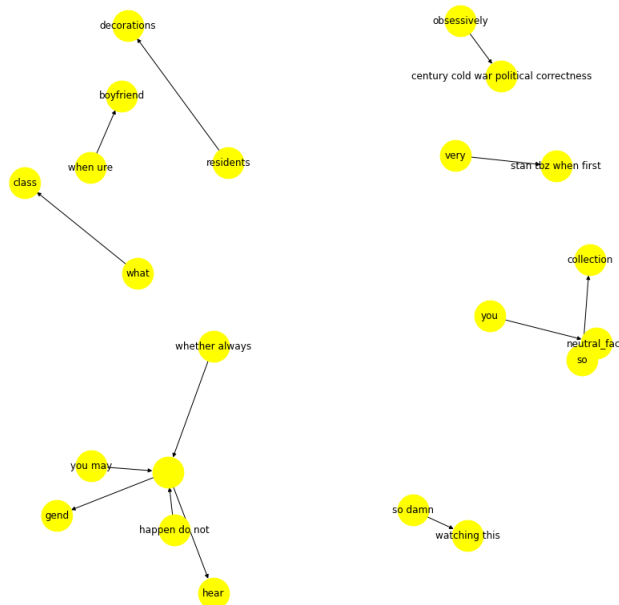


Figure 4.3: Example of ‘angry’ relations in the knowledge graph.





Figure 4.4: Example of 'sad' relations in the knowledge graph.

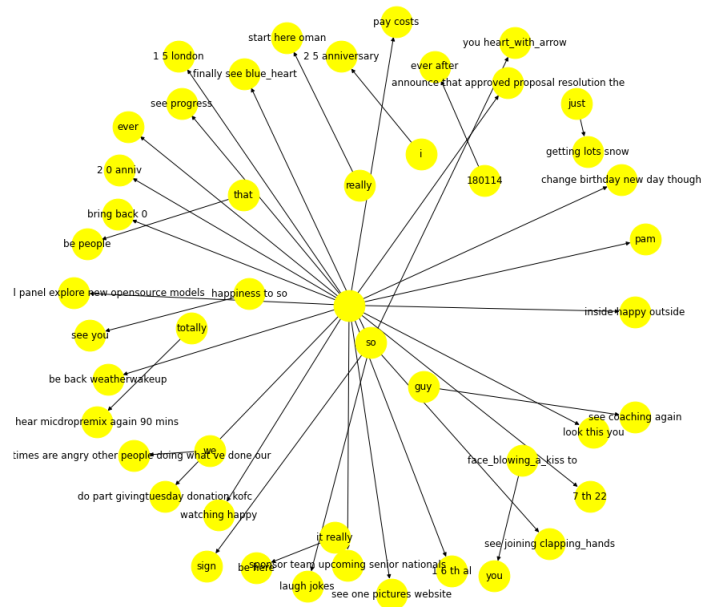


Figure 4.5: Example of 'happy' relations in the knowledge graph

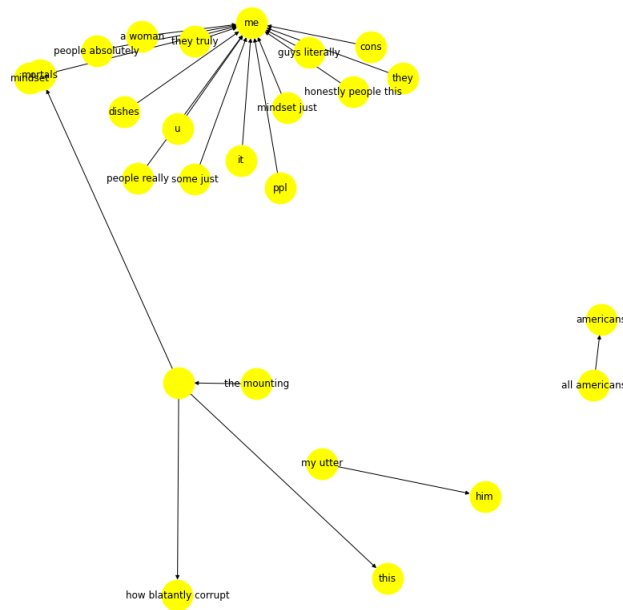


Figure 4.6: Example of ‘disgust’ relations in the knowledge graph.

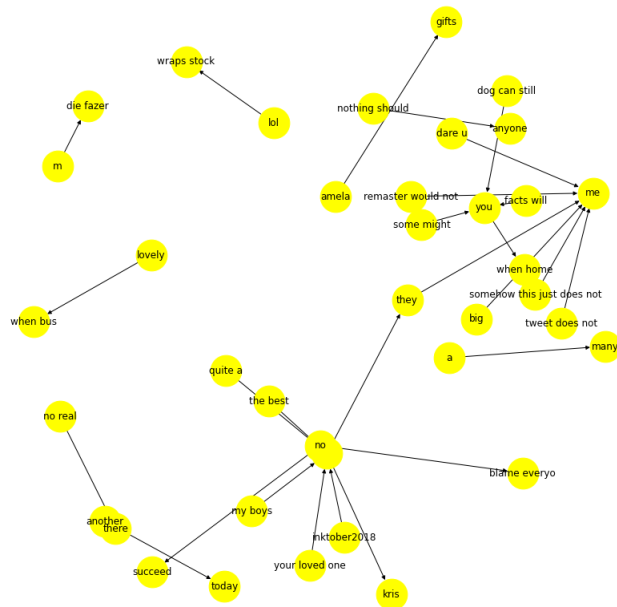


Figure 4.7: Example of ‘surprise’ relations in the knowledge graph.



Figure 4.8: Example of 'fear' relations in the knowledge graph.

#### 4.1.4 Discussion

In the following section, several challenges are outlined when it comes to creating a new KG from tweets, which affect different aspects of the process. Furthermore, it describes which common use cases there could be for this knowledge graph.

##### Challenges

A common downside of many KGs is that they only contain the knowledge that was explicitly referenced in the text and therefore fail to capture anything beyond that (Bosselut et al. 2019). This also applies to the emotions represented in this KG because, by default, these are limited to Ekman’s six basic emotions. It also affects the way that feelings or polarities are not considered, where it can be seen in Figure 4.9 that a triple extracted from this tweet still carries affective meaning but would not be captured as an emotion in this approach.

**Never** – NEG **speak** – ROOT **from** – PREP **a** – DET  
**place** – DOBJ **of** – PREP **hate** – DOBJ

Figure 4.9: Example of a feeling conveyed in tweet that is not captured

There are many challenges when working with this type of noisy data, which means that many existing NLP toolkits would fall short. One such example is using a NER toolkit that is commonly used in KG creation (Mesquita et al. 2019) to detect entities. In our case, we tried existing methodologies; however, in most cases, this was unsuccessful, and empty values were returned. This is attributed to two main issues: (i) many existing toolkits are trained for texts that are mostly procedural, e.g., news articles or Wikipedia entries, which means that models presented with a different language structure would not work well; and (ii) much of the language used in tweets is non-standard, where previous approaches have often relied on capitalisation to detect entities, and this is often not done in informal language (Mayhew et al. 2019).

### Application and future directions

There are many use cases for an emotion KB in a range of different tasks, which range from commercial and health care applications to robotics. This includes tasks such as classifying the affective state of text messages or helping generate language that contains emotive language (Strapparava et al. 2004). For example, tracking the sentiment people express towards a range of different topics (e.g., products or politics) or creating dialogue systems that can give an adequate response to emotions expressed by its users (Mohammad & Turney 2013). Another use case could be in health care to analyse suicide notes (Schoene, Lacey, Turner & Dethlefs 2019) or extract opinions from patient feedback forms.

One future avenue of research is to investigate the distribution of emojis in this KG, as qualitative visualisations have shown that emojis are in similar emotion categories as their own affective meaning. For example, in the *happy* emotion category, there tend to be emojis that have a positive connotation, e.g., *'face\_blowing\_a\_kiss'*. Another opportunity lies in potentially linking this work to a larger KB, such as DBpedia to increase its commonsense knowledge with emotional concepts.

## 4.2 Learning Embedding Representations

The key challenge in learning large-scale embedding representations that are sensitive towards emotion or sentiment lies in being able to learn word vector representations that not only reflect context but also ensure that emotion words of polar opposite meanings do not occupy the same vector space (Zhang, Wu & Dou 2019). Previous research has often focused on fine-tuning existing LMs, where most of these models have only been tested on one type of SA task such as polarity detection using the IMDB dataset (Lin et al. 2011). Therefore, it is important to investigate not only fine-tuning approaches for existing LM methods, but also

explore how new methods can be beneficial to a range of different SA tasks (e.g., fine-grained emotion detection).

The following section will outline a number of experiments to show how the knowledge graph RELATE (created in section 4.1) can be used in the context of learning embedding representations. Then RELATE will not only be benchmarked against various other embedding representation learning methods but also compared to several different other resources. Finally, the embedding representations will be visualised in order to see how emotion keywords are represented in the different embedding spaces.

The learning model used in these experiments to generate embedding representations is a Graph Convolutional Neural Network as first outlined in Chapter 2.3, section 2.3.1. The GCN that is used to generate embedding representations is a GCN as proposed by Yao et al. (2019a), where both TF-IDF and PMI are used to calculate the edges between nodes in the input KG. More specifically, emotion knowledge is implicitly included in the model by using emotion keywords or triples as nodes (see Figure 4.10 for an overview). The embedding representations are learned with the following experimental setup for the GCN learning rate = 0.001, hidden units = 200, dropout = 0.5. Figure 4.10 gives an overview of the process of learning the embedding representations. Blue circles show whole triples and green circles show each individual part of a triple. Green lines indicate word-to-word edges (using PMI) and blue lines show word-to-document edges (using TF-IDF). The first image in the blue box shows the Word-Document graph generated, the second image shows the learned representations for both words and documents and the third image shows the output of an  $m \times n$  dimensional matrix for each triple.

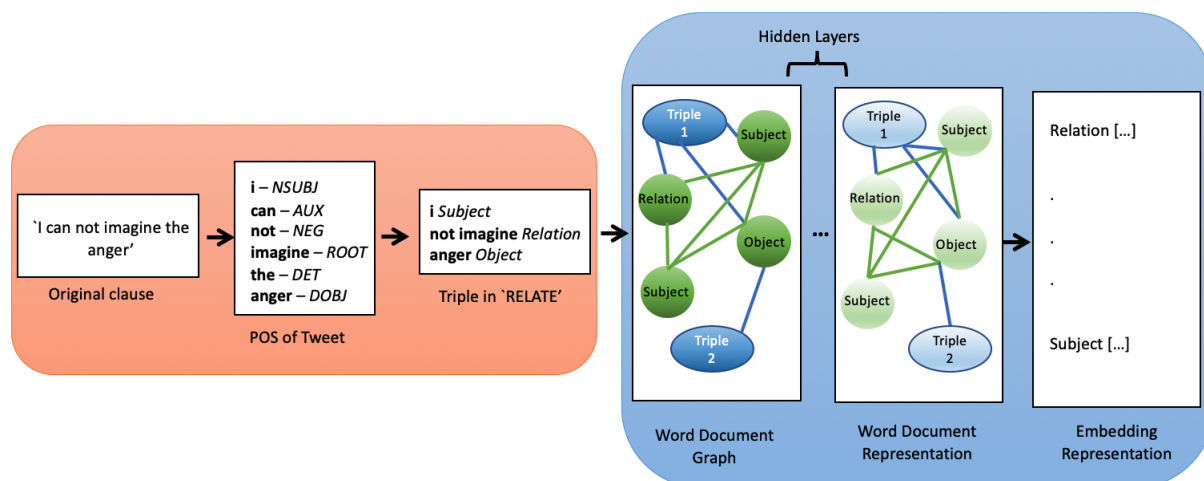


Figure 4.10: Example of a clause going through preprocessing of ‘RELATE’ (orange box to the left) and then input into the GCN (blue box to the right - graphic adapted from Yao et al. (2019a)).

### 4.2.1 Experimental setup

A performance baseline is established for the task of classifying the IEST dataset into six different emotion categories, where a simple two-layer LSTM is used with a simple embedding layer. The IEST data is split into 80% training, 10% validation and 10% test data, where the input into each network is a set of 200-dimensional embedding representations unless specified otherwise. All LSTMs share the same hyper-parameters, where the learning rate = 0.001, batch size = 128, dropout = 0.5 and the hidden size is 40 units. All experiments were conducted using Tensorflow (Abadi et al. 2016), and early stopping was used to prevent overfitting.

The embeddings learned by the GCN are then compared against two static word embedding methods: GloVe (Pennington et al. 2014) and Word2Vec (Mikolov, Chen, Corrado & Dean 2013) and three state-of-the-art LMs, including BERT (Devlin et al. 2018) and ELMO (Peters et al. 2019). Each competing embedding method was then used as an embedding layer to the plain vanilla LSTM to see if and how much they contribute to the successful classification of fine-grained emotions in tweets. Whilst

fine-tuning was used in both BERT and ELMO; it was decided not to train new LMs based on those architectures using the data provided in section 4.2.1, because of the nature of these learning models. More specifically, these LMs were developed using large amounts of training data and resources that are nearly impossible to match in an academic setting.

## Resources

In order to generate new embedding representations using a GCN, four different datasets are used (see Table 4.4). This is done to compare how effective embedding representations based on the knowledge graph ‘RELATE’ are.

Resource	Description	Size
RELATE	Triples are used as input into the GCN to generate 200-dimensional embeddings for each triple.	42,646
EEK	The full EEK dataset was used as input to the GCN	390,000
EEK - small	The resources was randomly generated to see how effective RELATE would be when the inputs are plain tweets and the resource size is the same.	42,646
SEMI	Not all triples in RELATE have labels, therefore a GCN was used to automatically label all missing triples with emotion labels so that all knowledge graph triples could be used as input	490,299
Word2Vec	Tweets collected and pre-trained by Godin et al. (2015)	400 million
GloVe	Tweets collected and pre-trained by Pennington et al. (2014)	2billion
ELMO	This language model was taken from Tensorflow-hub <a href="https://tfhub.dev/google/elmo/2">https://tfhub.dev/google/elmo/2</a>	1Billion words
BERT	This language model was taken from Tensorflow-hub <a href="https://tfhub.dev/google/bert_uncased_L-12_H-768_A-12/1">https://tfhub.dev/google/bert_uncased_L-12_H-768_A-12/1</a>	3.3 billion

Table 4.4: Overview of the different resources that were used as input to the GCN



### 4.2.2 Results and Evaluation

Table 4.5 show the results for all experiments that were outlined in the previous section, where the top half of the table shows experiments using the GCN and the different resources, and the bottom half shows the results of the already existing and established embedding models. Firstly, it can be seen that all embedding representations learned by the GCN, except EEK-small, outperform the baseline. Furthermore, it is shown that the best performing embedding representations generated by the GCN are based on the SEMI and RELATE resource. This is especially interesting, because of the size difference and nature of the two resources, which means that the same results can be achieved regardless of whether a GCN is trained on either a larger resource or a linguistically inspired knowledge graph. Furthermore, it can be seen that the best results were achieved using GloVe, which was pre-trained on 2billion tweets, and the BERT model produced the lowest results. This is particularly surprising, given the amount of data and methodology that is used, and the groundbreaking results that were achieved in other NLP tasks (Gao et al. 2019, Wu et al. 2019). Finally, it can also be seen that ELMO performs better than BERT on this task. This is surprising, especially because of the pre-training size of BERT (3.3 billion) compared to ELMO (1 billion).

Model	Precision	Recall	F-1 Score
LSTM PLAIN	0.52	0.52	0.52
EEK - small	0.47	0.46	0.46
EEK	0.57	0.56	0.56
RELATE	0.58	0.57	0.57
SEMI	0.57	0.57	0.57
Word2Vec	0.52	0.52	0.52
GloVe	0.60	0.59	0.59
ELMO	0.58	0.58	0.58
BERT	0.19	0.18	0.18

Table 4.5: Experiment results for the GCN embeddings and existing language models using f-1 scores.

Table 4.6 shows the training setup for each model, including the approximate

training time (in hours and minutes) and the model parameters for each embedding layer. For all experiments two Tesla P100 GPUs were used. From this it can be inferred that whilst GloVe achieves the best results, ‘RELATE’ is most efficient to train and can therefore be seen as a more lightweight embedding model. This is also in stark contrast to the time taken by ELMO and BERT to train for 100 epochs. Therefore evidence suggests that (i) it is important which type of data is used to train the embedding model on and (ii) the size of the data is important when not using additional linguistic rules.

Resource	Comments	Training Time	Embedding Parameters
RELATE		02:46	11,035,400
EEK		04:31	11,035,400
EEK - small		00:36	683,000
SEMI		03:43	11,035,400
Word2Vec		02:43	22,070,800
GloVe		03:11	11,035,400
ELMO		11:38	262,400
BERT	fine_tuning layers = 40	15:18	110104890

Table 4.6: Overview of the different settings, training times (hours :minutes) and model parameters for each embedding layer.

**Visualisation of the embedding space** In order to visualise the embedding representation, Tensorboard Embedding projector (Tensorflow 2020) was used. This was done for both the best performing GCN representations using RELATE and the representations learned by GloVe. Figures 4.11 - 4.22 show visualisations for the emotion keywords ‘joy’, ‘sad’, ‘surprise’, ‘disgust’, ‘anger’ and ‘fear’ in both RELATE and GloVe. The 100 closest words (measured in cosine similarity) are highlighted around the keyword. The x and y - axis are fixed to the left and right for the words ‘good’ and ‘bad’ respectively in order to identify bias in the embedding representation.





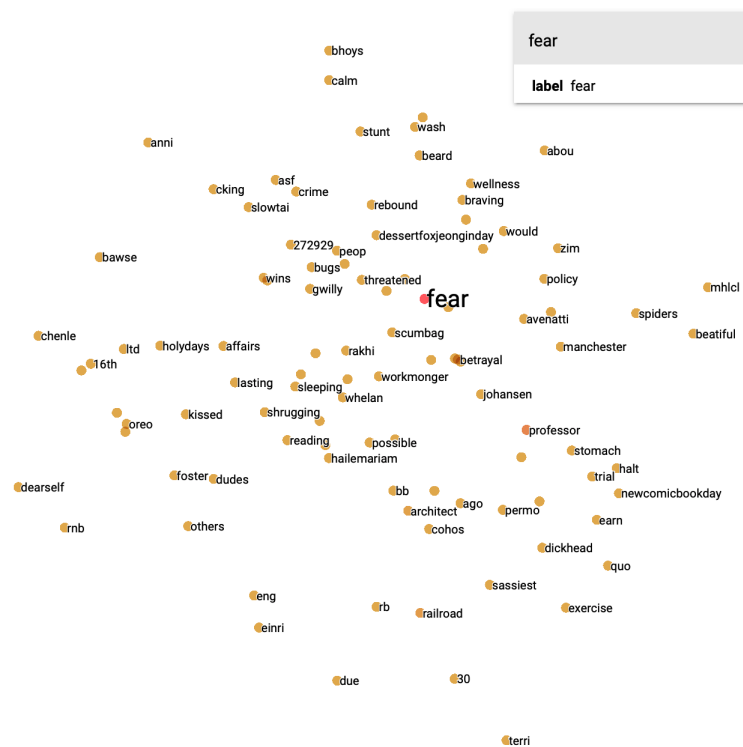


Figure 4.13: Visualisation of the emotion keyword ‘fear’ in the embedding representation of the GCN using RELATE.



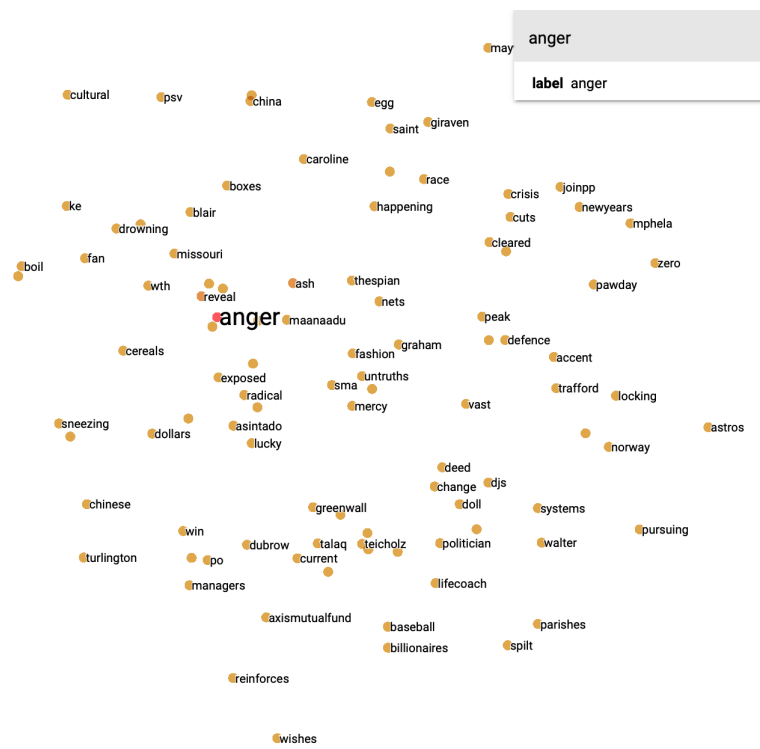


Figure 4.15: Visualisation of the emotion keyword ‘anger’ in the embedding representation of the GCN using RELATE.

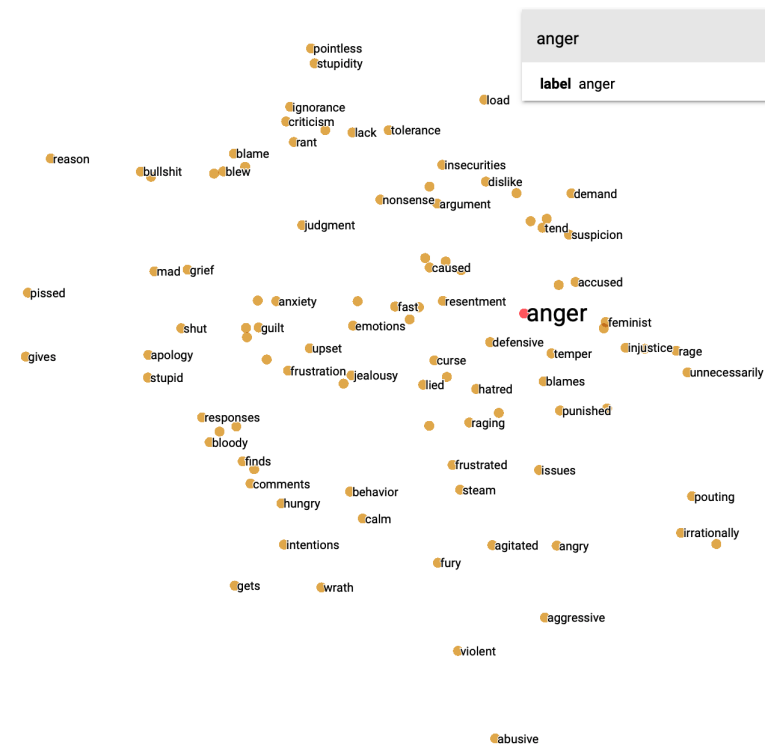


Figure 4.16: Visualisation of the emotion keyword ‘anger’ in the embedding representation of GloVe.





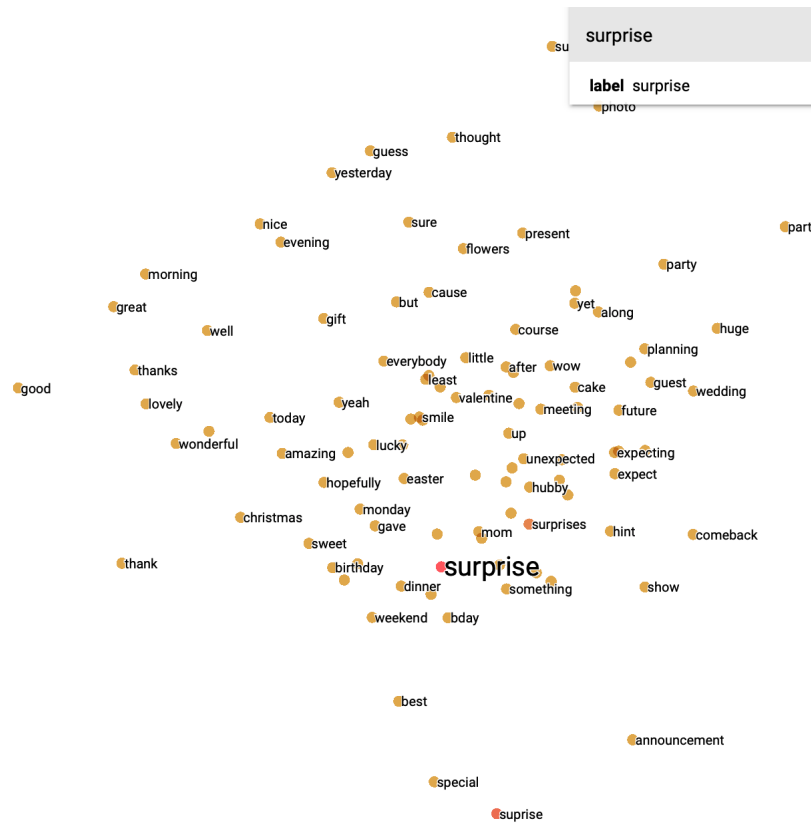


Figure 4.18: Visualisation of the emotion keyword ‘surprise’ in the embedding representation of GloVe.

Figure 4.19 shows that the emotion keyword is positioned more in the middle, whilst Figure 4.20 shows that it is more biased towards the word ‘bad’. Furthermore, it can be seen in RELATE that it has a high cosine similarity for words ‘happy’ and ‘pregnancy’. In GloVe terms such as ‘idiocy’ and ‘pettiness’ have higher scores and are therefore closer positioned to the emotion keyword. Additionally, it is interesting to observe how the emotion keyword ‘happy’ (as part of the ‘joy’ category) scores high, but is biased towards the concept of ‘good’.

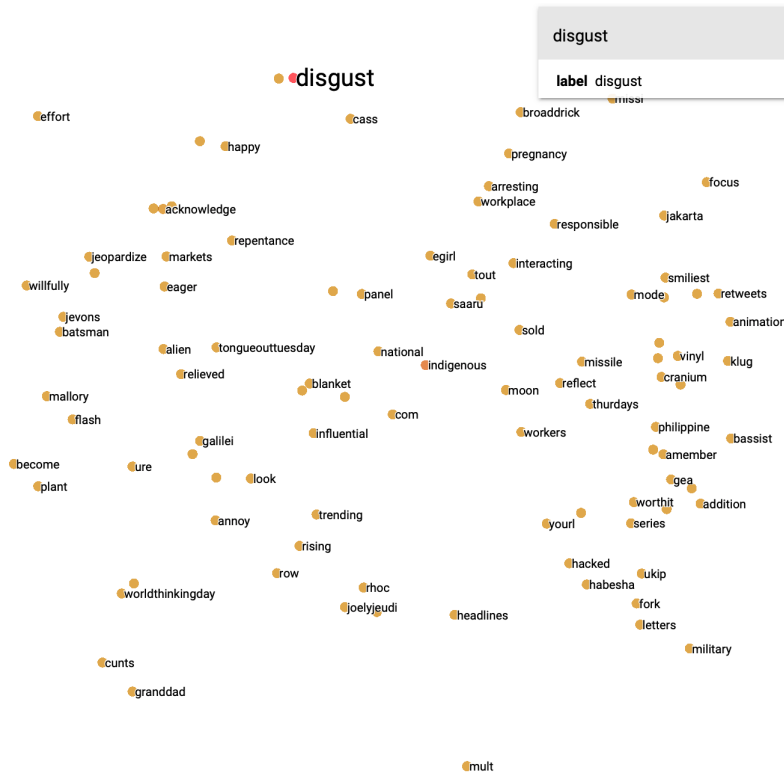


Figure 4.19: Visualisation of the emotion keyword ‘disgust’ in the embedding representation of the GCN using RELATE.

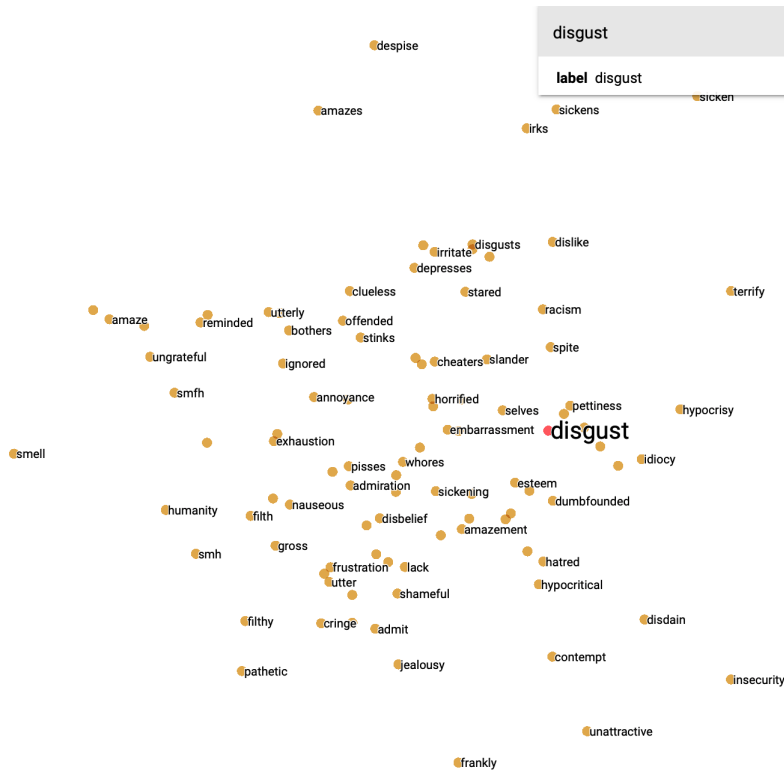


Figure 4.20: Visualisation of the emotion keyword ‘disgust’ in the embedding representation of GloVe.





---

related to ongoing events at the time of collection and incorporates non-standard language (e.g.: hashtags and colloquial language). On the other hand, RELATE exposes inherent bias in the embedding representation, which if used in real world task could lead to further increasing inequality and discrimination. Therefore, future work should first and foremost look at how to circumvent this bias in embedding representation to create fairer and ethical representations that are not potentially harmful. Furthermore, it should look at pretraining RELATE on a larger dataset to see if these findings change. Also, it should be considered to use not just cosine similarity, but also Euclidean distance to show how words are closely related to each other.

## 4.3 Conclusion

In this chapter, first the direct KG ‘RELATE’ was introduced and a detailed description of the data used and its creation was provided in section 4.1. Furthermore, RELATE was evaluated by looking at both the emotion keyword distributions and visualisations of the KG, where it was found that preprocessed emojis were closely positioned to their related emotion keywords. Finally, it was outlined what common challenges there are for creating new KGs (e.g.: NER) and what kind of future work should be considered (e.g.: linking this KG to larger commonsense KGs). In section 4.2, RELATE was evaluated on the downstream task of learning sentiment specific embedding representations using a Graph Convolutional Neural Network. Then it was shown that the embeddings generated with the GCN produce similar classification results compared to other large-scale LMs, such as BERT and ELMO in the task of fine-grained emotion classification in Twitter. However, one of its advantages is that it is more efficient and requires less resources compared to models such as BERT. Furthermore, it was established that any new LMs need to be tested on a variety of real world tasks, before they are applied to a particular type of task such as Sentiment Analysis. Although RELATE proves to be more efficient there are some important considerations to be taken into account before deploying such as model in real world scenarios, which includes the bias that is shown in the representations. Finally, the embedding representations were qualitatively evaluated through visualisations. For each emotion keyword, the two most successful learned embedding representations visualisations were created using Tensorflow’s embedding projector. There it was found that, GloVe produces more standard representations for words overall, whilst RELATE also includes hashtags and colloquial language. Furthermore, it was found that RELATE reflects emotions/opinions and concepts linked to events that happened at the time of data collections, whereas GloVe does not. Therefore, future

work needs to not only look at further methods of evaluating this knowledge graph, but also at ways to eliminate bias in embedding representations.

# Chapter 5

## AI for Social Good: Suicide Note

### Classification

Both machine and deep learning techniques have been predominantly used for commercial purposes; there has also been an increased awareness of how AI approaches could contribute to solving some of the biggest social problems humans face worldwide (ITU 2019). This awareness has led to the creation of new workshops and conferences that fall under the umbrella of "*AI for Social Good*", where machine learning researchers connect with NGOs (Non-Governmental Organisations), charities and other problem owners to create practical solutions. These problems and challenges are usually closely linked to accelerating progress towards the UN Sustainable Development Goals (SDG) produced by the United Nations (UN) (WHO 2019). These goals include, but are not limited to, protecting democracy, education, social welfare and justice as well as health care and environmental sustainability.

Especially within the SDG for health care, there is an increased focus on mental health. In a recent report, the World Health Organisation (WHO 2019) outlines that suicide is the second leading cause of death for people aged 15-29 worldwide. Reducing the rate of suicide worldwide has, therefore, been listed as one of the



objectives of the Sustainable Development Goals for health care. It is estimated that around 25-30% of people who die by suicide leave behind a suicide note; however, this figure can be as high as 50% depending on cultural or ethnic differences in demographics (Shioiri et al. 2005). Desmet & Hoste (2013) have found that there is an increasing trend amongst younger people to publish their suicide notes or express their suicidal feelings online. Furthermore, psychological studies have shown that our state of mind can manifest itself in the linguistic features we use to communicate (Osgood & Walker 1959, Cummings & Renshaw 1979). At the same time, the use of social media platforms, such as blogging websites has become part of everyday life, and there is increasing evidence emerging that social media can influence both suicide-related behaviour and other mental health conditions. Whilst there are efforts to tackle suicide and other mental health conditions online by social media platforms such as Facebook (Facebook 2019), there are still concerns that there is not enough support and protection, especially for younger users (BBC 2019).

Taking these trends into account and with the unprecedented availability of textual data from social media platforms, researchers now have the opportunity to analyse such data and use their findings in several different application areas. This has led to a notable increase in research of suicidal and depressed language usage (Coppersmith et al. 2015, Pestian et al. 2012) and subsequently triggered the development of new healthcare applications and methodologies that aid detection of concerning posts on social media platforms (Calvo et al. 2017). Traditionally, work on suicide notes has focused on distinguishing genuine from forged suicide notes in the field of forensic linguistics, where the findings were used as additional evidence in legal proceedings (Coulthard et al. 2016). However, in recent years and with the advances in machine and deep learning, there has been an increasing amount of research conducted to identify suicidal ideation or suicide notes in online settings, such as social media platforms (O’dea et al. 2017, Shahreen et al. 2018). Finally, there are a number of ethical considerations to be taken into account when working with mental health

data in a computational setting. Work by Chancellor & De Choudhury (2020) shows that there is no standardised process for either collecting, annotation or working with this type of data. It has to be noted that often the analysed textual data can only be seen as a proxy of the actual state of mind of the writer. Furthermore, it has to be considered that this work deals with a limited amount of textual data and that the results are not representative for each country or person that leaves a note behind.

In this chapter, there will firstly be an exploration of existing research and literature in the field of suicide note detection. Then there will be an analysis of the linguistic features for the different datasets used. Next, there will be a series of four experiments using different kinds of datasets and neural networks. The first experiment series looks at establishing baselines on existing related work using neural network-based approaches. The second experiment tests the hypothesis of whether a neural network can detect the linguistic differences in people who have died by suicide or are forced to die. In experiment three, it is investigated how well suicide notes can be distinguished from neutral texts using equally balanced datasets. The final experiment looks at how well suicide notes can be classified when there is a vast amount of neutral text data, which makes the task more applicable to real-world scenarios.

## **5.1 Related Work**

Over the years there has been much research conducted into the accurate classification of suicide notes or detection of suicidal ideation online (O’dea et al. 2017, Shahreen et al. 2018), where researchers use several different methodologies including but not limited to traditional machine learning (Liakata et al. 2012), deep learning (Coppersmith et al. 2018) and sentiment analysis (Wang et al. 2012). Such

research has been conducted in a range of different disciplines like psychology (Gunn & Lester 2015), linguistics (O’dea et al. 2017) or healthcare (Denecke & Deng 2015). Many experiments have also been conducted comparing different types of textual data with suicide notes such as depressed language or blog posts (Schoene, Lacey, Turner & Dethlefs 2019). Overall there has been a growing interest in looking at content created online that may solicit need for help (Jaiswal et al. 2017) or detecting mental health issues (Savova et al. 2016).

This literature review will focus on introducing work looking at the classification of suicide notes and suicidal ideation detection, but also review work in the space of depressed language and last statements due to the nature of the experiments.

**Suicide note classification** The analysis of suicide notes has been used in various academic settings such as psychology or forensic linguistics to either identify the genuineness of a suicide note or to predict the state of mind of a note writer (Coulthard et al. 2016). It has been argued in previous research that our drive or motivation affects how we communicate, and therefore it is believed that our spoken and written language represents those shifting psychological states (Osgood 1960). This argument has been taken further by Cummings & Renshaw (1979), who suggested that there is a shift in one’s linguistic expression due to the aroused cognitive state suicidal individuals’ experience. These findings have led to Desmet & Hoste (2013)’s argument that there is an increased need for ‘automatic procedures that can spot suicidal messages and allow stakeholders to quickly react to online suicidal behaviour or incitement’. Therefore recent research has looked at different aspects of suicide notes to find out what "makes" a suicide note, where identifying linguistic features and patterns, affective states or specific emotions, as well as dominant topics, have been used in different analysis and experiments. Ji et al. (2019) provide an overview of applications, methods and domains in suicide note research.

One of the settings in which the validation of a suicide note is important is in court cases or hearings where expert evidence is given by professionals such as forensic linguists to verify the author of the note or its genuineness (Coulthard et al. 2016). Another field where the analysis of suicide notes is crucial is psychology, where one of the most commonly cited studies has been conducted by Shneidman & Farberow (1956). In their study, they collected a corpus of 33 genuine suicide notes and another set of 33 suicide notes that were forged. Their analysis showed that there was a clear difference in language used (e.g.: the use of endearments in suicide notes), which made the genuine notes distinctive when compared to the forged notes. This study has been used as a foundation for many other studies afterwards (Shapero 2011) and researchers such as Osgood & Walker (1959) have compared this set of suicide notes with a set of normal letters to friends. Whilst especially early work in linguistics and psychology has mainly focused on the distinguishing factors of linguistics and topics (Ren et al. 2015), the availability of such data to researchers from other disciplines has opened up opportunities to use traditional machine learning and feature engineering for classifying suicide notes.

Jones & Bennell (2007) have used a supervised classification model and a set of linguistic features to distinguish genuine from forged suicide notes, achieving an accuracy of 82%. Studies using traditional machine learning have been taken further recently by Pestian et al. (2010) who also used a set of suicide notes and hypothesised that when applying the set to a machine learning algorithm it would outperform mental health professionals in classifying suicide notes correctly. This proved to be true, where the machine learning algorithm achieved a 78% and trained health professionals achieved 63% accuracy. Detecting affective states or emotions in such data has also grown in popularity. Particularly the work of Pestian et al. (2012) has been influential in the field, and in their study, they have found that there are fifteen different emotional concepts which prove to be significant in identifying genuine suicide notes. These fifteen sentiment features have also been used by

Yang et al. (2012) in the i2b2/VA/ Cincinnati Medical Natural Language Processing Challenge. The challenge aimed to develop a model which could automatically identify emotions on sentence-level of a suicide note. The hybrid model developed by Yang et al. (2012) achieved an accuracy of 61.39% in detecting emotions using various techniques such as machine learning-based emotion classification. Yang et al. (2012) argue that one of the key factors for successful identification of emotions is to split the 15 pre-specified emotions into three different classes (positive, negative and neutral). Work by Burnap et al. (2015) has looked at identifying suicidal ideation on Twitter by using lexical, structural and sentiment features, using traditional machine learning algorithm and achieved an F-measure of 0.728. Schoene & Dethlefs (2016) have focused on combining both sentiment and linguistic features which led to achieving a test accuracy of 86.6%. Chen, Aldayel, Bogoychev & Gong (2019) have used four different feature groups, including sentiment, to assess suicide risk using a hybrid model. Work by Cherry et al. (2012) looked at the role of emotions in suicide notes.

**Suicide Ideation Classification** Recent years have seen an increase in the analysis of suicidal ideation on social media platforms, such as Twitter. Shahreen et al. (2018) searched the Twitter API for specific keywords and analysed the data using both traditional machine learning techniques as well as neural networks, achieving an accuracy of 97.6% using neural networks. Research conducted by Burnap et al. (2017) has developed a classifier to distinguish suicide-related themes such as the reports of suicides and casual references to suicide. The increased use of deep learning in other areas of Natural Language Processing (Cambria & White 2014) has also led to more studies using Recurrent Neural Networks (RNN) or Convolutional Neural Networks (CNN) to detect suicide notes or suicidal ideation (Zirikly et al. 2019). Work by Sawhney et al. (2018) used multiple neural network architectures to detect suicidal ideation. Research by Benton et al. (2017) uses

multi-task learning to estimate the risk of suicide using multiple public datasets from various shared tasks. Work by Ji, Pan, Li, Cambria, Long & Huang (2020) reviews the most recent literature and progress in the space of identifying suicidal ideation.

Suicide note research has not only focused on the sentiment conveyed in notes but also on linguistic (Osgood & Walker 1959) and content (Handelman & Lester 2007) features. Research conducted by Jones & Bennell (2007) used Receiver Operating Characteristic (ROC) Analysis to distinguish genuine and forged suicide notes from each other, yielding an average accuracy of 0.82 AUC. Other work conducted by Schoene & Dethlefs (2016) has found that using a combination of both linguistic and sentiment features achieves an accuracy of 86.61% using a logistic model tree (LMT).

**Depression notes** Work on identifying depression and other mental health conditions have become more prevalent over recent years, where a shared task was dedicated to distinguishing depression and PTSD (Post Traumatic Stress Disorder) on Twitter using machine learning (Coppersmith et al. 2015). Morales et al. (2017) have argued that changes in the cognition of people with depression can lead to different language usage, which manifests itself in the use of specific linguistic features. Research conducted by Resnik et al. (2015) also used linguistic signals to detect depression with different topic modelling techniques. Work by Rude et al. (2004) used the Linguistic Inquiry and Word Count (LIWC) software to analyse written documents by students who have experienced depression, currently depressed students as well as students who never had experienced depression, where it was found that individuals who have experienced depression used more first-person singular pronouns and negative emotion words. Nguyen et al. (2014) used LIWC to detect differences in language in online depression communities, where it was found that negative emotion words are good predictors of depressed text compared

to control groups using a Lasso Model (Tibshirani 1996). Research conducted by Morales & Levitan (2016) showed that using LIWC to identify sadness and fatigue helped to classify depression accurately. Zhao et al. (2018) use Convolutional Neural Networks to model the relationship between depression and people who attempt suicide. Some work has focused on detecting mental health signals related to other conditions such as bipolar disorder, major depressive disorder, post-traumatic stress disorder and seasonal affective disorder (Coppersmith et al. 2014). In their work Mowery et al. (2016), have looked extensively at which features are relevant when classifying depression in tweets. This also included the use of features discovered by LIWC as well as other features such as sentiment and emoticons.

**Last statements** The analysis and classification of suicide notes, depression notes and last statements have traditionally been conducted separately. The main purpose of analysing last statements has been to identify psychological factors or key themes (Schuck & Ward 2008). Most work in the analysis of last statements of death row inmates has been conducted using data from The Texas Department of Criminal Justice, made available on their website (Texas Department of Criminal Justices 2019). Recent work conducted by Foley & Kelly (2018) has primarily focused on the analysis of psychological factors, where it was found that specifically themes of *'love'* and *'spirituality'* were constant whilst requests for forgiveness declined over time. Kelly & Foley (2017) have also identified that mental health conditions often occur in death row inmates with one of the most common conditions being depression. Research conducted by Heflick (2005) studied last statements using qualitative methods and have found that often afterlife belief and claims on innocence are common themes in these notes. Eaton & Theuer (2009) studied the level of apology and remorse in last statements qualitatively, whilst also using logistic regression to predict the presence of apologies achieving an accuracy of 92.7%. Lester & Gunn III (2013) used LIWC to analyse last statements, where they have found nine main

themes, including affective and emotional processes. Also, Foley & Kelly (2018) found in qualitative analysis that the most common themes in last statements were love (78%), spirituality (58%), regret (35%) and apology (35%).

**Social Media blogs** For the purpose of the experiments conducted in section 5.6, a brief overview of the work on social media blogs is given. Work on classifying blogs from social media platforms has focused on predicting sentiment or emotions (Binali et al. 2010) or characteristics of the author of a blog, such as age (Rosenthal & McKeown 2011) or gender (Bartle & Zheng 2015). Other work has focused on modelling ideologies in blogs using topic modelling techniques (Lin et al. 2008).

## 5.2 Data

The datasets used for the experiments in this chapter have been expanded over time. Therefore this section provides an overview of the different datasets as well as where and how they have been collected. All corpora have been anonymised in order to protect the authors' identity and those mentioned in their communication, which includes any places, names or references to identifying information. The examples of notes below have been chosen for their brevity; many of the notes in the corpus are of greater length.

**Data collection for Genuine Suicide Notes** Genuine suicide notes provide a unique insight into the mindset of a person who has committed suicide (Gregory 1999). Therefore we have chosen only to use genuine suicide notes in our experiments and made a conscious decision not to use other datasets such as Twitter suicide datasets (Burnap et al. 2015). The main reason for this being that these tweets have mainly been collected using specific keywords such as '*suicide*' to accumulate



Dataset	Website	Title	Accessed	URL
GSN3	TMZ	AARON HERNANDEZ SUICIDE NOTE TO FIANCEE RELEASED 'You're Rich'	2017	<a href="https://tinyurl.com/n4dl5p4">https://tinyurl.com/n4dl5p4</a>
	The Sun	Suicide Note of Woman after mothers death	2017	<a href="https://tinyurl.com/yv8dry7">https://tinyurl.com/yv8dry7</a>
	Says	Student In Penang Left A Suicide Note On Facebook Before Jumping To His Death	2017	<a href="https://tinyurl.com/y3lnd7j2">https://tinyurl.com/y3lnd7j2</a>
	NDTV	'Suicide by cop': Marathi Filmmaker Atul B Tapkir Found Dead	2017	<a href="https://tinyurl.com/lh4sexb">https://tinyurl.com/lh4sexb</a>
	Daily Mail Online	'Suicide by cop': Boy, 15, who was shot dead after he called 911 on himself and pointed a BB gun at the responding officers left behind a suicide note	2017	<a href="https://tinyurl.com/y6a9c2tk">https://tinyurl.com/y6a9c2tk</a>
	LadBible	Young Girl With Anorexia Penned Several 'Goodbye' Notes Before Her Suicide	2017	<a href="https://tinyurl.com/yvgh7h3">https://tinyurl.com/yvgh7h3</a>
	Thinking Humanity	Teen Who Was Bullied And Gang Raped Left This Heartbreaking Suicide Note	2017	<a href="https://tinyurl.com/y5dq6sdt">https://tinyurl.com/y5dq6sdt</a>

Table 5.1: Overview over the different outline resources for the GSN3 dataset

the data and there is no human verification that the person who wrote this tweet is indeed suicidal or has passed away. Due to the sparsity of genuine suicide notes that are publicly available, we have added new genuine suicide notes to this corpus over time. Below the different datasets used in the subsequent experiments are outlined (see Table 5.2) and Table 5.1 shows some references for additional source of suicide notes.

Figures 5.1 - 5.5 show examples of notes and posts for each dataset category.

when you receive this i will be dead please feed my  
pets cats as soon as possible put them to sleep at the  
vets not the pet shelter the vets will charge about  
2000 each you will find enough money in the house to  
pay for it you may also have my 2 rings my earrings the  
2000 trust you william can decide about me at peace  
at last elinor i did not move my body will be at the  
address i am not moving sorry

Figure 5.1: Example of a Suicide Note.

My boyfriend and I are in love with each other and its  
such a wonderful feeling. He tole me that he's in love  
with me and it was a wonderful overwhelming feeling  
that it made me cry tears of joy. I love him so much.

Figure 5.3: Example of a Love Note.

tonight is one of those nights when i want to cut  
my wrist and just end my life then again thats  
nearly every night for me

Figure 5.2: Example of a Depressed Note.

No sir. Warden, Since I dont have nothing to say, you  
can go ahead and send me to my Heavenly Father.

Figure 5.4: Example of a Last Statement.

“ so today is my moms 49th birthday we  
went out for dinner to timbers it was really  
good its hard to believe its already her  
birthday it seems like this summer is going  
by so fast sometimes well a lot of the time it  
scares me how fast time goes i guess you just  
have to live every minute to the fullest which  
is hard sometimes oh well thats life im not in  
the mood to dwell on serious topics like this  
right now so ill be moving on to something a  
bit happier im going to a cottage tomorrow  
with a whole bunch of my friends ...”

Figure 5.5: Example of an excerpt of a neutral post.

Furthermore, it has to be noted that LH posts were written by people who are happily in love and DL1 post were written by people identifying themselves as depressed and lonely respectively. LS contains records of prisoners who have received the death penalty between 1982 and 2017 in Texas, U.S.A. Datasets DL2, DL3, NEU1 and NEU2 were randomly selected posts from the overall dataset. Finally, the dataset size for NEU2 was chosen empirically to ensure that the overall amount of GSN notes is below 5% to make the task more applicable to the real world.

<b>Corpus Name</b>	<b>Resource</b>	<b>Experiment</b>	<b>Size</b>
GSN1 (Genuine Suicide notes)	This dataset was taken from Schoene & Dethlefs (2016)	1	142
GSN2 (Genuine Suicide notes)	GSN1 was extended by using notes introduced by The Kernel (2013) and Tumbler (2013)	2	161
GSN3 (Genuine Suicide notes)	GSN2 was extended by using public resources (see Appendix ??)	3 and 4	211
DL1 (Depression notes)	Data collected by Schoene & Dethlefs (2016)	1 and 2	142
DL2 (Depression notes)	Reddit data collected by Pirina & Çöltekin (2018)	3	211
DL3 (Depression posts)	Reddit data collected by Pirina & Çöltekin (2018)	4	1293
LH (Love/Happiness notes)	Data collected by Schoene & Dethlefs (2016)	1	142
LS (Last Statements)	This dataset has been made available by the Texas Department of Criminal Justices (2019)	2	431
NEU1 (Neutral posts)	Data made available by Schler et al. (2006)	3	211
NEU2 (Neutral posts)	Data made available by Schler et al. (2006)	4	3500

Table 5.2: Overview over the different datasets that were used in the following experiments.

## 5.3 Linguistic Analysis

To gain more insight into the content of the datasets, we performed a linguistic analysis to show differences in the structure and contents of the different datasets. For this study the LIWC software (Tausczik & Pennebaker 2010) is used, which has been developed to analyse textual data for psychological meaning in words. The average of all results across each dataset is reported. LIWC has been used in previous research to annotate datasets for suicide risks in addition to experts to determine linguistic profiles of suicide-related Twitter posts (Just et al. 2017). Other work by Rude et al. (2004) used LIWC to analyse written documents by students who have experienced depression, currently depressed students as well as students who never have experienced depression.

In the following section, there will be an analysis of the different datasets introduced in section 5.2 using LIWC. Furthermore, there will be an analysis of how statistically significant these features are for experiments 3 and 4, where it is tested if suicide notes can be distinguished from neutral text.

### 5.3.1 Analysis for Experiment 1

In this section the datasets GSN1, DL1 and LH were used for analysis, the results of this analysis are used as part of Experiment 1. Experiment 1 looks at establishing baselines on existing related work using various neural networks. Furthermore the Natural Language Tool Kit (NLTK) was used as part of this analysis (Bird & Loper 2004).

**Average note length** In Table 5.3 we can see the average number of grammatical and content features for each post or note for each dataset. The first feature to be analysed will be the length of each note collected as it is argued by Gregory

Corpora	GSN1	LH	DL1
Word count per note	137.77	65.12	112.37
Average Note Length	144.60	70.78	120.85
Average Sentence Length	15.0	12.0	15.0
Allness Terms	123	85	113
Cognitive Processes	12.36	15.05	16.95
Characters	773.97	340.35	614.63
Nouns	27.29	10.00	16.25
Verbs	25.83	12.96	23.28
Adjectives	7.23	4.11	6.56
Adverbs	8.01	4.88	9.86
Pronouns	20.13	10.57	15.80
Lexical Diversity	6.56	6.15	7.13

Table 5.3: Linguistic Features across all three corpora as provided by LIWC (Pennebaker et al. 2014).

(1999) that suicide notes are greater in length since the suicidal individual wants to convey as much information as possible. This is due to the note writer's feeling that they will not have time to convey this information at a later point (Gregory 1999). Table 5.3 also shows the overall length of each corpus which is then divided by the number of notes collected to compute the average length of each note. Another feature demonstrated in Table 5.3 is the length of communication overall in all three corpora. This observation proves to be true for the three corpora analysed as there is a significant difference between the lengths of the three corpora. In addition to that, it can be seen that the corpora differ significantly in the average length per note and the notes of the GSN1 corpus is almost double in length compared to the LH corpus. When looking at the word count for the four different corpora, it becomes apparent that the GSN1 corpus has by far the highest average word count and length per note.

**Average Sentence length** The next feature to be analysed is the average sentence length (ASL). It is argued by Osgood & Walker (1959) that in a genuine suicide note, the ASL is shorter and that there is a higher focus on conveying the most essential facts. Therefore it has been observed that there is usually a small number

of adjectives or adverbs in genuine notes (Osgood & Walker 1959). A similar observation was made by Montgomery (2000), who argues that in a higher cognitive state such as high levels of arousal, a person focuses on providing only essential information. Table 5.3 compares the ASL across all three corpora. The GSN1 corpus and the DL1 corpus have scored the same results in testing for ASL. One explanation for this may be found when looking at Alvarez (2002) who explains that it has been proven for a long time in clinical settings that there is a similarity between the state of mind of a suicidal person and the state of mind of a person who experiences depression. When comparing the LH corpus to the other two corpora, it becomes clear that although the number of tokens in the corpus is smaller, the sentence length is almost as high as in the GSN1 and DL1 corpora. It could be argued that this phenomenon may be due to a higher amount of adjectives used in a sentence, which will be tested at a later point. In addition to this, it has been argued that people who communicate under stress tend to break their communication down into shorter units (Osgood & Walker 1959). The research, however, suggested that there is no significant difference in the overall length per unit when comparing suicide notes to regular letters to friends and simulated suicide notes (Osgood & Walker 1959).

**Nouns** NLTK's Part-of-Speech Tagging was used to identify the linguistic characteristics (Bird 2006). Research has suggested that individuals who commit suicide tend to use more nouns and verbs in their notes (Gregory 1999) and only a small amount of adjectives and adverbs (Osgood & Walker 1959). In addition to that Jones & Bennell (2007) state that a person who is going to commit suicide is under a higher drive and therefore it is more likely for them to reference a high amount of objects (nouns) compared to any other type of word such as verbs. This has also been supported by other studies which found that under higher degrees of stress, the ability to retrieve nouns tends to stay the same, whereas the retrieval of

verbs was less successful (Hayiou-Thomas et al. 2004). When looking at the other three corpora for comparison, it can be seen that this is not true for either the LH or DL1 corpus; however, it is true for the GSN corpus. Therefore it could be argued that the people whose posts have been collected for the DL1 and LH corpus are under a lesser degree of stress. Another reason why this might be the case is that the amount of verbs is only higher than the number of nouns in the LH and DL1 corpus.

**Verbs** Another feature to be included is the number of verbs used in the three corpora. Table 5.3 compares the average number of verbs per note in each corpus. Although Jones & Bennell (2007) argue that there is a high number of verbs in the genuine suicide notes compared to forged notes, it can be seen that the GSN corpus has the lowest amount of verbs present. Therefore it could be argued that this assumption only holds when comparing genuine suicide notes with forged ones, but not when comparing genuine suicide notes to other notes such as depression or love notes. The number of verbs is compared to the combined number of adjectives and adverbs in Jones & Bennell (2007) study's, where the number of verbs is higher in both genuine and forged suicide notes. This also holds for this dataset, where on average there are more verbs in each type of note than adjectives and adverbs combined.

**Adverbs and Adjectives** Adverbs have been defined by Hengeveld et al. (1997) as a 'lexical modifier of a non-nominal head'. It could, therefore, be argued that the number of verbs is already decreasing in people who would like to commit suicide, and they tend not to use many adverbs either. On the other hand, adjectives are used to modify nouns (Baker 2003). The highest amount of adjectives can be found in the GSN1 corpus whereas the lowest is found in the LH corpus. This might be because there is already limited space to convey a message in a single post, and therefore people tend to use less amplifying language.

**Allness Terms** Furthermore, it has been suggested by Osgood (1960) that when a person is highly emotional, they tend to polarise or communicate points in a more extreme manner. They have called these words ‘allness terms’, and therefore the usage of the following terms (Table 5.3) has been explored in the three corpora using NLTK. Examples of these terms include words such as ‘always’, ‘never’ and ‘perfectly’. It can be seen that the GSN1 corpus has the highest amount of Allness terms, whereas the LH corpus has the lowest. These findings concur with the previously mentioned study by Osgood & Walker (1959) and it could be argued that the DL1 corpus is scoring in the middle because people who wrote in it may be in an emotionally similar state, but still not to the same extent as writers of suicide notes.

**Cognitive Processes** Another feature has been proposed by Gregory (1999), who suggested that there is a lower amount of cognitive processes identifiable in a genuine suicide note as the writer has already finished the decision-making process. Moreover, Gregory (1999) argued that this was done by identifying the number of cognitive process words, which would be higher in simulated notes as the writer would still try to justify his or her choice. Therefore LIWC has been used in order to identify the cognitive process in every individual note of each corpus. For comparison purposes, the results of each corpus have been added up and then divided by the number of notes collected to identify the average amount of cognitive activity per note (Table 5.3). It can be seen that the GSN1 corpus has the lowest cognitive activity per note compared to the other two corpora. The highest amount of cognitive processes was found in the DL1 corpus, and it could be argued that this is due to the note writer still being in the process of evaluating a situation. This analysis validates Gregory (1999)’s theory and also suggests that there is a significant difference between notes of the GSN1 and DL1 corpus in terms of cognitive activity. Furthermore, Ioannou & Debowska (2014) found in their study that there is a lower



amount of cognitive processes in suicide notes compared to simulated suicide notes.

**Pronouns** It has been stated by Lester & Leenaars (1988) that there are more references to other people in genuine suicide notes. This was measured by using pronouns to determine whether there are any significant differences. All personal pronouns will be used for comparison with NLTK. The personal pronouns referring to oneself are marked as ‘self’, pronouns referring to other people will be called ‘other’ and those referring to a group in the first person plural will be classed ‘both’. As it can be seen in Table 5.3 there is a higher reference in each corpus to oneself compared to others. Therefore the findings of Lester & Leenaars (1988) do not prove to be true for the data collected in this experiment. However, it could be argued that due to the number of pronouns used in the three corpora that there are differences which may be useful to further investigate to analyse whether the overall usage of pronouns is significant on an individual note level. The average amount of pronouns used per note in each corpus proves to be significantly different as the usage in the DL1 corpus is almost half of the amount used in the GSN1 corpus (Table 5.3).

**Lexical Diversity** Another feature described by Osgood & Walker (1959) in their study showed how they used Type/Token Ratio (TTR) to discriminate between suicide notes and regular letters written to friends. TTR is computed by dividing the number of individual words by the number of total words in a corpus. They have found that due to the heightened cognitive state of mind of a suicidal person, there should be a lower TTR ratio in suicide notes than in letters to friends. This process has been called Lexical Diversity in Bird (2006) and was computed for all three corpora. Although it has been argued previously that people intending to commit suicide can demonstrate similar behavioural patterns like people who suffer from depression, it can now be seen that there is a significant difference in how many different words they use in a note (Table 5.3). It can also be seen that the LH

corpus is less lexical diverse than the GSN1 and DL1 corpora. One explanation for this phenomenon could be because the LH corpus is shorter in length, and therefore the number of individual words is lower compared to the other two corpora.

**Topics per Note** LIWC (Tausczik & Pennebaker 2010) has been used in order to identify a note writer's personal concern and topics analysed including money, family and death. The reason these topics have been chosen for analysis is that organisations such as Mind (2013) note that these are the most common reasons for suicide. The following Table 5.4 gives an overview of the topics note writers are most concerned about. It can be seen that there is a clear difference in topics referenced within the three corpora. In the GSN1 corpus, the highest reference was made to work and money, whereas in the LH and DL1 corpora, the highest reference was made to work and leisure. The findings of the GSN1 corpus reflect some of the topics mentioned in the Mind (2013) report on reasons for suicide. Furthermore, it could be argued that the topics work and money could mean people are writing about financial insecurity which may be due to not earning enough money or not being able to work. Since the LH and DL1 corpus both have a high reference to work and leisure, it could be verified in future work that there may be a good balance (LH corpus) and a negative balance in the writer's life (DL1 corpus) regarding those two activities in the individual corpora. The lowest reference in the GSN1 corpus was made to home, which contradicts the findings of Handelman & Lester (2007) who argue that the lowest reference in suicide notes is made to religion. This may be due to the type of suicide notes collected as there are also three notes included from martyrs' deaths. As it can be seen in Table 5.4 the lowest number of references to death was made in the LH corpus, which may be due to the overall happy tone in the corpus. Finally, the low reference to religion in the DL1 corpus may be because note writers do not look for external reasons for their current situation.

Corpora	GSN1	LH	DL1
Work	1.28	0.49	0.97
Leisure	0.55	0.31	1.03
Home	0.53	0.22	0.51
Money	1.45	0.27	0.30
Religion	0.88	0.3	0.09
Death	0.74	0.01	0.64

Table 5.4: Average reference to topic per note.

**Sentiment Features** Substantial research has been conducted on the emotional words of people who contemplate suicide or have committed suicide (Lester & Leenaars 1988) over many years. This research also included the sentiment conveyed in suicide notes and Pestian et al. (2012) have argued that a total of fifteen different sentiment features are most significant to suicide notes. Table 5.5 shows the number of sentiment features occurring in each of the datasets.

There are some general observations to be made about the emotion distribution in the three different corpora. All 15 emotions considered in this work are present in the GSN1 and DL1 corpus; however, not all emotions are present in the LH dataset. It could be argued that the incompleteness of all features in the LH corpus is to be expected as these sentiment features were designed for a the clinical domain (Pestian et al. 2012). However, it is interesting to note that all sentiment features are present in the GSN1 and DL1 corpus, which could support the hypothesis that sentiments in both corpora are close. An important observation has been made by Leenaars (1988), who argues that there is a greater mix of emotions in suicide notes. This may be one of the reasons why the different emotions occur with a higher percentage in the GSN1 corpus and less or not at all in the other two. The Mental Health Foundation (2015) describes on their website typical feelings people experience when suffering from depression such as hopelessness, sorrow as well as anxiety. These emotional concepts match the ones primarily found in the DL1 corpus, and therefore it could be argued that overall the emotions found in the individual corpora demonstrate

that the collected notes reflect the purpose of each corpus.

<b>Corpora</b>	<b>GSN1</b>	<b>LH</b>	<b>DL1</b>
Instruction	15.23	1.93	2.39
Information	41.55	44.74	53.85
Anger	4.65	0.35	5.05
Fear	1.55	2.11	2.94
Blame	2.00	0.18	2.94
Hopelessness	6.26	0.70	10.37
Abuse	0.13	0.18	0.09
Sorrow	5.94	4.56	17.06
Guilt	4.39	1.93	0.09
Thankfulness	2.26	2.98	0.28
Forgiveness	2.65	0.00	0.09
Hopefulness	2.32	1.93	3.03
Love	10.13	9.30	0.55
Pride	0.32	2.11	0.46
Happiness	0.65	27.02	0.83

Table 5.5: Sentiment features in % across all three corpora based on the work by Pestian et al. (2012).

### 5.3.2 Analysis for Experiments 2, 3 and 4

In this section the datasets GSN2, DL1 and LS are used for analysis as well as GSN2, DL2, DL3, NEU1 and NEU2. All datasets are evaluated using LIWC Tausczik & Pennebaker (2010) only.

<b>Type</b>	Experiment 2			Experiment 3 and 4				
	<b>GSN2</b>	<b>LS</b>	<b>DL1</b>	<b>GSN3</b>	<b>NEU1</b>	<b>NEU2</b>	<b>DL2</b>	<b>DL3</b>
Word Count	110.65	109.72	98.58	155.43	198.89	247.70	180.25	182.78
Word per Sent	14.87	11.42	16.88	16.20	20.34	18.32	17.32	17.83
SixItr	12.10	9.84	12.83	12.10	16.48	17.09	13.99	13.91
Analytic	33.63	30.14	20.12	32.85	53.19	50.95	29.04	25.91
Clout	47.73	67.68	19.94	46.73	45.08	47.54	23.64	22.88
Authentic	61.69	52.57	82.88	64.21	55.93	54.51	82.18	81.65
Tone	54.83	75.43	25.51	54.67	53.09	51.60	23.25	23.06

Table 5.6: LIWC Dimension Analysis for Experiment 2, 3 and 4 showing the average number of occurrence per note.

**Dimension Analysis** It has previously been argued by Tausczik & Pennebaker (2010) that the words people use can give insight into the emotions, thoughts and motivations of a person, where LIWC dimensions correlate emotions as well as social relationships.

Firstly, the word count per note is analysed as well as the different dimensions of each dataset (see Table 5.6). In Experiment 2 the word count is highest in GSN2 and lowest in DL1 notes, whereas LS notes are closer to GSN2 notes. This was also noted by Gregory (1999), who has argued that the overall greater length of a suicide note could be due to the writer knowing that they cannot convey any information at a later point. Experiment 3 also shows that DL2 and DL3 posts are lowest in word count, whilst NEU2 posts are highest. It was also found by Just et al. (2017) that the most concerning suicide-related content found on Twitter also had a higher word count here the highest word count is in blog posts, whilst GSN3 notes have the lowest word count.

This could be due to the fact that there is no character restriction placed upon bloggers. The number of *words per sentence* in Experiment 1 are highest for DL1 notes and are lowest in LS notes. In Experiment 3 the words per sentence are highest in DL2/DL3 posts and lowest in GSN3 notes. Research by Osgood & Walker (1959) has suggested that people in stressful situations break their communication down into shorter units. This may indicate alleviated stress levels in individuals writing notes before receiving the death sentence or before taking their own life.

*Clout* stands for the social status or confidence expressed in a person's use of language (Pennebaker et al. 2014). In Experiment 2, this dimension is highest for people writing their last statements, whereas depressed people rank lowest on this. Similarly, in Experiment 3 this feature is highest for people writing blog posts, whereas DL2/DL3 notes rank lowest on this. Cohan et al. (2018) have noted that this might be because depressed individuals often have a lower socioeconomic status.

The *Tone* of a note refers to the emotional tone, including both positive and negative emotions, where numbers below 50 indicate a more negative emotional tone Cohn et al. (2004). Results of Experiment 2 show that the tone for LS notes is highest overall and lowest in DL1 notes, indicating a more overall negative tone in DL1 and positive tone in LS. In Experiment 3 the tone for GSN3 is highest overall and the lowest in DL2/DL3, also indicating a more overall negative tone in DL and positive tone in GSN.

*SixItr* in Table 5.6 refers to words that are longer than 6 letters and are meant to indicate the social class and level of education of a person (Pennebaker et al. 2014). It can be seen that the lowest scores were observed by LS and the highest by DL1 writers. There is no additional information available for both GSN2 and DL1 note writers, but it is known that LS writers on average left education after year 10 (Texas Department of Criminal Justices 2019). Results of Experiment 3 show that that the lowest scores were observed by GSN3 and the highest by blog posts writers. There is no additional information available to evaluate both educational or socio-economic factors, but it could be argued that the lack of longer words in GSN notes is due to the argument made by Osgood & Walker (1959) that GSN writers break communication down to shorter units due to the alleviated stress-levels and not their educational or socio-economic background.

The *Analytical thinking dimension* indicates to which extent people use ‘formal, logical, and hierarchical thinking patterns’ (Tausczik & Pennebaker 2010). Experiment 2 shows that this score is very similar for GSN2 and LS notes, but considerably lower for the DL1 corpus. NEU posts score highest in this category and GSN3 writers score lowest as shown in Experiment 3. It has been found that people who score low in analytical thinking tend to write and use spoken language more narratively and focus on the present as well as personal experiences, compared to people who score highly in this (Pennebaker et al. 2014).

The term *Authenticity* refers to which extent people write about themselves in an honest way, where they are typically portrayed as more humble, personal and vulnerable (Newman et al. 2003). DL1 notes were the most authentic whilst the least authentic words were written by LS writers, previous studies have confirmed this when comparing demographics with and without mental health conditions (Cohan et al. 2018). Experiment 3 results show that DL2/DL3 notes were the most authentically written whilst the least authentic words were written in NEU1/NEU 2 posts. Arguably blog posts do not require a writer to be vulnerable and with the increasing amount of blogging as a marketing tool there may be less personal or humble language found in these posts.

**Function Words and Content Words** The next section looks at selected function words and grammatical differences, which can be split into two categories called *Function Words* (see Table 5.7), reflecting how humans communicate and *Content words* (see Table 5.7), demonstrating what humans say (Tausczik & Pennebaker 2010). Function words refer to a variety of different word categories, such as pronouns or auxiliary verbs and make up the majority of all words that are persons uses (Tausczik & Pennebaker 2010). It was found that there is a difference in how human brains process function and content words (Miller 1991). Research has also found that function words have been connected with indicators of people’s social and psychological worlds Tausczik & Pennebaker (2010), where it has been argued that the use of function words require basic skills.

Type	Experiment 2			Experiment 3 and 4				
	GSN2	LS	DL1	GSN3	NEU1	NEU2	DL2	DL3
Function	56.35	56.33	60.20	56.80	47.87	49.12	58.27	59.35
Personal pronouns	16.23	20.44	15.19	15.85	10.06	10.23	14.35	14.32
I	11.04	12.65	12.8	10.64	6.45	6.26	11.63	11.60
Negations	2.71	1.71	4.06	2.87	1.47	1.65	3.10	3.24
Verb	19.29	19.58	21.65	19.06	16.46	15.92	21.10	21.40
Adjective	4.45	2.58	4.98	4.54	4.71	4.25	4.80	4.82
Adverb	4.43	3.14	7.69	4.79	5.27	5.64	6.91	7.20

Table 5.7: LIWC Function and Content Words for Experiment 2, 3 and 4.

The results of Experiment 2 show that the highest amount of function words were used in DL1 notes, whilst both GSN2 and LS have a similar amount of function words. Similarly, in Experiment 3, the highest amount of function words were used in DL2/DL3 posts, whilst NEU1/NEU2 posts have the least amount of function words. Rude et al. (2004) has found that high usage, specifically of first-person singular pronouns ('I') could indicate higher emotional and/or physical pain as the focus of their attention is towards themselves. Overall Just et al. (2017) has also identified a larger amount of personal pronouns in suicide-related social media content. This may be the reason why GSN notes are high in personal pronouns overall and the first-person singular, whilst NEU1/NEU2 are lowest in both categories. However, it has to be noted that LS notes are considerably higher in personal pronouns when compared to GSN2 notes, which could indicate a higher level of emotional pain due to the lack of choice of dying involved.

Previous work by Hancock et al. (2007) has found that people use a higher amount of negations when also expressing negative emotions and used fewer words overall, compared to more positive emotions. This seems to be also true for the number of negations used in Experiment 2, where the number of *Negations* is highest in the DL1 corpus and lowest in the LS corpus. The same results apply to Experiment 3, where the number of negations used were also highest in the DL corpus and lowest in the NEU1/NEU2. As it can be seen in Table 5.6, whilst the word count for both Experiment 2 and 3 showed that DL1, DL2 and DL3 are lowest, the number of negative emotions was highest in DL notes and posts.

Furthermore, it was found that *Verbs*, *Adverb* and *Adjectives* are often used to communicate content; however previous studies have found (Jones & Bennell 2007, Gregory 1999) that individuals that die by suicide are under a higher drive and therefore would reference a higher amount of objects (through nouns) rather than using descriptive language such as adjectives and adverbs. This may explain why



the number of adjectives and adverbs are lowest in GSN3 notes, and highest in DL2 / DL3 notes. However, it is notable that in Experiment 2, LS notes contain the least amount of adjectives and adverbs, which may indicate an even higher drive than people who die by suicide.

**Affect Analysis** The analysis of emotions in suicide notes and last statements has often been addressed in research (Schoene & Dethlefs 2018, Lester & Gunn III 2013). Table 5.8 shows sentiments and emotions that were detected for the datasets using LIWC.

Type	Experiment 2			Experiment 3 and 4				
	GSN2	LS	DL1	GSN3	NEU1	NEU2	DL2	DL3
Affect	9.1	11.58	8.44	8.92	5.90	5.84	7.78	8.06
Positive emotion	5.86	8.99	3.15	5.69	3.91	3.82	2.97	3.01
Negative emotion	3.15	2.58	5.21	3.16	1.95	1.95	4.72	4.97
Anxiety	0.26	0.16	0.5	0.28	0.27	0.22	0.65	0.67
Anger	0.61	0.65	1.03	0.62	0.68	0.68	1.24	1.35
Sadness	1.09	1.08	2.53	1.06	0.38	0.39	1.74	1.86

Table 5.8: LIWC Affect Analysis for Experiment 2, 3 and 4

Experiment 2 shows that the number of *Affect words* is highest in LS notes, whilst they are lowest in DL1 notes, this could be related to the argument made in regards to the emotional *Tone* of a note. This argument also applies to the amount of *Negative emotions* as they are highest in DL1 notes and *Positive emotions* as these are highest in LS notes. The results of Experiment 3 indicate that overall the highest amount of affect words are in DL2 / DL3 posts, whilst the lowest amount is in blog posts. This may also relate back to the level of authenticity usually found in DL2 / DL3 posts and lacking in NEU1 / NEU2 posts due to blog posts writers not being as vulnerable. Furthermore, it was found that the highest amount of *Negative emotions* are also highest in DL2 / DL3 posts and lowest in NEU1 / NEU2 posts, similarly as before this may refer back to the *Tone* used in those type of corpora where a higher amount of negative emotions are often correlated with the tone used in a note. Also, positive emotions are highest in GSN3 notes, whilst they are lowest in

DL2 / DL3 posts. This has been found previously by Schoene & Dethlefs (2016), who have found that emotions such as ‘love’ are more frequently found in genuine suicide notes compared to other corpora. Previous research has analysed the amount of *Anger* and *Sadness* in genuine suicide notes and depression notes, where it was shown that they are more prevalent in depression notes, because these are typical feelings expressed when people suffer from depression (Schoene & Dethlefs 2016).

**Social Processes** This section highlights the social relationships and references to family and friends in each dataset.

Type	Experiment 2			Experiment 3 and 4				
	GSN2	LS	DL1	GSN3	NEU1	NEU2	DL2	DL3
Social processes	12.21	18.19	8.33	11.87	7.93	8.42	8.10	8.10
Family	1.17	2.17	0.47	1.11	0.33	0.40	0.49	0.42
Friends	0.77	0.38	0.73	0.71	0.31	0.44	0.62	0.56

Table 5.9: LIWC Social Processes for Experiment 2, 3 and 4

The results of Experiment 2 show that the highest amount of social processes can be found in LS and the lowest in DL1. Furthermore, LS notes tend to speak most about family relations and least about friends, and this was also found by Kelly & Foley (2017) who found a low frequency in interpersonal relationships in last statements. Experiment 3 shows that the highest amount of social processes can be found in GSN3, and the notes tend to contain mostly references to family relations and less to friends. Furthermore, it can be seen that social processes overall are lowest in NEU1, which could be due to the nature of personal blogs where writers mainly tend to write about their own experiences.

**Cognitive Processes** The term *Cognitive processes* encompasses a number of different aspects, including *Insight* and *Cause*. Table 5.10 shows the results for all datasets.

Experiment 2 shows that the highest amount of cognitive processes was in DL notes,

Type	Experiment 2			Experiment 3 and 4				
	GSN2	LS	DL1	GSN3	NEU1	NEU2	DL2	DL3
Cognitive Processes	12.19	10.85	16.77	12.63	10.41	10.63	16.31	16.30
Insight	2.37	2.3	4.07	2.46	2.09	2.12	3.83	3.54
Cause	0.95	0.8	1.94	1.07	1.47	1.34	2.09	2.06
Tentativeness	2.57	1.5	3.23	2.65	2.52	2.63	3.38	3.66

Table 5.10: LIWC Psychological Processes for Experiment 2, 3 and 4

and the lowest in LS notes. Boals & Klein (2005) have found that people use more cognitive mechanisms to cope with traumatic events such as breakups by using more causal words to organise and explain events and thoughts for themselves. Arguably this explains why there is a lower amount in LS notes as LS writers often have a long time to organise their thoughts, events and feelings whilst waiting for their sentence (Death Penalty Information Centre 2019). It can also be seen in Experiment 3 that the lowest amount of cognitive processes is found in NEU1 / NEU2 posts, whereas the highest amount are in DL2 / DL3 posts. One explanation for this could be that often people who struggle with depression may have had traumatic or challenging experiences in their past (UK 2020) that they now write about online.

*Insight* encompasses words such as *think* or *consider*, whilst *Cause* encompasses words that express reasoning or causation of events, e.g.: *because* or *hence*. These terms have previously been coined as *cognitive process words* by Gregory (1999), who argued that these words are less used in genuine suicide notes as the writer has already finished the decision making process whilst other types of discourse would still try to justify and reason over events and choices. This can also be found in the results of Experiment 2, where both GSN2 and LS notes show similar, but a lower frequency of terms in those to categories compared to DL1 writers. Results for Experiment 3 show similar results, where both Insight and Cause are low in GSN3 notes, but high in DL2 / DL3 notes.

*Tentativeness* refers to the language use that indicates a person is uncertain about a topic and uses several filler words. It has been argued that participants who

use more tentative words, may not have expressed an event to another person and therefore have not processed an event yet and it has not been formed into a story (Tausczik & Pennebaker 2010). The amount of tentative words used in DL1 / DL2 / DL3 notes is highest in both Experiment 2 and 3, whilst it is lowest in LS for Experiment 2 and lowest in GSN3 notes for Experiment 3. This might be because LS writers already had to reiterate over certain events multiple times as they go through the process of prosecution, and people who die by suicide have made their final decision (Gregory 1999).

**Personal Concerns** This category refers to the topics most commonly brought up in the different notes, where Table 5.11 shows the results for all datasets.

Type	Experiment 2			Experiment 3 and 4				
	GSN2	LS	DL1	GSN3	NEU1	NEU2	DL2	DL3
Work	1.24	0.41	0.99	1.26	1.96	1.96	1.70	1.50
Money	0.68	0.18	0.31	0.67	0.36	0.49	0.40	0.35
Leisure	0.56	1.12	0.95	0.54	1.56	1.51	0.67	0.77
Home	0.45	1.29	0.5	0.46	0.39	0.39	0.48	0.41
Religion	0.82	2.7	0.09	0.68	0.39	0.32	0.14	0.12
Death	0.76	0.68	0.64	0.73	0.14	0.17	0.36	0.57

Table 5.11: LIWC Personal Concerns for Experiment 2, 3 and 4.

Experiment 2 shows that both *Money* and *Work* are most often referred to in GSN notes and lowest in LS notes. This might be because Mind (2013) lists these two topics as some of the most common reasons for a person to commit suicide. *Religion* is most commonly referenced in LS notes, which confirms the previous analysis of such notes (Foley & Kelly 2018, Kelly & Foley 2017) and lowest in DL1 notes. Just et al. (2017) has found that the topic of *Death* is commonly referenced in suicide-related communication on Twitter. This was also found in this dataset, where GSN2 notes most commonly referenced death, whilst DL1 notes were least likely to reference this topic. These findings are also true for results in Experiment 3. Furthermore the references to *Leisure* and *Home* are highest in LS notes and

lowest in GSN2 notes.

Results for Experiment 3 indicate that *Work* is most often referred to in NEU1/NEU2 posts and lowest in GSN3 notes, which could be due to blogging often being used for marketing and advertising (Onishi & Manchanda 2012). Similar to the results for Experiment 1, *Money* is most often referenced in GSN3 notes and lowest in DL2/DL3 posts. Furthermore, the references to *Leisure* are highest in the NEU1/NEU2 posts and lowest in GSN3 notes. References to *Home* were highest in GSN3 notes and lowest in NEU1/NEU2 posts, which might be due to GSN3 writers often leaving instructions behind (Pestian et al. 2010), which could reference places within a house.

**Time Orientation and Relativity** Looking at the *Time Orientation* of a note can give an interesting insight into the temporal focus of attention and differences in verb tenses can show psychological distance or to which extend disclosed events have been processed (Tausczik & Pennebaker 2010). Table 5.12 show the results for all datasets.

Type	Experiment 2			Experiment 3 and 4				
	GSN2	LS	DL1	GSN3	NEU1	NEU2	DL2	DL3
Focus past	3.24	2.86	3.32	3.37	3.21	3.46	4.14	3.71
Focus present	14.39	1.43	16.11	14.14	11.15	10.65	14.97	15.67
Focus future	2.1	2.27	1.51	1.89	1.72	1.54	1.22	1.44
Relativity	10.18	7.66	12.71	10.72	13.34	12.95	13.40	13.16

Table 5.12: LIWC Time orientation for Experiment 2, 3 and 4.

Experiment 2 shows that the focus of LS notes is primarily in the past, whilst GSN2 and DL1 notes focus on the present. The high focus on the past in DL1 notes, as well as GSN2 notes, could be because these notes might draw on their past experiences to express the issues of their current situation or problems. The most frequent use of future tense is in LS letters which could be due to a LS notes writer's common focus on afterlife (Heflick 2005). *Relativity* refers to references to space, motion and

time in a note. Research by Bond & Lee (2005) has found that the reference to fewer motion words was important when predicting whether a prisoner is deceptive. Results for Experiment 3 show that the focus of DL2 / DL3 posts is primarily in the past whilst GSN3 and NEU1 / NEU2 posts focus on the future. The most frequent use of future tense in GSN3 notes could be due to the writer leaving behind instructions for others (Pestian et al. 2012).

### 5.3.3 Cohen's d effect size

In order to provide some insights and show statistical significance of the features discovered during the linguistic analysis, Cohen's d effect size was used to calculate the pairwise importance (Cohen 2013) of each feature. This was done purposefully for Experiment 3 and 4 only because the main goal of this work was to create a 'real-world' scenario, where suicide notes are distinguished from both 'neutral' and 'depressed' posts.

An effect size over  $d=0.2$  (highlighted blue) indicates a small effect,  $d=0.5$  (highlighted green) indicates a medium effect and  $d=0.8$  (highlighted yellow) shows a large effect. Furthermore, Cohen (2013) argued that an effect size of  $d=0.5$  or higher should be easily seen by humans in real-world examples. The minimum value for the Cohen's D effect size is above 0, where the maximum can be over 2 indicating that the effect is larger than 1 standard deviation. It can be seen in Table 5.13, that most features have a small effect (36.48%), whereas both medium and large effects make up 22.97% and 6.08% of the features respectively and should be clearly visible when examining any posts or notes. Furthermore, it can be seen that categories such as *Dimensional Analysis*, *Affect* or *Function words* show a medium to large effect size across its subcategories, whereas *Cognitive Processes* seem to only have a small to medium effect size for GSN to DL pairwise comparison. Also, it can be seen that features such as *Word per sentence*, *Adjectives* or *Home*

do not have any effect on any on the datasets. Other features such as *Clout*, *Tone*, *Anxiety*, *Anger*, *Insight*, *Tentativeness* and *Focus past* do not appear to be important when measuring statistical significance between GSN3 and NEU1/NEU2 posts. In comparison, there is only one feature (*Leisure*) that is not statistically significant when comparing GSN3 to DL2/DL3 notes. When comparing GSN3 to DL2/DL3 notes, the *Affect* category seems to be most important, whereas for a comparison of GSN3 to NEU1/NEU2 the *Function word* category is most significant. Therefore, it could be argued that in future work, a more fine-grained analysis of sentiment would provide more insight and distinct features to accurately classify suicide notes from depressed notes. On the other hand, for a comparison of suicide notes to ‘neutral’ posts, a focus on *function* words seems most appropriate. Overall, the category that seems most important across all three datasets is *Function words*, where only one feature (*Negations*) is not statistically significant when comparing GSN3 to DL2.

Type	GSN3/NEU1	GSN3/NEU2	GSN3/DL2	GSN3/DL3
Word Count	0.207	0.165	0.138	0.24
Word per Sent	0.11	0.03	0.078	0.111
SixItr	0.448	0.61	0.339	0.181
Analytic	0.712	0.861	0.163	0.348
Clout	0.065	0.137	0.808	0.928
Authentic	0.237	0.354	0.64	0.723
Tone	0.036	0.068	0.914	0.962
Function	0.669	0.789	0.217	0.392
Pers. pro.	0.901	1.132	0.248	0.203
I	0.814	1.048	0.21	0.319
Negations	0.658	0.706	0.102	0.3
Verb	0.344	0.627	0.385	0.533
Adjective	0.048	0.04	0.085	0.008
Adverb	0.142	0.236	0.673	0.78
Affect	0.632	0.625	0.239	0.259
Pos. emotion	0.438	0.412	0.714	0.754
Neg. emotion	0.456	0.508	0.516	0.528
Anxiety	0.014	0.12	0.396	0.367
Anger	0.034	0.015	0.37	0.393
Sadness	0.499	0.396	0.35	0.348
Social proc.	0.605	0.618	0.586	0.678
Family	0.501	0.521	0.38	0.448
Friends	0.333	0.178	0.059	0.122
Cognitive proc.	0.374	0.402	0.668	0.701
Insight	0.151	0.092	0.554	0.467
Cause	0.219	0.082	0.629	0.631
Tentativeness	0.053	0.015	0.301	0.371
Focus past	0.045	0.006	0.233	0.112
Focus present	0.464	0.702	0.166	0.397
Focus future	0.067	0.184	0.329	0.107
Relativity	0.451	0.391	0.502	0.493
Work	0.302	0.296	0.2	0.177
Money	0.255	0.304	0.21	0.221
Leisure	0.593	0.584	0.105	0.141
Home	0.059	0.108	0.023	0.045
Religion	0.143	0.257	0.288	0.288
Death	0.437	0.519	0.272	0.11

Table 5.13: Table 8: Cohens' d effect size for pairwise significance testing of linguistic features.



## 5.4 Experiment 1: Bidirectional LSTM with attention

Experiment 1 was conducted to test how well deep learning methodologies would perform compared to traditional machine learning techniques in classifying suicide notes from other type of notes. More specifically, it is shown that similar results can be achieved using a recurrent neural network-based model that works on the word-level alone, i.e. without the requirement for hand-labelled data or features that are often used in traditional machine learning-based approaches.

**Experimental setup** There are three different experiments conducted for this task, where word embeddings are used (Mikolov, Chen, Corrado & Dean 2013) as input features into the learning model. All word embeddings were pre-trained on the dataset with the word embedding dimension set to 100, and the maximum number of most common words was set to 5000. Furthermore, 80% of the data is used for training and 20% for validation. All sentiment features are the same features discussed in Schoene & Dethlefs (2016). All experiments were conducted using Keras (Chollet et al. 2015), including a custom attention layer. To assess the importance of different features, three experiments are conducted, where Table 5.14 shows a mock example of the data used in each experiment. The learning model used in these experiments is a Bidirectional LSTM (biLSTM) with attention as outlined in Chapter 2, section 2.3.3.

Experiment type	Mock example of input data
Classification based on text-only	<i>I hate dogs, but I love cats.</i>
Classification based on sentiment and text	<i>I hate dogs [hate], but I love cats [love]</i>
Classification based on sentiment features only	<i>[hate],[love]</i>

Table 5.14: Description of experiment series.

### 5.4.1 Results and Evaluation

All results are summarised in Table 5.15. As it can be seen, the best results are achieved by the biLSTM model with attention based on sentiment features only. Classification from text and sentiments is second best, while text-only is only slightly worse than using both text and emotions. Overall, it therefore seems that hand-labelled emotions are important to make accurate predictions; however, classification from unlabelled data is still significantly better than a majority baseline of 33.33%. In the following, an analysis of the result is provided on text and sentiment features and sentiment features only to shed some light on relevant features and patterns.

Model	Text-only	Text and sentiment	Sentiment features
Majority	33.33%	33.33%	20.46%
Vanilla LSTM	53.77%	50.00%	57.55%
biLSTM with attention	69.41%	71.76%	75.29%

Table 5.15: Experiment results comparing a vanilla LSTM with a bidirectional LSTM with attention for classification from text only, emotions only and text and emotions. All results are the test accuracy in %.

The results show that learning with unannotated data is possible (see Table 5.15), achieving an accuracy of 55.00% with a vanilla LSTM and 75.95% with the biLSTM with an attention mechanism. The results achieved with the biLSTM attention LSTM are encouraging as they are exceeding those of Yang et al. (2012), who achieved 61.39% accuracy using traditional machine learning techniques in a similar task. This result also shows that sentiment features are important and relevant to more accurate classification of suicide notes. Figures 5.6, 5.8 and 5.7 show example predictions for each category illustrating prediction making.

**Results for sentiment features only** This experiment was inspired by the observations made in Gunn & Lester (2015), where qualitative reports indicated that positive emotions increase the closer a person gets to dying by suicide. Therefore

dear Elinor [information] the reason for my  
despondency is that you'd prefer the company of  
almost anyone to mine [anger] you told me you had  
nothing to look forward to on week ends [blame] you  
told me you preferred living alone [blame] ...

i'm so tired of feeling sad [sorrow]

its like a shot of uber confidence i always feel  
better about myself for making someone  
happy just by being me [happiness]

Figure 5.6: Example of a GSN1 note

Figure 5.7: Example of a DL1 note

Figure 5.8: Example of a LH note

all textual data was removed from the data, and only sentiment features were used that occur within a note. This hypothesis might also explain why sentiments such as love are occurring more frequently in the GSN1 corpus compared to the other two corpora. In order to further investigate this phenomenon, we chose ten random notes from each corpus and visualised the sentiment sequences using heatmaps (see Figure 5.9). Figure 5.10 describes the key we used to represent sentiment categories numerically, where all light coloured sentiment could fall into the category 'negative', and all darker coloured sentiment fall into the category 'positive'. Due to the varying length of individual notes, a placeholder has been assigned to the number 16.

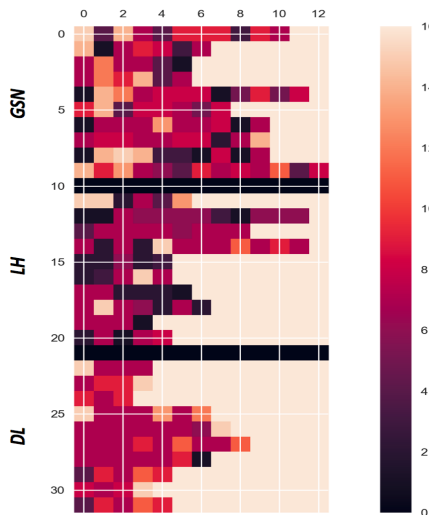


Figure 5.9: Sentiment feature heatmap

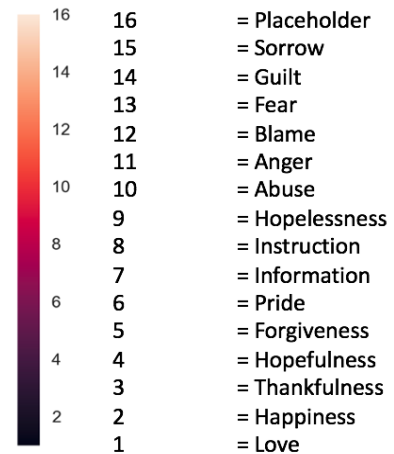


Figure 5.10: Legend for the sentiment feature heatmap

The results displayed in Figure 5.9 show for the DL1 corpus that there is not a large variety of different sentiments in the selected examples and would mostly fit into the category of 'negative' emotions such as sorrow or anger. However, due to

the nature of the notes collected for this corpus, it was to be anticipated that most emotions would be negative. A similar principle applies to the results shown for the LH Corpus, where the majority of sentiments could be labelled ‘positive’, with only a small amount of variation throughout the notes. Finally, the results for the GSN1 demonstrate that there could be some evidence for the hypothesis formulated by Gunn & Lester (2015). Many of the sentiments that appear towards the end of a note could be labelled ‘positive’, whereas the lighter colours at the start of each note indicate more ‘negative’ sentiments.

**Conclusion** Overall the results show that it is possible to accurately classify both unannotated data and sentiment category annotated data using deep learning methodologies, whilst achieving results that are close to previous results achieved by traditional machine learning algorithms. This may lead to some interesting discussion about how important hand-crafted features are overall when considering the amount of time and money is used to develop these. In addition to this, the experiments have shown that the sentiment features contribute to the accurate classification of suicide notes. It has also been shown that one can accurately classify suicide notes by just using sentiment features. A heatmap was used to further shed light on this point and provide initial evidence to the hypothesis. This could mean that there is some importance in how often and in which order sentiments occur in suicide notes. Finally, it has been observed that classification accuracy can be improved by using more tailored learning models as it will be explored in the next section.

## 5.5 Experiment 2: Dilated LSTM with attention

Last Statements have been of interest to researchers in both the legal and mental health community, because an inmate’s last statement is written, similarly to a

suicide note, closely before their death (Texas Department of Criminal Justices 2019). However, the main difference remains that unlike in cases of suicide, inmates on death row have no choice left in regards to when, how and where they will die. Furthermore, there has been extensive analysis conducted on the mental health of death row inmates where depression was one of the most common mental illnesses. Work in suicide note identification has also compared the different states of mind of depressed and suicidal people because depression is often related to suicide (Mind 2013). In Experiment 2, the previous RNN architecture is extended by dilations and recurrent skip connections. These changes were made to enable the network to model long sequences at the document level. By exploring and comparing suicide notes with last statements and depressed notes, both qualitatively and quantitatively, it could contribute to finding further differentiating factors and aid in identifying suicidal messages online in the future. Datasets GSN1, DL1 and LS are used in this Experiment.

The learning model used in these Experiments is a Dilated LSTM with attention as outlined in Chapter 2.3, section 2.3.4.

**Experimental setup** Three performance baselines are established on the datasets by using three different algorithms previously used on similar datasets. Firstly, the ZeroR and LMT (Logistic Model Tree) are used as previously used by Schoene & Dethlefs (2016). Additional benchmarks for the algorithm are chosen against the originally proposed Bidirectional LSTM with attention introduced by Yang et al. (2016), which was also used on similar existing datasets before (Schoene & Dethlefs 2018). Furthermore, the Dilated LSTM with attention is benchmarked against two other types of recurrent neural networks. *200-dimensional* word embeddings are used as input into each network and all neural networks share the same hyper-parameters, where learning rate = 0.001, batch size = 128, dropout = 0.5, hidden size = 150 units and the *Adam* optimiser is used. For the proposed model - the

Dilated LSTM with attention - the number of dilations is established empirically. There are 2 dilated layers with exponentially increasing dilations starting at 1. Due to the size of the dataset the data was split into 70% training, 15% validation and 15% test data.

### 5.5.1 Results and Evaluation

All three datasets are used for the Experiments, Table 5.16 shows the results for the Experiments series.

Model	Test Accuracy	Precision	Recall	F1-score
ZeroR	42.85	0.43	0.41	0.42
LMT	80.35	0.81	0.79	0.80
LSTM	62.16	0.63	0.61	0.62
BiLSTM	65.82	0.66	0.64	0.65
BiLSTM with attention	82.27	0.85	0.83	0.84
DLSTM with attention	87.34	0.88	0.87	0.87

Table 5.16: Experiment results using test accuracy and F-1 score of different learning models.

It can be seen in Table 5.16 that the Dilated LSTM with an attention layer outperforms the BiLSTM with attention by 5.07%. Furthermore, it was found that both the LMT and a vanilla bidirectional LSTM outperform a standard LSTM on this task. Previous results on similar tasks have yielded an accuracy of 69.41% using BiLSTM with attention (Schoene & Dethlefs 2018) and 86% using an LMT (Schoene & Dethlefs 2016). In Figure 5.11a the confusion matrix for the Dilated LSTM with attention layer is shown, where it is found that LS notes are most likely and DL1 notes are least likely to be accurately predicted. The same applies to results of the competing model (BiLSTM with attention), Figure 5.11b shows that this model still misclassifies LS notes with DL1 notes.

Table 5.17 shows the results for each dataset, where in both tables DL1 notes are least likely to be classified correctly and LS notes are most likely to be classified

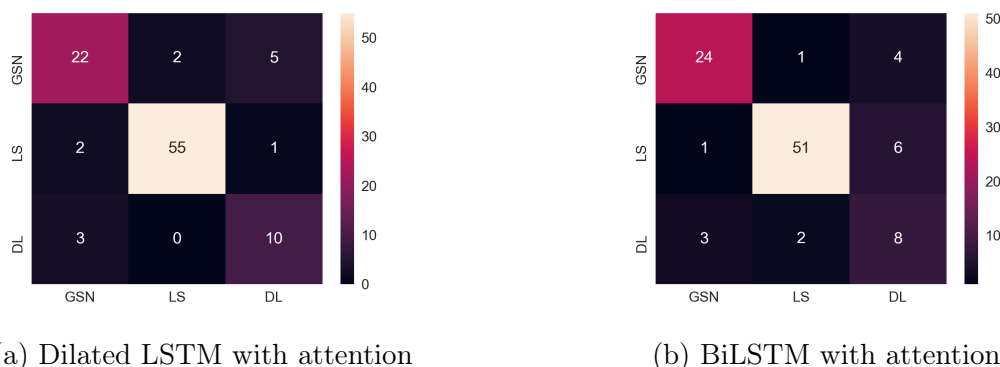


Figure 5.11: Confusion Matrices of the predicted test set labels of the BiLSTM and DLSTM with attention.

correctly.

	Dilated LSTM with attention			BiLSTM with attention		
<b>Data</b>	<b>Precision</b>	<b>Recall</b>	<b>F1-score</b>	<b>Precision</b>	<b>Recall</b>	<b>F1-score</b>
GSN1	0.81	0.76	0.79	0.86	0.83	0.84
LS	0.96	0.95	0.96	0.94	0.88	0.91
DL1	0.62	0.77	0.69	0.44	0.62	0.52

Table 5.17: Comparison of F-1 scores per dataset for both a Dilated LSTM with attention and BiLSTM with attention

**Linguistic Evaluation** In order to see which features are most important to accurate classification, examples are visualised from the test set of each dataset, where Figures 5.12a, 5.13a and 5.14a show the visualisation of attention weights in the *GSN2*, *LS* and *DL1* datasets respectively. Furthermore, three more examples of the test data are visualised to show typical errors the learning model makes in Figures 5.12b, 5.13b and 5.14b. Words highlighted in darker shades have a higher attention weight.

The most important words highlighted in a last statement note (see Figure 5.12a) are personal pronouns as well as an apology and expression of love towards friends and family members. This corresponds with the higher amount of personal pronouns, positive emotions and references to family in LS notes compared to GSN2 and DL1 notes. Furthermore, it can be seen that there is a low amount of cognitive

process words and more action verbs such as *'killing'* or *'hurt'*, which could confirm that inmates have had more time to process events and thoughts and do not need cognitive words as a coping mechanism anymore (Boals & Klein 2005). Figure 5.12b shows the errors the learning model made. In this instance the word *God* was replaced with *'up'*, when looking into the usage of the word *'up'* in other LS notes, it was found that it was commonly used in reference to religious topics such as *'God'*, *'heaven'* or *'up there'*.

(a) Example of a correctly classified LS note

(b) Error analysis of LS note

Figure 5.12: Comparison of attention weights in a correctly classified LS note and wrongly classified LS note

Figure 5.13a shows a GSN2 note, where the most important words are also pronouns, references to family, requests for forgiveness and endearments. Previous research has shown that forgiveness is an important feature as well as the giving instructions such as *'help'* or phrases like *'do not follow'* are key to accurately classify suicide notes (Pestian et al. 2010). Terms of endearment for loved ones at the start or towards the end of a note are also of importance (Gregory 1999). In Figure 5.13b a visualised GSN2 note is shown. Whilst there is still consistency in highlighting personal pronouns (e.g., *'you'*), it can be seen that the end of the note is missing and more action verbs such as *'hurt'* or *'take'* are more important.

The DL1 note in Figure 5.14a shows that there is a greater amount of cognitive process verbs present, such as *'feeling'* or *'know'* as well as negations, which confirms previous analyses using LIWC. The visualisation in Figure 5.14b demonstrates how the personal pronoun *'I'* has been removed from several DL1 notes, where DL1 notes are least likely to be predicted accurately as shown in Table 5.17.



my dearest family am terribly sick and it is all my fault blame no one but myself know it is going to hard with william and sister please see that charles gets a mickey mouse watch for his birthday jane am counting on you to take care of mother please do not follow in my footsteps elinor my darling know you did everything possible to avoid this but please forgive me as think it was the only way out god forgive me and help take care of my family

(a) Example of a correctly classified DL1 note

with jesus that have prayed for him to lookafter you and jane have prayed that you arent destroyed by this because that would be something could never be forgiven for my love for you has always been the deepest and hopefully ill see you again you are my mircale have accepted the lord jesus as my saviour but kow that he wouldnt condone this accept the just dues and pray that maybe you wont hurt anymore make our kid something for your strength and love does work miracles you and jesus pray can forgive me for copping out its me who accepts the responsibility of my actions apolagize to all of you beg jesus forgiveness love all of our friends loved ones pray for me know if there is heaven ill hopefully meet you there someday you have been and will always be the brightest ray of sunshine that eve entered my life and no one can take that away if see mom ill see that christopher is taken care of will try to be with him too love thos kids and am asking yours jesus forgiveness

(b) Error analysis of GSN2 note

Figure 5.13: Comparison of attention weights in a correctly classified GSN2 note and wrongly classified DL1 note

has anyone ever been so depressed for so long that you cant even tell what youre feeling anymore dont know if im depressed or just empty at this point

(a) Example of a correctly classified DL1 note

spend most of my weekends simply hating myself thought could figure it out but its tough guess it is chemical just dont really know what to do dont seem to be willing to do the things needed to get more out of life or maybe my expectations are all out of whack feeling like really dont get it

(b) Error analysis of DL1 note

Figure 5.14: Comparison of attention weights in a correctly classified DL1 note and wrongly classified DL1 note

**Conclusion** These Experiments have shown that using a recurrent neural network that can take advantage of long sequence helps to improve classification accuracies by 6.99 % and by 5.07 % compared to a competitor model. Furthermore, the evaluation of the visualised attention weights has shown that the neural network pays attention to similar linguistic features as provided by LIWC and found in human evaluated related research. All experiments conducted so far have not been reflective of a ‘real-world’ scenario, where the goal might be to distinguish a suicide note from more neutral data or where a suicide note is not part of a balanced dataset. Therefore the following section will look at introducing a new task of classifying suicide notes in a more realistic scenario, where a suicide note may be need to be distinguished from more neutral data.

## 5.6 Experiment 3 and 4: Dilated LSTM with ranked units

This section introduces a new task of classifying suicide notes in a more realistic scenario, where the goal of Experiments 3 and 4 is to accurately classify suicide notes from neutral blog posts and post of people who suffer from depression. In these experiments, datasets GSN3, DL2, DL3, NEU1 and NEU2 are used as outlined in section 5.2. For the first experiment series (Experiment 3) a balanced dataset is used to classify suicide notes, depressed posts and blog posts to see how hard the task proves in this setting. The second experiment (Experiment 4) aims to make the task even more applicable to the real world, and both depressed and blog posts are increased to reflect the rarity of genuine suicide notes on social media platforms. Finally, the experimental results will be discussed, and a linguistic analysis will be provided through the visualisation of attention weights. However, it has to be acknowledged that there are some limitations to this type of work, where the data collected for this work is sourced from different social-media platforms. This could lead to additionally learning features that are associated to a specific source.

The learning model used in these experiments is a Dilated LSTM with ranked units.

**Learning Model** The chore architecture is similar to the dilated LSTM with attention as outlined in Chapter 3, section 3.4. The standard dilated LSTM is extended in two ways for these experiments. The standard dilated LSTM alleviates the problem of learning long sequences, but not each document has the same sequence length, so in order to overcome this variability a fixed boundaries is provided to each layer by reducing the number of hidden units per sub-LSTM hierarchically. Therefore larger sub-LSTMs focus on learning long-term dependencies, whilst smaller sub-LSTMs focus on more frequently occurring

short-term dependencies. This leads to improved performance as it has been shown in other contexts (El Hahi & Bengio 1996, Chung et al. 2016).

**Experimental setup** For the two different classification experiments, a Maximum Entropy classifier is used to establish a performance baseline. This is due to its suitability to textual data where conditional independence of the features cannot be assumed. Additionally, the proposed model is benchmarked against the originally proposed Bidirectional LSTM with attention proposed by Yang et al. (2016), as it also utilises attention.

Furthermore, the Dilated LSTM with ranked units is also benchmarked against two other types of RNNs. *200-dimensional* word embeddings are used as input into each network and all neural networks share the same hyper-parameters, where learning rate = 0.001, batch size = 128, dropout = 0.5 and the *Adam* optimiser is used. The full sequence length of each document is used as input. For the proposed model - the Dilated LSTM with ranked units - the number of dilations is established empirically. There are 2 dilated layers with exponentially increasing dilations starting at 1. The number of hidden units is adjusted according to the sequence length used as input to each sub-LSTM, where the number of hidden units is always half of the given sequence length. For example, given a sequence length of 160 and 2 dilations, the input length to the sub-LSTM is [160,80], whilst the number of hidden units adjusts from 80 to 40. For all other learning models, the number of hidden units is set to 300. For experiment 3, datasets GSN3, DL2 and NEU1 are used, which yields an overall dataset size of 633 posts. Due to the small size of the dataset k-fold cross-validation is used, where  $k = 10$ . For experiment 4 datasets GSN3, DL3 and NEU2 are used, where the overall dataset size is 5004, and the data is split into 80% training, 10% validation and 10% test data.

### 5.6.1 Results and Evaluation

Results and Evaluations are provided for experiment 3 and 4 separately.

#### Experiment 3

All results are shown in Table 5.18 and precision, recall and f1-score are used as evaluation metrics. It can be seen that the Dilated LSTM with ranked units and an attention layer outperforms both established benchmarks by 21.93% (Maximum Entropy) and 4% (BiLSTM with attention) respectively. This is due to their ability to handle sequential data of variable length, where as the networks' units decrease hierarchically the information is better retained and different timesteps. Of particular interest are the results of the vanilla LSTM as they are considerably below the Maximum Entropy classifiers baseline and the next related model, the Bidirectional LSTM. Taking into account earlier observations that LSTMs may struggle to learn sequences above a certain length given a small dataset, another experiment was conducted, where the sequence length was restricted to 100. This experiment yielded substantially better results with an f1-score of *0.66*. However, this has also meant that over 50% of the documents used in these experiments were cut short and not all information available was utilised.

<b>Model</b>	<b>Precision</b>	<b>Recall</b>	<b>F1-score</b>
Maximum Entropy	0.80	0.63	60.73
LSTM (original sequence length)	0.42	0.41	38.05
LSTM (restricted)	0.69	0.66	66.39
BiLSTM	0.75	0.74	74.21
BiLSTM with attention	0.78	0.77	78.10
Dilated LSTM with attention	0.82	0.81	81.25
DilatedLSTM ranked units	0.83	0.82	82.66

Table 5.18: Results of Experiment 3 using precision, recall and f1-score

In Figures 5.15, 5.16 and 5.17, three confusion matrices are shown, which demonstrate how well the dilated LSTM with ranked units does compared to the baseline and the best comparable model. Firstly, it has to be noted that in all three figures NEU1 posts are most accurately classified, then GSN3 notes and finally DL2 notes.

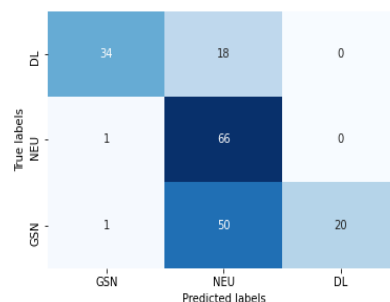


Figure 5.15: Maximum Entropy Classifier

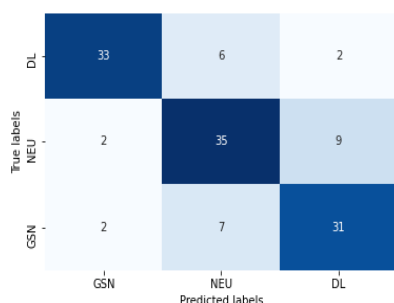


Figure 5.16: Bidirectional LSTM with attention

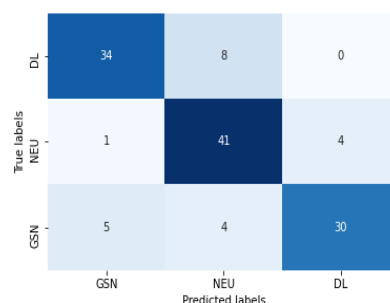


Figure 5.17: Dilated LSTM with ranked units

farewell letter no more joy no more joy no more love no more sun or moon  
to see a little bit nasty just a corpse not very nice for you either reckon  
:UNK: sun gives warmth love strength :UNK: moon is cold and white  
clouds deprive :UNK: sun of its strength but :UNK: night is clear and  
bright ive often dreamt of beautiful things all ive found is smiles if ever  
rebelled i gained nothing just pain and anguish it sounds resigned thats what  
am life has stolen my life away it can all be so simple but went off course  
built up stupid hopes such a pity about my love love was string and  
beautiful but time is stronger it makes you forget may be forgiven for my fit  
of sentimentality wouldnt have made a very good poet

Figure 5.18: Example of a correctly classified GSN3 note

**Linguistic Evaluation** In order to see which features are most important to accurate classification the attention words of the learning model are visualised and examples are shown from the test set of each dataset (see Figures 5.18, 5.19 and 5.20), where words highlighted in darker shades have higher attention weights.

One of the main differences in these three types of documents is the usage of personal pronouns, where in GSN3 notes there is frequent usage of ‘you’, whilst both other documents mainly refer to the first person singular or plural. It can also be seen in Table 5.13 that personal pronouns have a large effect size for GSN3/NEU1 and small effect sizes for GSN3/DL2. There are a range of different topics and emotions

present in each document. Emotions in GSN3 notes are *love*, *joy* and *peacefulness* are present, whilst in DL2 blogs *anger* and *hate* are predominant. Table 5.13 also shows that there are small and medium effect sizes for GSN/DL2 comparisons, but fewer effect sizes for GSN3/NEU1. This can be seen in NEU1 notes, which use less emotionally intense language when discussing topics and seem to talk about multiple aspects of a topic. Furthermore, the DL2 blog mentions suicidal ideation; however, from a linguistic and sentiment perspective, it is clearly distinct from a GSN3 note.

dont know if doing this right but putting it down on paper gets it out of my head at least temporarily and im dying inside im trapped feel so low unwelcome familiar thoughts in my head and nowhere to turn my two teenage kids in bed my bloke away at his kids for weekend want to cry but cant anymore dont want to bring anyone down with my feelings but so lonely at moment life has nothing new to offer me and it seems so easy to leave it all but how can when have two beautiful kids upstairs cant do that to them and thats why feel so low hate life always have feel comfortable in depression after a while and **thats when** get to point of not caring **who** hurt had several episodes through teens and adult life drink used to help unemployment let me be selfish being single mom meant didnt have to explain my feelings but this is first episode since been with partner of 3 years he wont understand and just dread tomorrow when he comes home just want to sleep and never wake up trapped angry im trapped feel selfish for feeling this way and that makes me even angrier any parents out there understand what mean

Figure 5.19

one of the best things about travelling is meeting other backpackers. have met and befriended people from: canada, many states, new zealand, australia, britain, ireland, italy, spain, france, and many more. there are always plenty of stories :UNK: be told **and**, of course, many pints of beer :UNK: bed had. the following story comes from poor max, a teacher from san francisco living and teaching in cairo, egypt. max looks and acts a lot like a u.s. marine, might add. so one day max was in budapest, and a bar owner invited him in for a drink. having a few hours :UNK: kill before heading off on a train, max accepted and had a beer at the bar. soon the bill comes-\$500 u.s. he say, 'this can't be right, only had one beer!' the menacing barman instructs him :UNK: look at the menu where he sees that, indeed, a beer costs 500 dollars. which, might add, is the usual limit for a cash advance off of your credit card. max refuses :UNK: pay, and is promptly escorted :UNK: the basement by burly hungarian men who sit him down and show him pictures of badly beaten up guys who refused :UNK: pay. giving in, said beefy guys accompany him :UNK: the atm nearby and thank him for \$500. needless :UNK: say, max didn't enjoy budapest nearly as much as **did**, **this scam was** actually highlighted in my guide book, and his too, but unfortunately he hadn't gotten around :UNK: reading it. **just realized this might** scare some of you, but promise i'm being careful, not taking creepy invitations, and know that there are risks inherent :UNK: travelling around develeoping former communist bloc nations. the biggest threat here in dubrovnik, however, seems :UNK: be the multitude of lascivious italian men muttering, 'como estai?' under their breath as my friend and walk by. **it was pouring this morning when woke up-after an early night** in, finally, thank goodness-and so windy. do they have hurricanes in croatia? it has cleared up by now though, so we have our bikinis in our bags, ready :UNK: sneak into a beach club. keep the emails coming!

Figure 5.20

### Experiment 4

The results for experiment 4 can all be seen in Table 5.19, where we also use precision, recall and f1-score as an evaluation metric. It can be seen in table 5.19 that the dilated LSTM with ranked units also outperforms the baselines and comparable learning models by more than 10%. Furthermore, we note that when establishing a baseline, using the Maximum Entropy classifier, the f1-score is lower than in experiment 3, which reflects how much harder the task is when using an imbalanced dataset. Using the original sequence length on the LSTM in this experiment also shows that there is improved performance. Overall it can be seen that all neural network approaches outperform the classification results of the baseline and are considerably higher than results from experiment 3. Firstly, this could be due to the increased data size which naturally helps neural networks to perform better and secondly it could also be argued that the different learning models find it easier to classify NEU3 posts due to the imbalance in the dataset.

<b>Model</b>	<b>Precision</b>	<b>Recall</b>	<b>F1-score</b>
Maximum Entropy	0.55	0.67	55.69
LSTM (original sequence length)	0.77	0.71	59.00
LSTM (restricted)	0.78	0.73	64.86
BiLSTM	0.90	0.90	90.43
BiLSTM with attention	0.90	0.90	90.43
Dilated LSTM with attention	0.80	0.81	80.70
DilatedLSTM ranked units	0.96	0.96	96.1

Table 5.19: Results of Experiment 2 using precision, recall and f1-score

Figures 5.22, 5.23 and 5.21 show three confusion matrices comparing the best performing model to the baseline and the best competing model. Overall it can be seen that the baseline model only classifies NEU2 posts correctly and only 1 GSN3 note, whilst it assumes that most DL2 notes are NEU2 posts. When comparing the results of the Bidirectional LSTM with attention to the dilated LSTM with ranked units, it can be seen that the latter is able to also classify both GSN3 and DL2 notes

more often. It could be argued that this is due to the learning model’s ability to access the full sequence length.

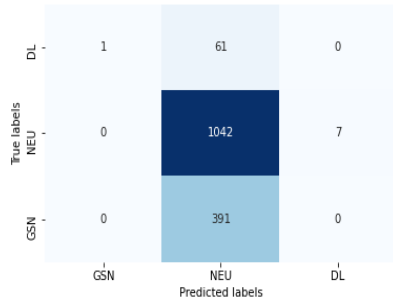


Figure 5.22: Maximum Entropy Classifier

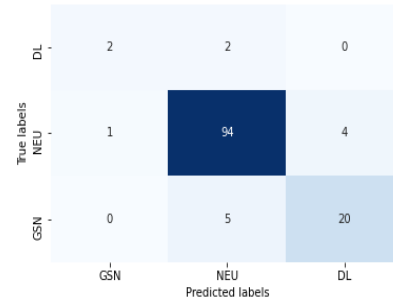


Figure 5.23: Bidirectional LSTM with attention

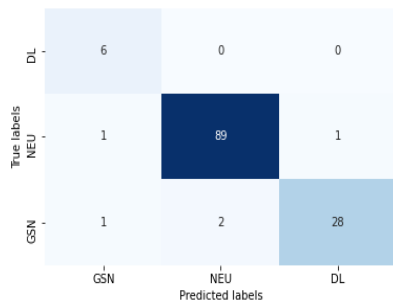


Figure 5.21: Dilated LSTM with ranked units

### Linguistic Evaluation

Figures 5.25, 5.26 and 5.24 all show correctly classified examples of each dataset. It can be seen in the GSN note (see Figure 5.25), where similar to the findings in the linguistic analysis and for the linguistic evaluation in section 5.6.1. Personal pronouns (*‘you’*), positive emotions (*‘love’*) and an increased focus on the present (*‘is’*) seem to be most important for accurate classification. Similarly in DL3 notes (see Figure

5.26) references to death (*‘im dying inside’*) and work (*‘unemployment’*) as well as negative emotions (*‘hate’/‘angry’*) and a increased focus on the past (*‘had’/‘used’*) are assigned the highest attention weights. However, in NEU2 notes (see Figure 5.24) there seem to be less personal pronouns, increased use of adjectives and adverbs (*‘burly’*, *‘beefy’* or *‘creepy’*) and there seem fewer references to emotions. These findings also correspond with the small to large Cohen’s d effect size that was calculated pairwise for each dataset.



dear schnucki, please forgive me for what ive done, but love you so much that cant be without you any more. hope you have a happy life; once im out of your way, you might be a bit more sensible and keep your hands off married men. lay red carnations on my grave just once.

Figure 5.25

:UNK: days are manageable; it's :UNK: years that get me day to day, aside from :UNK: struggle with getting out of bed, can do ok. put :UNK: mask on, survive :UNK: day, go home, go back to sleep (i guess i'm what they call high-functioning). :UNK: idea that i'm going to have to do this forever until die makes me want to move that date up a bit.

Figure 5.26

**Conclusion** This section introduced the Dilated LSTM with ranked units and have shown that the learning model is able to successfully distinguish suicide notes from both depressed blogs and ‘neutral’ blogs. The learning model was tested in two different experimental settings, where it was found that accurate classification of suicide notes was easier when the dataset was balanced. However, it was also found that when using the dilated LSTM with ranked units on an imbalanced dataset that makes the overall task more realistic, it was able to identify more suicide notes compared to other learning models. The learning model outperforms the baseline of 60.73%, when using F1-score for evaluation in experiment 3 and achieves and F1-score of 96.1% in experiment 4. Furthermore it was shown that it is possible to achieve better results when significantly reducing the sequence length in a standard LSTM on a small dataset in experiment 3. Therefore demonstrating that accurate classification is possible solely on linguistic patterns in this type of textual data. Therefore these linguistic differences could substantially contribute to future analysis of mental health issues online. Finally, it was shown by visualising attention weights which words are most important to each text category and the results were compared to the findings of Cohen’s effect size in section 5.3.3.

in 1913, :UNK: first direct federal-income-tax was imposed upon your pocketbook by congress, this motion succeeded only after :UNK: much debated 16th amendment had been ratified and accepted by :UNK: states. (an action that is now thought to have been unlawfully executed!) anyhow, :UNK: original income tax was incredibly low--only a mere, yet principally wrong, one-percent! moreover, this tax was only applicable to incomes in excess of \$20,000(1913), which calculates out to a little under \$400,000 in 2004, after being adjusted for inflation. suprising? thought so. :UNK: founding fathers had forseen :UNK: possibility of congress instating taxes for essential purposes, yet they were horrified at :UNK: suggestion of a massachusetts senator who implied that congress may, at one time, abuse :UNK: constitutonal ability to lay taxes(section 8, clause 1) by levying a direct income tax. additionally, when that senator suggested that :UNK: aforesaid tax could reach levels as high as 20%, he was flaty refuted by alexander hamilton who angrily stated that no elected congress could possibly be so lacking in fairness, justice and patriotism. nevertheless, we have seen :UNK: feds pulling up sixty-five percent of your annual paycheck! (1970's) so, what all this really means is this: every year, hundreds of millions of americans citizens are financially plundered by :UNK: very governing body they elected to office. unless you make over 370 grand a year, your filing 1040's that our founding fathers would riot over, your welcome. have a great day. :)

Figure 5.24: Example of a correctly classified NEU2 note

## 5.7 Conclusion and future work

In this chapter, a series of four Experiments have been conducted to classify suicide notes from a variety of other types of notes and posts. In Experiment 1, the task of classifying suicide notes using recurrent neural networks was introduced, where the main findings were that sentiment features are hugely important for accurate classification and that the sequence of sentiments also matters.

Experiment 2 looked at applying the previously introduced dilated LSTM with attention for a document classification task for the first time. The task in this Experiment was to distinguish suicide notes from last statements and depression notes. Furthermore, a linguistic analysis of the datasets was conducted using LIWC, and several features were outlined that make each dataset unique. Then the outputs of the neural networks attention weights were visualised, where it was found that the dilated LSTM with attention assigns high attention weights to words and concepts that are also important in the linguistic analysis. Finally, an error analysis was conducted to show which notes were wrongly classified and which features were assigned high attention weights in those notes. Overall, it was found that GSN2 notes fall between the two extremes of LS and DL1 notes, where LS notes tend to share many close features with GSN2 notes.

In Experiment 3 and 4, a new task was introduced that aimed at classifying suicide notes from ‘neutral’ data, where several random blog posts were chosen for the ‘neutral’ category. Then the two Experiments were conducted in two different settings: (i) using a balanced dataset and (ii) an unbalanced dataset. This section also introduced the dilated LSTM with ranked units, which was created to overcome the issues of variable sequence length in the different datasets. Additionally, another linguistic analysis was conducted, and statistical significance testing was introduced to show the pairwise effect size of each feature found in the linguistic analysis.

Using Cohen's  $d$  effect size, it was found that many features that showed a medium or large effect size were also important to the accurate classification of the data. This was shown by visualising attention weights for all three datasets. Furthermore, it was found that the NEU1/NEU2 datasets are considerably different in terms of the sentiment expressed in them and other linguistic features.

Overall, it has to be pointed out that most of the work in this chapter has focused on the suitability of RNNs for these tasks and any future work should also investigate the use of other deep learning methods (e.g., CNNs, Attention and Graph Neural Networks) as well as state-of-the-art embedding representations (eg.: BERT, ELMO and ERNIE). Furthermore, recent advances in mental health research for text data has found that knowledge graphs can be useful in classifying the different phases of bipolar disorders. Therefore, it would be interesting to see how knowledge-graphs could further suicide note classification research.

Finally, this chapter has introduced the only publicly available dataset of genuine suicide notes at this point. However, more data is needed in order to be able to generalise this kind of research further. There is also a need for controlling other attributes of those who die by suicide, such as age or gender because most research is currently conducted using public datasets that contain notes from white, American men. This makes it harder to generalise topic or sentiments expressed in the data to other groups of people, such as women or transgender people, where there is a significant rise in suicide for the latter. Provided this type of fairness and bias can be considered in future work and resources can be shared in a responsible manner this type of research could have real-world impact and help to achieve the SGD goal of reducing suicide rates. Given further research is conducted such work could be useful in some scenarios, including but not limited to assessing the seriousness of a social media post or suicide attempt in a clinical setting.

# Chapter 6

## Conclusion

This final chapter summarises the contributions of this work, addresses its limitations and outlines avenues for future research.

### 6.1 Summary of contributions

This research had two overarching goals, where firstly it aimed to address the issue of exploding and vanishing gradient descent in RNNs for the task of fine-grained emotion classification on tweets. This also included evaluating and applying the proposed solution to a document-level suicide note classification task. Secondly, it was argued that most current knowledge resources for SA tasks either lack in emotion granularity (e.g., large resources only using polarities) or overall size (e.g., resources that lack coverage, but use fine-grained emotions). Furthermore, it aimed to show that including knowledge into learning embedding representations will lead to better performance on the same task.

- Based on the results in Chapter 3, it was established that tweets should no longer be considered short sequences or sentence-level SA. This was done by

showing how classification performance drops significantly when not resorting to techniques such as sequence pruning to achieve better performance. Then a new learning model, the bidirectional dilated LSTM with attention was introduced to overcome the aforementioned problem in learning long sequences using RNNs (vanishing and exploding gradient descent). The results show how the learning model takes advantage of the overall sequence length of a tweet and outperforms other competitive RNN- and CNN-based learning models. Furthermore, Chapter 3 outlined the task of fine-grained emotion classification based on Ekman's six basic emotions (Ekman 1999) to help overcome the issue of relying on a costly human annotation to generate new datasets. Furthermore, it also showed how humans perform on the same task using annotators from the AMT (Turk 2012) platform. There it was found that the learning models performed better than humans on the task.

- The results in Chapter 5 indicate that the proposed learning model in Chapter 3, can be applied to the task of suicide note classification. Furthermore, this chapter identified how important sentiment and linguistic features are for accurate classification of suicide notes in the tasks of distinguishing them from depressed notes and love notes, depressed notes and last statements as well as depressed posts and random blog posts. It also provided a qualitative analysis of visualisations that showed which features are important to accurate classification using neural networks, where it was found that the proposed learning models match the features found during the linguistic analysis. Finally, this chapter introduced the task of detecting suicide notes in real-world scenarios.
- Chapter 4 saw the introduction of a new resource (RELATE) and sentiment-specific word embeddings. RELATE addressed the issue of having only knowledge resources that either lack fine-grained emotion knowledge

or size. It was built on Twitter data, includes emotion keywords for Ekman's six basic emotions and incorporates linguistic rules. The embedding representations were generated by a Graph Convolutional Neural Network and tested on the task of fine-grained emotion classification. Therefore, this chapter also showed how embedding representations that incorporate both context and sentiment can further improve classification results. Furthermore, a qualitative analysis was conducted that showed how RELATE performs in comparison to GloVe. There it was shown that GloVe incorporates more standard language in its representations and RELATE also uses colloquial/non-standard language. Finally, it was argued that any language model used for real world tasks should be evaluated against the bias it introduced in order to avoid more inequality and discrimination in AI-based tools.

## 6.2 Limitations and future work

- Chapter 3 proposed a new dataset for fine-grained emotion classification and a new method to overcome the issues of vanishing and exploding gradient descent. However, there are some limitations to this, and future work should consider the following avenues of research. Firstly, any proposed RNN learning model should be evaluated on a number of different long sequence learning tasks. It should be more carefully considered when new methods are proposed if they work for the whole field of SA. This is because as a field, SA has so many different subtasks and approaches (Cambria et al. 2019). Recent advances in NLP and the rise of attention-based neural networks also raise the question to which extent RNN-based learning models should be the only or best solution to sequences learning tasks. Additionally, it should also be considered to not only further increase the dataset size, but also further increase the number of

emotion categories by using Plutchik's emotion model (Plutchik 2001). While getting more data is a valid future path, it should also be considered how this would be possible given that most data is monopolised and monetised by large corporations. Therefore, for any future work, its impact and applicability is limited by default to those who are in the position to access the data. This also means that any findings or models developed are inherently biased due to the aforementioned situation and arguably any work tackling discrimination or equality can only be fully carried out by large companies. Finally, it would be interesting to investigate why neural network based learning models are so much better at classifying tweets into emotion categories compared to humans.

- Chapter 5 demonstrated in 4 different experiment series that it is possible to use neural network approaches for suicide note classification. Furthermore it showed in its linguistic analysis, which features are important for distinguishing suicide notes from other types of posts and notes. There are two main limitations to this work which should be addressed in any future work. Firstly, the datasets need to be increased in order to validate any findings further and make the task even more applicable to a real-world scenario. Secondly, other factors such as the note writers' information need to be more carefully considered, because this could lead to unfairly generalising findings to multiple groups of people. Furthermore, it should also be considered whether 'AI for social good' is a title or term that should be adopted by the field of AI, given that the tasks tackled in it are more often than not also political issues (e.g.: reducing poverty or mental health conditions). It also leaves discussion open to what is 'good' and whether tasks such as automatic face recognition for governments really fall under that term.
- In Chapter 4 a new knowledge resource was introduced and used as input to learning embedding representations using a Graph Convolutional Neural

Network. While this addresses the initial hypothesis and proposes a working solution, several future areas of work need to be considered. Firstly, this work could be taken further by increasing the size of RELATE, linking it to a larger knowledge resource and evaluating it on other learning tasks. Also, RELATE could be evaluated against other, albeit smaller or less granular resources, on the same task. Finally, there is much room for future work in the area of developing learning models that are able to pick up on non-standard and colloquial language in social media data. Furthermore, it is important to note that RELATE specifically only covers the English language and therefore cannot be applied or transferred to non-English domains or tasks. As shown in Chapter 4, RELATE clearly shows bias in its embedding representations, not unlike many other large LMs. Therefore it is important to investigate in any future work how to reduce bias in order to create LMs that help to create more equality and reduce discrimination. Furthermore, the learned embedding representations should be further evaluated on more advanced learning models (e.g., dilated LSTM with attention) and applied to other SA tasks. Finally, it would also be interesting to see if this approach would generalise over other domains outside of SA.

Based on the quantitative results described in this thesis, it can be concluded that the proposed methods and approaches do confirm the initial hypotheses as outlined in section 1. However, it also leaves room for important discussions specifically around the applicability of RNNs to sequence classification problems, bias in embedding representations and what ‘AI for social good’ means.



# Bibliography

- Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M. et al. (2016), Tensorflow: a system for large-scale machine learning., *in* ‘OSDI’, Vol. 16, pp. 265–283.
- Abdul-Mageed, M. & Ungar, L. (2017), Emonet: Fine-grained emotion detection with gated recurrent neural networks, *in* ‘Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)’, Vol. 1, pp. 718–728.
- Agarwal, A., Xie, B., Vovsha, I., Rambow, O. & Passonneau, R. (2011), Sentiment analysis of twitter data, *in* ‘Proceedings of the workshop on languages in social media’, Association for Computational Linguistics, pp. 30–38.
- Akhtar, M. S., Kumar, A., Ekbal, A. & Bhattacharyya, P. (2016), A hybrid deep learning architecture for sentiment analysis., *in* ‘COLING’, pp. 482–493.
- Allen, C., Balazevic, I. & Hospedales, T. (2019), What the vec? towards probabilistically grounded embeddings, *in* ‘Advances in Neural Information Processing Systems’, pp. 7465–7475.
- Alm, C. O., Roth, D. & Sproat, R. (2005), Emotions from text: machine learning for text-based emotion prediction, *in* ‘Proceedings of human language technology conference and conference on empirical methods in natural language processing’, pp. 579–586.

- Alvarez, A. (2002), *The savage god: A study of suicide*, A&C Black.
- Aman, S. & Szpakowicz, S. (2007), Identifying expressions of emotion in text, *in* ‘International Conference on Text, Speech and Dialogue’, Springer, pp. 196–205.
- Amazon (2018a), ‘Alexa and alexa devices faq’, <https://www.amazon.co.uk/gp/help/customer/display.html>. Accessed on 2018-10-01.
- Amazon (2018b), ‘Mechanical turk’, <https://www.mturk.com>. Accessed on 2017-01-10.
- Amodei, D., Ananthanarayanan, S., Anubhai, R., Bai, J., Battenberg, E., Case, C., Casper, J., Catanzaro, B., Cheng, Q., Chen, G. et al. (2016), Deep speech 2: End-to-end speech recognition in english and mandarin, *in* ‘International Conference on Machine Learning’, pp. 173–182.
- Anthology, A. (2020), ‘Workshop on computational approaches to subjectivity and sentiment analysis (wassa)’.
- URL:** <https://www.aclweb.org/anthology/venues/wassa/>
- Appel, O., Chiclana, F., Carter, J. & Fujita, H. (2016), ‘A hybrid approach to the sentiment analysis problem at the sentence level’, *Knowledge-Based Systems* **108**, 110–124.
- Arik, S. O., Chrzanowski, M., Coates, A., Diamos, G., Gibiansky, A., Kang, Y., Li, X., Miller, J., Ng, A., Raiman, J. et al. (2017), ‘Deep voice: Real-time neural text-to-speech’, *arXiv preprint arXiv:1702.07825*.
- Arjovsky, M., Shah, A. & Bengio, Y. (2016), Unitary evolution recurrent neural networks, *in* ‘International Conference on Machine Learning’, pp. 1120–1128.
- Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R. & Ives, Z. (2007), Dbpedia: A nucleus for a web of open data, *in* ‘The semantic web’, Springer, pp. 722–735.

- Augenstein, I., Padó, S. & Rudolph, S. (2012), Lodifier: Generating linked data from unstructured text, *in* ‘Extended Semantic Web Conference’, Springer, pp. 210–224.
- Averill, J. R. (1980), A constructivist view of emotion, *in* ‘Theories of emotion’, Elsevier, pp. 305–339.
- Baccianella, S., Esuli, A. & Sebastiani, F. (2010), Sentiwordnet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining., *in* ‘LREC’, Vol. 10, pp. 2200–2204.
- Bahdanau, D., Cho, K. & Bengio, Y. (2014), ‘Neural machine translation by jointly learning to align and translate’, *arXiv preprint arXiv:1409.0473* .
- Bahdanau, D., Cho, K. & Bengio, Y. (2015), Neural Machine Translation by Jointly Learning to Align and Translate, *in* ‘Proc. of the International Conference on Learning Representations (ICLR)’, San Diego, CA, USA.
- Baker, M. C. (2003), *Lexical categories: Verbs, nouns and adjectives*, Vol. 102, Cambridge University Press.
- Balabantaray, R. C., Mohammad, M. & Sharma, N. (2012), ‘Multi-class twitter emotion classification: A new approach’, *International Journal of Applied Information Systems* 4(1), 48–53.
- Balahur, A., Boldrini, E., Montoyo, A. & Martinez-Barco, P., eds (2011), *Proceedings of the 2nd Workshop on Computational Approaches to Subjectivity and Sentiment Analysis (WASSA 2.011)*, Association for Computational Linguistics, Portland, Oregon.  
**URL:** <https://www.aclweb.org/anthology/W11-1700>
- Balahur, A., Klinger, R., Hoste, V., Strapparava, C. & De Clercq, O., eds (2019), *Proceedings of the Tenth Workshop on Computational Approaches to Subjectivity,*

*Sentiment and Social Media Analysis*, Association for Computational Linguistics, Minneapolis, USA.

**URL:** <https://www.aclweb.org/anthology/W19-1300>

Balahur, A., Mohammad, S. M., Hoste, V. & Klinger, R., eds (2018), *Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, Association for Computational Linguistics, Brussels, Belgium.

**URL:** <https://www.aclweb.org/anthology/W18-6200>

Balahur, A., Mohammad, S. M. & van der Goot, E., eds (2017), *Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, Association for Computational Linguistics, Copenhagen, Denmark.

**URL:** <https://www.aclweb.org/anthology/W17-5200>

Balahur, A., van der Goot, E. & Montoyo, A., eds (2013), *Proceedings of the 4th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, Association for Computational Linguistics, Atlanta, Georgia.

**URL:** <https://www.aclweb.org/anthology/W13-1600>

Barrett, L. F. (2009), 'Variety is the spice of life: A psychological construction approach to understanding variability in emotion', *Cognition and Emotion* **23**(7), 1284–1306.

Barrett, L. F., Lewis, M. & Haviland-Jones, J. M. (2016), *Handbook of emotions*, Guilford Publications.

Bartle, A. & Zheng, J. (2015), Gender classification with deep learning, in 'Technical report', The Stanford NLP Group.

Baziotis, C., Pelekis, N. & Doulkeridis, C. (2017a), Datastories at semeval-2017 task 4: Deep lstm with attention for message-level and topic-based sentiment analysis,

- in* ‘Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)’, pp. 747–754.
- Baziotis, C., Pelekis, N. & Doulkeridis, C. (2017*b*), Datastories at semeval-2017 task 4: Deep lstm with attention for message-level and topic-based sentiment analysis, *in* ‘Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)’, Association for Computational Linguistics, Vancouver, Canada, pp. 747–754.
- BBC (2019), ‘Facebook ‘sorry’ for distressing suicide posts on instagram’.  
**URL:** <https://www.bbc.co.uk/news/uk-46976753>
- Ben-Zeev, A. & Ben-Zeev, A. (2001), *The subtlety of emotions*, MIT press.
- Bengio, Y., Ducharme, R., Vincent, P. & Jauvin, C. (2003), ‘A neural probabilistic language model’, *Journal of machine learning research* **3**(Feb), 1137–1155.
- Bengio, Y., Simard, P., Frasconi, P. et al. (1994), ‘Learning long-term dependencies with gradient descent is difficult’, *IEEE transactions on neural networks* **5**(2), 157–166.
- Benton, A., Mitchell, M. & Hovy, D. (2017), ‘Multi-task learning for mental health using social media text’, *arXiv preprint arXiv:1712.03538* .
- Bhatt, A., Patel, A., Chheda, H. & Gawande, K. (2015), ‘Amazon review classification and sentiment analysis’, *International Journal of Computer Science and Information Technologies* **6**(6), 5107–5110.
- Binali, H., Wu, C. & Potdar, V. (2010), Computational approaches for emotion detection in text, *in* ‘4th IEEE International Conference on Digital Ecosystems and Technologies’, IEEE, pp. 172–177.
- Bird, S. (2006), Nltk: the natural language toolkit, *in* ‘Proceedings of

- the COLING/ACL on Interactive presentation sessions', Association for Computational Linguistics, pp. 69–72.
- Bird, S. & Loper, E. (2004), Nltk: the natural language toolkit, *in* 'Proceedings of the ACL 2004 on Interactive poster and demonstration sessions', Association for Computational Linguistics, p. 31.
- Boals, A. & Klein, K. (2005), 'Word use in emotional narratives about failed romantic relationships and subsequent mental health', *Journal of Language and Social Psychology* **24**(3), 252–268.
- Bollacker, K., Evans, C., Paritosh, P., Sturge, T. & Taylor, J. (2008), Freebase: a collaboratively created graph database for structuring human knowledge, *in* 'Proceedings of the 2008 ACM SIGMOD international conference on Management of data', pp. 1247–1250.
- Bond, G. D. & Lee, A. Y. (2005), 'Language of lies in prison: Linguistic classification of prisoners' truthful and deceptive natural language', *Applied Cognitive Psychology* **19**(3), 313–329.
- Bordes, A., Chopra, S. & Weston, J. (2014), Question answering with subgraph embeddings, *in* 'Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)', pp. 615–620.
- Bordes, A. & Gabrilovich, E. (2014), Constructing and mining web-scale knowledge graphs: Kdd 2014 tutorial, *in* 'Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining', pp. 1967–1967.
- Bordes, A., Glorot, X., Weston, J. & Bengio, Y. (2012), Joint learning of words and meaning representations for open-text semantic parsing, *in* 'Artificial Intelligence and Statistics', pp. 127–135.

- Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J. & Yakhnenko, O. (2013), Translating embeddings for modeling multi-relational data, *in* ‘Advances in neural information processing systems’, pp. 2787–2795.
- Bordes, A., Weston, J., Collobert, R. & Bengio, Y. (2011), Learning structured embeddings of knowledge bases, *in* ‘Twenty-Fifth AAAI Conference on Artificial Intelligence’.
- Bordes, A., Weston, J. & Usunier, N. (2014), Open question answering with weakly supervised embedding models, *in* ‘Joint European conference on machine learning and knowledge discovery in databases’, Springer, pp. 165–180.
- Bosselut, A., Rashkin, H., Sap, M., Malaviya, C., Celikyilmaz, A. & Choi, Y. (2019), ‘Comet: Commonsense transformers for automatic knowledge graph construction’, *arXiv preprint arXiv:1906.05317*.
- Bottou, L. (2010), Large-scale machine learning with stochastic gradient descent, *in* ‘Proceedings of COMPSTAT’2010’, Springer, pp. 177–186.
- Bouazizi, M. & Ohtsuki, T. (2016), Sentiment analysis: From binary to multi-class classification: A pattern-based approach for multi-class sentiment analysis in twitter, *in* ‘2016 IEEE International Conference on Communications (ICC)’, IEEE, pp. 1–6.
- Brants, T., Chen, F. & Tsochantaridis, I. (2002), Topic-based document segmentation with probabilistic latent semantic analysis, *in* ‘Proceedings of the eleventh international conference on Information and knowledge management’, ACM, pp. 211–218.
- Bravo-Marquez, F., Mendoza, M. & Poblete, B. (2014), ‘Meta-level sentiment models for big social data analysis’, *Knowledge-Based Systems* **69**, 86–99.

- Brooks, M., Kuksenok, K., Torkildson, M. K., Perry, D., Robinson, J. J., Scott, T. J., Anicello, O., Zukowski, A., Harris, P. & Aragon, C. R. (2013), Statistical affect detection in collaborative chat, *in* 'Proceedings of the 2013 conference on Computer supported cooperative work', ACM, pp. 317–328.
- Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A. et al. (2020), 'Language models are few-shot learners', *arXiv preprint arXiv:2005.14165* .
- Bückner, J., Pahl, M., Stahlhut, O. & Liedtke, C.-E. (2002), A knowledge-based system for context dependent evaluation of remote sensing data, *in* 'Joint Pattern Recognition Symposium', Springer, pp. 58–65.
- Buechel, S., Rücker, S. & Hahn, U. (2020), 'Learning and evaluating emotion lexicons for 91 languages', *arXiv preprint arXiv:2005.05672* .
- Burnap, P., Colombo, G., Amery, R., Hodorog, A. & Scourfield, J. (2017), 'Multi-class machine classification of suicide-related communication on twitter', *Online social networks and media* **2**, 32–44.
- Burnap, P., Colombo, W. & Scourfield, J. (2015), Machine classification and analysis of suicide-related communication on twitter, *in* 'Proceedings of the 26th ACM conference on hypertext & social media', ACM, pp. 75–84.
- Busch, J. E., Lin, A. D., Graydon, P. J. & Caudill, M. (2006), 'Ontology-based parser for natural language processing'. US Patent 7,027,974.
- Calvo, R. A., D'Mello, S., Gratch, J. & Kappas, A. (2015), *The Oxford handbook of affective computing*, Oxford Library of Psychology.
- Calvo, R. A., Milne, D. N., Hussain, M. S. & Christensen, H. (2017), 'Natural language processing in mental health applications using non-clinical texts', *Natural Language Engineering* **23**(5), 649–685.



- Cambria, E. (2016), ‘Affective computing and sentiment analysis’, *IEEE Intelligent Systems* **31**(2), 102–107.
- Cambria, E., Das, D., Bandyopadhyay, S. & Feraco, A. (2017), *A practical guide to sentiment analysis*, Vol. 5, Springer.
- Cambria, E., Fu, J., Bisio, F. & Poria, S. (2015), Affectivespace 2: Enabling affective intuition for concept-level sentiment analysis, in ‘Twenty-ninth AAAI conference on artificial intelligence’.
- Cambria, E., Poria, S., Bajpai, R. & Schuller, B. (2016), Senticnet 4: A semantic resource for sentiment analysis based on conceptual primitives, in ‘Proceedings of COLING 2016, the 26th international conference on computational linguistics: Technical papers’, pp. 2666–2677.
- Cambria, E., Poria, S., Gelbukh, A. & Thelwall, M. (2017), ‘Sentiment analysis is a big suitcase’, *IEEE Intelligent Systems* **32**(6), 74–80.
- Cambria, E., Poria, S., Hazarika, D. & Kwok, K. (2018), Senticnet 5: discovering conceptual primitives for sentiment analysis by means of context embeddings, in ‘AAAI’.
- Cambria, E., Poria, S., Hussain, A. & Liu, B. (2019), ‘Computational intelligence for affective computing and sentiment analysis [guest editorial]’, *IEEE Computational Intelligence Magazine* **14**(2), 16–17.
- Cambria, E., Speer, R., Havasi, C. & Hussain, A. (2010), Senticnet: A publicly available semantic resource for opinion mining, in ‘2010 AAAI Fall Symposium Series’.
- Cambria, E. & White, B. (2014), ‘Jumping nlp curves: A review of natural language processing research’, *IEEE Computational intelligence magazine* **9**(2), 48–57.

- Caragea, C., McNeese, N., Jaiswal, A., Traylor, G., Kim, H.-W., Mitra, P., Wu, D., Tapia, A. H., Giles, L., Jansen, B. J. et al. (2011), Classifying text messages for the haiti earthquake, *in* ‘Proceedings of the 8th international conference on information systems for crisis response and management (ISCRAM2011)’, Citeseer.
- Carlson, A., Betteridge, J., Kisiel, B., Settles, B., Hruschka, E. R. & Mitchell, T. M. (2010), Toward an architecture for never-ending language learning, *in* ‘Twenty-Fourth AAAI Conference on Artificial Intelligence’.
- Cattoni, R., Corcoglioni, F., Girardi, C., Magnini, B., Serafini, L. & Zanolini, R. (2012), The knowledgestore: an entity-based storage system., *in* ‘LREC’, Citeseer, pp. 3639–3646.
- Chaffey, D. (2016), ‘Global social media research summary 2016’, *Smart Insights: Social Media Marketing* .
- Chami, I., Wolf, A., Juan, D.-C., Sala, F., Ravi, S. & Ré, C. (2020), ‘Low-dimensional hyperbolic knowledge graph embeddings’, *arXiv preprint arXiv:2005.00545* .
- Champoux, V., Durgee, J. & McGlynn, L. (2012), ‘Corporate facebook pages: when “fans” attack’, *Journal of Business Strategy* .
- Chancellor, S. & De Choudhury, M. (2020), ‘Methods in predictive techniques for mental health status on social media: a critical review’, *NPJ digital medicine* **3**(1), 1–11.
- Chang, S., Zhang, Y., Han, W., Yu, M., Guo, X., Tan, W., Cui, X., Witbrock, M., Hasegawa-Johnson, M. A. & Huang, T. S. (2017), Dilated recurrent neural networks, *in* ‘Advances in Neural Information Processing Systems’, pp. 77–87.

- Chekol, M. W., Pirrò, G., Schoenfish, J. & Stuckenschmidt, H. (2017), Marrying uncertainty and time in knowledge graphs, *in* ‘Thirty-First AAAI Conference on Artificial Intelligence’.
- Chen, D. & Manning, C. (2014), A fast and accurate dependency parser using neural networks, *in* ‘Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)’, pp. 740–750.
- Chen, L., Aldayel, A., Bogoychev, N. & Gong, T. (2019), Similar minds post alike: Assessment of suicide risk using a hybrid model, *in* ‘Proceedings of the Sixth Workshop on Computational Linguistics and Clinical Psychology’, pp. 152–157.
- Chen, M., Mao, S. & Liu, Y. (2014), ‘Big data: A survey’, *Mobile networks and applications* **19**(2), 171–209.
- Chen, Z., Zha, H., Liu, H., Chen, W., Yan, X. & Su, Y. (2019), ‘Global textual relation embedding for relational understanding’, *arXiv preprint arXiv:1906.00550* .
- Cheng, J., Dong, L. & Lapata, M. (2016), ‘Long short-term memory-networks for machine reading’, *arXiv preprint arXiv:1601.06733* .
- Cherry, C., Mohammad, S. M. & De Bruijn, B. (2012), ‘Binary classifiers and latent sequence models for emotion detection in suicide notes’, *Biomedical informatics insights* **5**, BII-S8933.
- Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H. & Bengio, Y. (2014), ‘Learning phrase representations using rnn encoder-decoder for statistical machine translation’, *arXiv preprint arXiv:1406.1078* .
- Choi, E., Bahadori, M. T., Schuetz, A., Stewart, W. F. & Sun, J. (2016), Doctor ai:

- Predicting clinical events via recurrent neural networks, *in* ‘Machine Learning for Healthcare Conference’, pp. 301–318.
- Chollet, F. et al. (2015), ‘Keras’.
- Chowdhury, G. G. (2003), ‘Natural language processing’, *Annual review of information science and technology* **37**(1), 51–89.
- Chung, J., Ahn, S. & Bengio, Y. (2016), ‘Hierarchical multiscale recurrent neural networks’, *arXiv preprint arXiv:1609.01704* .
- Chung, J., Gulcehre, C., Cho, K. & Bengio, Y. (2014), ‘Empirical evaluation of gated recurrent neural networks on sequence modeling’, *arXiv preprint arXiv:1412.3555* .
- Chung, J., Gulcehre, C., Cho, K. & Bengio, Y. (2015), Gated feedback recurrent neural networks, *in* ‘International Conference on Machine Learning’, pp. 2067–2075.
- Clark, K., Khandelwal, U., Levy, O. & Manning, C. D. (2019), ‘What does bert look at? an analysis of bert’s attention’, *arXiv preprint arXiv:1906.04341* .
- Cohan, A., Desmet, B., Yates, A., Soldaini, L., MacAvaney, S. & Goharian, N. (2018), ‘Smhd: A large-scale resource for exploring online language usage for multiple mental health conditions’, *arXiv preprint arXiv:1806.05258* .
- Cohen, J. (2013), *Statistical power analysis for the behavioral sciences*, Academic press.
- Cohn, M. A., Mehl, M. R. & Pennebaker, J. W. (2004), ‘Linguistic markers of psychological change surrounding september 11, 2001’, *Psychological science* **15**(10), 687–693.

- Coppersmith, G., Dredze, M. & Harman, C. (2014), Quantifying mental health signals in twitter, *in* 'Proceedings of the Workshop on Computational Linguistics and Clinical Psychology: From Linguistic Signal to Clinical Reality', pp. 51–60.
- Coppersmith, G., Dredze, M., Harman, C., Hollingshead, K. & Mitchell, M. (2015), Clpsych 2015 shared task: Depression and ptsd on twitter, *in* 'Proceedings of the 2nd Workshop on Computational Linguistics and Clinical Psychology: From Linguistic Signal to Clinical Reality', pp. 31–39.
- Coppersmith, G., Leary, R., Crutchley, P. & Fine, A. (2018), 'Natural language processing of social media as screening for suicide risk', *Biomedical informatics insights* **10**, 1178222618792860.
- Coppin, G. & Sander, D. (2016*a*), Theoretical approaches to emotion and its measurement, *in* 'Emotion measurement', Elsevier, pp. 3–30.
- Coppin, G. & Sander, D. (2016*b*), Theoretical approaches to emotion and its measurement, *in* H. L. Meiselman, ed., 'Emotion Measurement', Woodhead Publishing, Oxford, chapter 1, pp. 3–30.
- Coulthard, M., Johnson, A. & Wright, D. (2016), *An introduction to forensic linguistics: Language in evidence*, Routledge.
- CrowdFlower (2018), 'Crowdfower', <https://www.crowdfower.com>. Accessed on 2018-10-01.
- Crystal, D. & McLachlan, E. (2004), *Rediscover grammar*, Longman.
- Cummings, H. W. & Renshaw, S. L. (1979), 'Slca iii: A metatheoretic approach to the study of language', *Human Communication Research* **5**(4), 291–300.
- Cunha, T. O., Weber, I., Haddadi, H. & Pappa, G. L. (2016), The effect of social feedback in a reddit weight loss community, *in* 'Proceedings of the 6th International Conference on Digital Health Conference', pp. 99–103.

- Dai, Z., Yang, Z., Yang, Y., Carbonell, J., Le, Q. V. & Salakhutdinov, R. (2019), ‘Transformer-xl: Attentive language models beyond a fixed-length context’, *arXiv preprint arXiv:1901.02860* .
- Das, R., Munkhdalai, T., Yuan, X., Trischler, A. & McCallum, A. (2018), ‘Building dynamic knowledge graphs from text using machine reading comprehension’, *arXiv preprint arXiv:1810.05682* .
- Dauphin, Y. N., Fan, A., Auli, M. & Grangier, D. (2017), Language modeling with gated convolutional networks, *in* ‘Proceedings of the 34th International Conference on Machine Learning-Volume 70’, JMLR. org, pp. 933–941.
- Death Penalty Information Centre (2019), ‘Time on death row’.  
**URL:** <https://deathpenaltyinfo.org/time-death-row>
- Demszky, D., Movshovitz-Attias, D., Ko, J., Cowen, A., Nemade, G. & Ravi, S. (2020), ‘Goemotions: A dataset of fine-grained emotions’, *arXiv preprint arXiv:2005.00547* .
- Denecke, K. & Deng, Y. (2015), ‘Sentiment analysis in medical settings: New opportunities and challenges’, *Artificial intelligence in medicine* **64**(1), 17–27.
- Deng, Y., Stoehr, M. & Denecke, K. (2014), Retrieving attitudes: Sentiment analysis from clinical narratives., *in* ‘MedIR@ SIGIR’, pp. 12–15.
- Desmet, B. & Hoste, V. (2013), ‘Emotion detection in suicide notes’, *Expert Systems with Applications* **40**(16), 6351–6358.
- Dettmers, T., Minervini, P., Stenetorp, P. & Riedel, S. (2018), Convolutional 2d knowledge graph embeddings, *in* ‘Thirty-Second AAAI Conference on Artificial Intelligence’.
- Devlin, J., Chang, M.-W., Lee, K. & Toutanova, K. (2018), ‘Bert: Pre-training

- of deep bidirectional transformers for language understanding’, *arXiv preprint arXiv:1810.04805* .
- Dhingra, B., Jin, Q., Yang, Z., Cohen, W. W. & Salakhutdinov, R. (2018), ‘Neural models for reasoning over multiple mentions using coreference’, *arXiv preprint arXiv:1804.05922* .
- Distiawan, B., Weikum, G., Qi, J. & Zhang, R. (2019), Neural relation extraction for knowledge base enrichment, *in* ‘Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics’, pp. 229–240.
- Dong, L., Wei, F., Tan, C., Tang, D., Zhou, M. & Xu, K. (2014), Adaptive recursive neural network for target-dependent twitter sentiment classification, *in* ‘Proceedings of the 52nd annual meeting of the association for computational linguistics (volume 2: Short papers)’, Vol. 2, pp. 49–54.
- Dong, X., Gabrilovich, E., Heitz, G., Horn, W., Lao, N., Murphy, K., Strohmann, T., Sun, S. & Zhang, W. (2014), Knowledge vault: A web-scale approach to probabilistic knowledge fusion, *in* ‘Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining’, pp. 601–610.
- Dos Santos, C. N. & Gatti, M. (2014), Deep convolutional neural networks for sentiment analysis of short texts., *in* ‘COLING’, pp. 69–78.
- Dragoni, M. & Petrucci, G. (2017), ‘A neural word embeddings approach for multi-domain sentiment analysis’, *IEEE Transactions on Affective Computing* **8**(4), 457–470.
- Dragoni, M., Poria, S. & Cambria, E. (2018), ‘Ontosenticnet: A commonsense ontology for sentiment analysis’, *IEEE Intelligent Systems* **33**(3), 77–85.
- Duque, A. B., Santos, L. L. J., Macêdo, D. & Zanchettin, C. (2019), Squeezed

- very deep convolutional neural networks for text classification, *in* ‘International Conference on Artificial Neural Networks’, Springer, pp. 193–207.
- Eaton, J. & Theuer, A. (2009), ‘Apology and remorse in the last statements of death row prisoners’, *Justice Quarterly* **26**(2), 327–347.
- Eck, D. & Schmidhuber, J. (2002), Learning the long-term structure of the blues, *in* ‘International Conference on Artificial Neural Networks’, Springer, pp. 284–289.
- Ehrlinger, L. & Wöß, W. (2016), ‘Towards a definition of knowledge graphs.’, *SEMANTiCS (Posters, Demos, SuCCESS)* **48**.
- Ekman, P. (1992), ‘An argument for basic emotions’, *Cognition & emotion* **6**(3-4), 169–200.
- Ekman, P. (1999), ‘Basic emotions’, *Handbook of cognition and emotion* pp. 45–60.
- Ekman, P. E. & Davidson, R. J. (1994), *The nature of emotion: Fundamental questions.*, Oxford University Press.
- Ekman, P., Friesen, W. V., O’sullivan, M., Chan, A., Diacoyanni-Tarlatzis, I., Heider, K., Krause, R., LeCompte, W. A., Pitcairn, T., Ricci-Bitti, P. E. et al. (1987), ‘Universals and cultural differences in the judgments of facial expressions of emotion.’, *Journal of personality and social psychology* **53**(4), 712.
- El Hih, S. & Bengio, Y. (1996), Hierarchical recurrent neural networks for long-term dependencies, *in* ‘Advances in neural information processing systems’, pp. 493–499.
- Elman, J. L. (1990), ‘Finding structure in time’, *Cognitive science* **14**(2), 179–211.
- Ernst, P., Meng, C., Siu, A. & Weikum, G. (2014), Knowlife: a knowledge graph for health and life sciences, *in* ‘2014 IEEE 30th International Conference on Data Engineering’, IEEE, pp. 1254–1257.



- Ethayarajh, K. (2019a), ‘How contextual are contextualized word representations? comparing the geometry of bert, elmo, and gpt-2 embeddings’, *arXiv preprint arXiv:1909.00512* .
- Ethayarajh, K. (2019b), Rotate king to get queen: Word relationships as orthogonal transformations in embedding space, *in* ‘Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)’, pp. 3494–3499.
- Exner, P. & Nugues, P. (2012), Entity extraction: From unstructured text to dbpedia rdf triples., *in* ‘WoLE@ ISWC’, pp. 58–69.
- Explosion, A. (2017), ‘spacy-industrial-strength natural language processing in python’, *URL: <https://spacy.io>* .
- Facebook (2017), ‘Facebook for developers’, <https://developers.facebook.com>. Accessed on 2017-02-10.
- Facebook (2019), ‘Suicide prevention’.  
**URL:** <https://www.facebook.com/help/594991777257121/>
- Facebook (2020), ‘Graph api’, <https://developers.facebook.com/docs/graph-api/>. Accessed on 2020-05-20.
- Fader, A., Soderland, S. & Etzioni, O. (2011), Identifying relations for open information extraction, *in* ‘Proceedings of the conference on empirical methods in natural language processing’, Association for Computational Linguistics, pp. 1535–1545.
- Fan, A., Gardent, C., Braud, C. & Bordes, A. (2019), ‘Using local knowledge graph construction to scale seq2seq models to multi-document inputs’, *arXiv preprint arXiv:1910.08435* .

- Fan, C., Yan, H., Du, J., Gui, L., Bing, L., Yang, M., Xu, R. & Mao, R. (2019), A knowledge regularized hierarchical approach for emotion cause analysis, *in* ‘Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)’, pp. 5618–5628.
- Fan, M., Zhou, Q., Chang, E. & Zheng, F. (2014), Transition-based knowledge graph embedding with relational mapping properties, *in* ‘Proceedings of the 28th Pacific Asia Conference on Language, Information and Computing’, pp. 328–337.
- Fan, Y., Lu, X., Li, D. & Liu, Y. (2016), Video-based emotion recognition using cnn-rnn and c3d hybrid networks, *in* ‘Proceedings of the 18th ACM International Conference on Multimodal Interaction’, pp. 445–450.
- Fares, M., Moufarrej, A., Jreij, E., Tekli, J. & Grosky, W. (2019), ‘Unsupervised word-level affect analysis and propagation in a lexical knowledge graph’, *Knowledge-Based Systems* **165**, 432–459.
- Feng, J., Huang, M., Wang, M., Zhou, M., Hao, Y. & Zhu, X. (2016), Knowledge graph embedding by flexible translation, *in* ‘Fifteenth International Conference on the Principles of Knowledge Representation and Reasoning’.
- Fernández, S., Graves, A. & Schmidhuber, J. (2007), Sequence labelling in structured domains with hierarchical recurrent neural networks, *in* ‘Proceedings of the 20th International Joint Conference on Artificial Intelligence, IJCAI 2007’.
- Foley, S. & Kelly, B. (2018), ‘Forgiveness, spirituality and love: thematic analysis of last statements from death row, texas (2002-2017)’, *QJM: An International Journal of Medicine* .
- Fout, A., Byrd, J., Shariat, B. & Ben-Hur, A. (2017), Protein interface prediction using graph convolutional networks, *in* ‘Advances in neural information processing systems’, pp. 6530–6539.

- Fox, D. (2015), ‘Collaborative first order logic system with dynamic ontology’. US Patent 8,996,989.
- Gao, Z., Feng, A., Song, X. & Wu, X. (2019), ‘Target-dependent sentiment classification with bert’, *IEEE Access* **7**, 154290–154299.
- Gardner, M., Talukdar, P., Krishnamurthy, J. & Mitchell, T. (2014), Incorporating vector space similarity in random walk inference over knowledge bases, *in* ‘Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)’, pp. 397–406.
- Gardner, M. W. & Dorling, S. (1998), ‘Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences’, *Atmospheric environment* **32**(14-15), 2627–2636.
- Gers, F. A. & Schmidhuber, E. (2001), ‘Lstm recurrent networks learn simple context-free and context-sensitive languages’, *IEEE Transactions on Neural Networks* **12**(6), 1333–1340.
- Gers, F. A. & Schmidhuber, J. (2000), Recurrent nets that time and count, *in* ‘Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks. IJCNN 2000. Neural Computing: New Challenges and Perspectives for the New Millennium’, Vol. 3, IEEE, pp. 189–194.
- Giatsoglou, M., Vozalis, M. G., Diamantaras, K., Vakali, A., Sarigiannidis, G. & Chatzisavvas, K. C. (2017), ‘Sentiment analysis leveraging emotions and word embeddings’, *Expert Systems with Applications* **69**, 214–224.
- Gievska, S., Koroveshovski, K. & Chavdarova, T. (2014), A hybrid approach for emotion detection in support of affective interaction, *in* ‘Data Mining Workshop (ICDMW), 2014 IEEE International Conference on’, IEEE, pp. 352–359.

- Gilbert, C. & Hutto, E. (2014), Vader: A parsimonious rule-based model for sentiment analysis of social media text, *in* 'Eighth International Conference on Weblogs and Social Media (ICWSM-14). Available at (20/04/16) <http://comp.social.gatech.edu/papers/icwsm14.vader.hutto.pdf>', Vol. 81, p. 82.
- Gill, A. J., French, R. M., Gergle, D. & Oberlander, J. (2008), The language of emotion in short blog texts, *in* 'Proceedings of the 2008 ACM conference on Computer supported cooperative work', pp. 299–302.
- Glorot, X., Bordes, A. & Bengio, Y. (2011), Domain adaptation for large-scale sentiment classification: A deep learning approach, *in* 'Proceedings of the 28th international conference on machine learning (ICML-11)', pp. 513–520.
- Go, A., Bhayani, R. & Huang, L. (2009), 'Twitter sentiment classification using distant supervision', *CS224N project report, Stanford* 1(12), 2009.
- Godbole, N., Srinivasaiah, M. & Skiena, S. (2007), 'Large-scale sentiment analysis for news and blogs.', *IcwsM* 7(21), 219–222.
- Godin, F., Vandersmissen, B., De Neve, W. & Van de Walle, R. (2015), Multimedia lab@ acl wnut ner shared task: Named entity recognition for twitter microposts using distributed word representations, *in* 'Proceedings of the workshop on noisy user-generated text', pp. 146–153.
- Goodfellow, I., Bengio, Y., Courville, A. & Bengio, Y. (2016), *Deep learning*, Vol. 1, MIT press Cambridge.
- Google (2020), 'Access the google assistant with your voice', <https://support.google.com/assistant/answer/7394306?co=GENIE.Platform%3DAndroid&hl=en>. Accessed on 2020-10-01.
- Grassi, M. (2009), Developing the human emotions ontology, *in* 'European Workshop on Biometrics and Identity Management', Springer, pp. 244–251.

- Graves, A., Fernández, S., Gomez, F. & Schmidhuber, J. (2006), Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks, *in* ‘Proceedings of the 23rd international conference on Machine learning’, ACM, pp. 369–376.
- Graves, A. & Schmidhuber, J. (2009), Offline handwriting recognition with multidimensional recurrent neural networks, *in* ‘Advances in neural information processing systems’, pp. 545–552.
- Gregory, A. (1999), ‘The decision to die: The psychology of the suicide note’, *Interviewing and deception* pp. 127–156.
- Gudivada, V., Rao, D. & Raghavan, V. (2015), ‘Big data driven natural language processing research and applications’, *Big Data Analytics* **33**, 203.
- Gunn, J. F. & Lester, D. (2015), ‘Twitter postings and suicide: An analysis of the postings of a fatal suicide in the 24 hours prior to death’, *Suicidologi* **17**(3).
- Guo, L., Sun, Z. & Hu, W. (2019), ‘Learning to exploit long-term relational dependencies in knowledge graphs’, *arXiv preprint arXiv:1905.04914* .
- Hagberg, A., Schult, D. & Swart, P. (2005), ‘Networkx: Python software for the analysis of networks’, *Mathematical Modeling and Analysis, Los Alamos National Laboratory* .
- Hamaguchi, T., Oiwa, H., Shimbo, M. & Matsumoto, Y. (2017), ‘Knowledge transfer for out-of-knowledge-base entities: A graph neural network approach’, *arXiv preprint arXiv:1706.05674* .
- Han, J. & Moraga, C. (1995), The influence of the sigmoid function parameters on the speed of backpropagation learning, *in* ‘International Workshop on Artificial Neural Networks’, Springer, pp. 195–201.

- Hancock, J. T., Landrigan, C. & Silver, C. (2007), Expressing emotion in text-based communication, *in* ‘Proceedings of the SIGCHI conference on Human factors in computing systems’, ACM, pp. 929–932.
- Handelman, L. D. & Lester, D. (2007), ‘The content of suicide notes from attempters and completers’, *Crisis* **28**(2), 102–104.
- Haque, T. U., Saber, N. N. & Shah, F. M. (2018), Sentiment analysis on large scale amazon product reviews, *in* ‘2018 IEEE International Conference on Innovative Research and Development (ICIRD)’, IEEE, pp. 1–6.
- Harper, F. M. & Konstan, J. A. (2016), ‘The movielens datasets: History and context’, *Acm transactions on interactive intelligent systems (tiis)* **5**(4), 19.
- Hayiou-Thomas, M. E., Bishop, D. V. & Plunkett, K. (2004), ‘Simulating sli: General cognitive processing stressors can produce a specific linguistic profile’, *Journal of Speech, Language, and Hearing Research* **47**(6), 1347–1362.
- He, B., Zhou, D., Xiao, J., Liu, Q., Yuan, N. J., Xu, T. et al. (2019), ‘Integrating graph contextualized knowledge into pre-trained language models’, *arXiv preprint arXiv:1912.00147*.
- He, S., Liu, K., Ji, G. & Zhao, J. (2015), Learning to represent knowledge graphs with gaussian embedding, *in* ‘Proceedings of the 24th ACM International on Conference on Information and Knowledge Management’, pp. 623–632.
- Heflick, N. A. (2005), ‘Sentenced to die: Last statements and dying on death row’, *Omega-Journal of Death and Dying* **51**(4), 323–336.
- Henaff, M., Weston, J., Szlam, A., Bordes, A. & LeCun, Y. (2016), ‘Tracking the world state with recurrent entity networks’, *arXiv preprint arXiv:1612.03969*.
- Hengeveld, K. et al. (1997), ‘Adverbs in functional grammar’.

- Herrera, J., Penas, A. & Verdejo, F. (2005), Textual entailment recognition based on dependency analysis and wordnet, *in* ‘Machine Learning Challenges Workshop’, Springer, pp. 231–239.
- Hochreiter, S., Bengio, Y., Frasconi, P., Schmidhuber, J. et al. (2001), ‘Gradient flow in recurrent nets: the difficulty of learning long-term dependencies’.
- Hochreiter, S. & Schmidhuber, J. (1997a), ‘Long short-term memory’, *Neural Comput.* **9**(8), 1735–1780.
- Hochreiter, S. & Schmidhuber, J. (1997b), ‘Long short-term memory’, *Neural computation* **9**(8), 1735–1780.
- Hochreiter, S. & Schmidhuber, J. (1997c), Lstm can solve hard long time lag problems, *in* ‘Advances in neural information processing systems’, pp. 473–479.
- Holzman, L. E. & Pottenger, W. M. (2003), ‘Classification of emotions in internet chat: An application of machine learning using speech phonemes’, *Retrieved November* **27**(2011), 50.
- Hwang, H. & Matsumoto, D. (2013), ‘Functions of emotions’, *R. Biswas-Diener & E. Diener* .
- Inc, A. (2020), ‘Google knowledge graph search api’, <https://developers.google.com/knowledge-graph>. Accessed on 2020-05-20.
- Instagram (2017), ‘Instagram developer’, <https://www.instagram.com/developer>. Accessed on 2017-02-10.
- Ioannou, M. & Debowska, A. (2014), ‘Genuine and simulated suicide notes: An analysis of content’, *Forensic science international* **245**, 151–160.
- Irsoy, O. & Cardie, C. (2014), Opinion mining with deep recurrent neural networks, *in* ‘Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)’, pp. 720–728.

ITU (2019), ‘Ai for good summit’.

**URL:** <https://aiforgood.itu.int/>

Izard, C. E. (1992), ‘Basic emotions, relations among emotions, and emotion-cognition relations.’.

Jaeger, H. & Haas, H. (2004), ‘Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication’, *science* **304**(5667), 78–80.

Jaiswal, M., Tabibu, S. & Cambria, E. (2017), ““hang in there”: Lexical and visual analysis to identify posts warranting empathetic responses’.

Ji, G., He, S., Xu, L., Liu, K. & Zhao, J. (2015), Knowledge graph embedding via dynamic mapping matrix, *in* ‘Proceedings of the 53rd annual meeting of the association for computational linguistics and the 7th international joint conference on natural language processing (volume 1: Long papers)’, pp. 687–696.

Ji, S., Pan, S., Cambria, E., Marttinen, P. & Yu, P. S. (2020), ‘A survey on knowledge graphs: Representation, acquisition and applications’, *arXiv preprint arXiv:2002.00388* .

Ji, S., Pan, S., Li, X., Cambria, E., Long, G. & Huang, Z. (2019), ‘Suicidal ideation detection: A review of machine learning methods and applications’, *arXiv preprint arXiv:1910.12611* .

Ji, S., Pan, S., Li, X., Cambria, E., Long, G. & Huang, Z. (2020), ‘Suicidal ideation detection: A review of machine learning methods and applications’, *IEEE Transactions on Computational Social Systems* .

Jones, N. J. & Bennell, C. (2007), ‘The development and validation of statistical prediction rules for discriminating between genuine and simulated suicide notes’, *Archives of Suicide Research* **11**(2), 219–233.



- Just, M. A., Pan, L., Cherkassky, V. L., McMakin, D. L., Cha, C., Nock, M. K. & Brent, D. (2017), ‘Machine learning of neural representations of suicide and emotion concepts identifies suicidal youth’, *Nature human behaviour* **1**(12), 911.
- Kelly, B. D. & Foley, S. R. (2017), ‘Analysis of last statements prior to execution: methods, themes and future directions’, *QJM: An International Journal of Medicine* **111**(1), 3–6.
- Kertkeidkachorn, N. & Ichise, R. (2018), ‘An automatic knowledge graph creation framework from natural language text’, *IEICE TRANSACTIONS on Information and Systems* **101**(1), 90–98.
- Khandelwal, U., He, H., Qi, P. & Jurafsky, D. (2018), ‘Sharp nearby, fuzzy far away: How neural language models use context’, *arXiv preprint arXiv:1805.04623*.
- Kim, S., Bak, J. & Oh, A. H. (2012), Do you feel what i feel? social aspects of emotions in twitter conversations., *in* ‘ICWSM’.
- Kim, Y. (2014), ‘Convolutional neural networks for sentence classification’, *arXiv preprint arXiv:1408.5882*.
- Kingma, D. P. & Ba, J. (2014), ‘Adam: A method for stochastic optimization’, *arXiv preprint arXiv:1412.6980*.
- Kipf, T. N. & Welling, M. (2016), ‘Semi-supervised classification with graph convolutional networks’, *arXiv preprint arXiv:1609.02907*.
- Kiritchenko, S. & Mohammad, S. M. (2018), ‘Examining gender and race bias in two hundred sentiment analysis systems’, *arXiv preprint arXiv:1805.04508*.
- Kiritchenko, S., Mohammad, S. M. & Salameh, M. (2016), Semeval-2016 task 7: Determining sentiment intensity of english and arabic phrases, *in* ‘Proceedings of the International Workshop on Semantic Evaluation’, SemEval ’16, San Diego, California.

- Kiritchenko, S. & Nejadgholi, I. (2020), 'Towards ethics by design in online abusive content detection', *arXiv preprint arXiv:2010.14952* .
- Kiritchenko, S., Zhu, X. & Mohammad, S. M. (2014), 'Sentiment analysis of short informal texts', *Journal of Artificial Intelligence Research* **50**, 723–762.
- Klinger, R., De Clercq, O., Mohammad, S. M. & Balahur, A. (2018), 'Test: Wassa-2018 implicit emotions shared task', *arXiv preprint arXiv:1809.01083* .
- Kong, L., Schneider, N., Swayamdipta, S., Bhatia, A., Dyer, C. & Smith, N. A. (2014), A dependency parser for tweets, *in* 'Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)', pp. 1001–1012.
- Konstas, I., Iyer, S., Yatskar, M., Choi, Y. & Zettlemoyer, L. (2017), 'Neural amr: Sequence-to-sequence models for parsing and generation', *arXiv preprint arXiv:1704.08381* .
- Kontopoulos, E., Berberidis, C., Dergiades, T. & Bassiliades, N. (2013), 'Ontology-based sentiment analysis of twitter posts', *Expert systems with applications* **40**(10), 4065–4074.
- Kouloumpis, E., Wilson, T. & Moore, J. D. (2011), 'Twitter sentiment analysis: The good the bad and the omg!', *Icwsn* **11**(538-541), 164.
- Koutnik, J., Greff, K., Gomez, F. & Schmidhuber, J. (2014), 'A clockwork rnn', *arXiv preprint arXiv:1402.3511* .
- Krasnowska-Kieraś, K. & Wróblewska, A. (2019), Empirical linguistic study of sentence embeddings, *in* 'Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics', pp. 5729–5739.
- Kříž, V., Hladká, B., Nečaský, M. & Knap, T. (2014), Data extraction using nlp techniques and its transformation to linked data, *in* 'Mexican International Conference on Artificial Intelligence', Springer, pp. 113–124.

- Krizhevsky, A., Sutskever, I. & Hinton, G. E. (2012), Imagenet classification with deep convolutional neural networks, *in* ‘Advances in neural information processing systems’, pp. 1097–1105.
- Kumar, A., Irsoy, O., Ondruska, P., Iyyer, M., Bradbury, J., Gulrajani, I., Zhong, V., Paulus, R. & Socher, R. (2016), Ask me anything: Dynamic memory networks for natural language processing, *in* ‘International conference on machine learning’, pp. 1378–1387.
- Kumar, A. & Jaiswal, A. (2017), Empirical study of twitter and tumblr for sentiment analysis using soft computing techniques, *in* ‘Proceedings of the world congress on engineering and computer science’, Vol. 1, pp. 1–5.
- Kumara, A., Kawaharab, D. & Kurohashib, S. (2018), Knowledge-enriched two-layered attention network for sentiment analysis, *in* ‘Proceedings of NAACL-HLT’, pp. 253–258.
- Labutov, I. & Lipson, H. (2013), Re-embedding words, *in* ‘Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)’, pp. 489–493.
- Lan, M., Zhang, Z., Lu, Y. & Wu, J. (2016), Three convolutional neural network-based models for learning sentiment word vectors towards sentiment analysis, *in* ‘2016 International Joint Conference on Neural Networks (IJCNN)’, IEEE, pp. 3172–3179.
- Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P. & Soricut, R. (2019), ‘Albert: A lite bert for self-supervised learning of language representations’, *arXiv preprint arXiv:1909.11942* .
- Le, Q. V., Jaitly, N. & Hinton, G. E. (2015), ‘A simple way to initialize recurrent networks of rectified linear units’, *arXiv preprint arXiv:1504.00941* .

- LeCun, Y., Bengio, Y. & Hinton, G. (2015), 'Deep learning', *Nature* **521**(7553), 436–444.
- Lee, J. Y. & Dernoncourt, F. (2016), Sequential short-text classification with recurrent and convolutional neural networks, *in* 'Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies', pp. 515–520.
- Leenaars, A. (1988), 'Suicide notes'.
- Lester, D. & Gunn III, J. F. (2013), 'Ethnic differences in the statements made by inmates about to be executed in texas', *Journal of Ethnicity in Criminal Justice* **11**(4), 295–301.
- Lester, D. & Leenaars, A. A. (1988), 'The moral justification of suicide in suicide notes.', *Psychological reports* .
- Levine, S., Pastor, P., Krizhevsky, A., Ibarz, J. & Quillen, D. (2018), 'Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection', *The International Journal of Robotics Research* **37**(4-5), 421–436.
- Levy, N. (2017), 'The bad news about fake news', *Social epistemology review and reply collective* **6**(8), 20–36.
- Li, C. & Goldwasser, D. (2019), Encoding social information with graph convolutional networks for political perspective detection in news media, *in* 'Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics', pp. 2594–2604.
- Li, J., Ren, P., Chen, Z., Ren, Z., Lian, T. & Ma, J. (2017), Neural attentive session-based recommendation, *in* 'Proceedings of the 2017 ACM on Conference on Information and Knowledge Management', pp. 1419–1428.

- Li, M., Lu, Q., Long, Y. & Gui, L. (2017), ‘Inferring affective meanings of words from word embedding’, *IEEE Transactions on Affective Computing* **8**(4), 443–456.
- Li, W., Zhu, L., Shi, Y., Guo, K. & Zheng, Y. (2020), ‘User reviews: Sentiment analysis using lexicon integrated two-channel cnn-lstm family models’, *Applied Soft Computing* p. 106435.
- Li, Y., Tarlow, D., Brockschmidt, M. & Zemel, R. (2015), ‘Gated graph sequence neural networks’, *arXiv preprint arXiv:1511.05493* .
- Li, Z., Wei, Y., Zhang, Y. & Yang, Q. (2018), Hierarchical attention transfer network for cross-domain sentiment classification, *in* ‘Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, AAAI 2018, New Orleans, Louisiana, USA, February 2–7, 2018’.
- Liakata, M., Kim, J.-H., Saha, S., Hastings, J. & Rebholz-Schuhmann, D. (2012), ‘Three hybrid classifiers for the detection of emotions in suicide notes’, *Biomedical informatics insights* **5**, BII–S8967.
- Liang, B., Du, J., Xu, R., Li, B. & Huang, H. (2019), ‘Context-aware embedding for targeted aspect-based sentiment analysis’, *arXiv preprint arXiv:1906.06945* .
- Liang, X., Shen, X., Feng, J., Lin, L. & Yan, S. (2016), Semantic object parsing with graph lstm, *in* ‘European Conference on Computer Vision’, Springer, pp. 125–143.
- Lin, D., Matsumoto, Y. & Mihalcea, R. (2011), Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies, *in* ‘Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies’.
- Lin, H., Liu, Y., Wang, W., Yue, Y. & Lin, Z. (2017), ‘Learning entity and relation embeddings for knowledge resolution’, *Procedia Computer Science* **108**, 345–354.

- Lin, T., Horne, B. G., Tino, P. & Giles, C. L. (1996), 'Learning long-term dependencies in narx recurrent neural networks', *IEEE Transactions on Neural Networks* **7**(6), 1329–1338.
- Lin, W.-H., Xing, E. & Hauptmann, A. (2008), A joint topic and perspective model for ideological discourse, *in* 'Joint European Conference on Machine Learning and Knowledge Discovery in Databases', Springer, pp. 17–32.
- Lin, Y., Liu, Z., Sun, M., Liu, Y. & Zhu, X. (2015), Learning entity and relation embeddings for knowledge graph completion, *in* 'Twenty-ninth AAAI conference on artificial intelligence'.
- Lindquist, K. A., Siegel, E. H., Quigley, K. S. & Barrett, L. F. (2013), 'The hundred-year emotion war: Are emotions natural kinds or psychological constructions? comment on lench, flores, and bench (2011).'.<sup>1</sup>
- Liu, H. & Singh, P. (2004), 'Conceptnet—a practical commonsense reasoning tool-kit', *BT technology journal* **22**(4), 211–226.
- Liu, J., Lu, Z. & Du, W. (2019), Combining enterprise knowledge graph and news sentiment analysis for stock price prediction, *in* 'Proceedings of the 52nd Hawaii International Conference on System Sciences'.
- Liu, X., Bordes, A. & Grandvalet, Y. (2014), Fast recursive multi-class classification of pairs of text entities for biomedical event extraction, *in* 'Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics', pp. 692–701.
- Logan, R., Liu, N. F., Peters, M. E., Gardner, M. & Singh, S. (2019), Barack's wife hillary: Using knowledge graphs for fact-aware language modeling, *in* 'Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics', pp. 5962–5971.

- Loper, E. & Bird, S. (2002), 'Nltk: the natural language toolkit', *arXiv preprint cs/0205028*.
- Lu, B., Ott, M., Cardie, C. & Tsou, B. K. (2011), Multi-aspect sentiment analysis with topic models, *in* 'Data Mining Workshops (ICDMW), 2011 IEEE 11th International Conference on', IEEE, pp. 81–88.
- Luong, M.-T., Pham, H. & Manning, C. D. (2015), 'Effective approaches to attention-based neural machine translation', *arXiv preprint arXiv:1508.04025*.
- Ma, Y., Peng, H. & Cambria, E. (2018), Targeted aspect-based sentiment analysis via embedding commonsense knowledge into an attentive lstm, *in* 'Proceedings of AAAI'.
- Ma, Y., Peng, H., Khan, T., Cambria, E. & Hussain, A. (2018), 'Sentic lstm: a hybrid network for targeted aspect-based sentiment analysis', *Cognitive Computation* **10**(4), 639–650.
- Ma, Y., Zong, L., Yang, Y. & Su, J. (2019), News2vec: News network embedding with subnode information, *in* 'Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)', pp. 4845–4854.
- Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y. & Potts, C. (2011), Learning word vectors for sentiment analysis, *in* 'Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies-volume 1', Association for Computational Linguistics, pp. 142–150.
- Manning, C. D. & Schütze, H. (1999), *Foundations of statistical natural language processing*, MIT press.
- Martens, J. & Sutskever, I. (2011), Learning recurrent neural networks with

hessian-free optimization, *in* ‘Proceedings of the 28th International Conference on Machine Learning (ICML-11)’, Citeseer, pp. 1033–1040.

Mayhew, S., Tsygankova, T. & Roth, D. (2019), ner and pos when nothing is capitalized, *in* ‘Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)’, Association for Computational Linguistics, Hong Kong, China, pp. 6256–6261.

**URL:** <https://www.aclweb.org/anthology/D19-1650>

McCann, B., Bradbury, J., Xiong, C. & Socher, R. (2017), Learned in translation: Contextualized word vectors, *in* ‘Advances in Neural Information Processing Systems’, pp. 6294–6305.

Meiselman, H. L. (2016), *Emotion Measurement*, Woodhead Publishing.

Mental Health Foundation (2015), ‘Depression’.

**URL:** <https://www.mentalhealth.org.uk/a-to-z/d/depression>

Mesnil, G., Mikolov, T., Ranzato, M. & Bengio, Y. (2014), ‘Ensemble of generative and discriminative techniques for sentiment analysis of movie reviews’, *arXiv preprint arXiv:1412.5335* .

Mesquita, F., Cannavicchio, M., Schmidek, J., Mirza, P. & Barbosa, D. (2019), Knowledgenet: A benchmark dataset for knowledge base population, *in* ‘Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)’, pp. 749–758.

Miikkulainen, R. (1993), *Subsymbolic natural language processing: An integrated model of scripts, lexicon, and memory*, MIT press.



- Mikolov, T., Chen, K., Corrado, G. & Dean, J. (2013), ‘Efficient estimation of word representations in vector space’, *arXiv preprint arXiv:1301.3781* .
- Mikolov, T., Deoras, A., Povey, D., Burget, L. & Černocký, J. (2011), Strategies for training large scale neural network language models, *in* ‘2011 IEEE Workshop on Automatic Speech Recognition & Understanding’, IEEE, pp. 196–201.
- Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S. & Dean, J. (2013), Distributed representations of words and phrases and their compositionality, *in* ‘Advances in neural information processing systems’, pp. 3111–3119.
- Miller, G. A. (1991), ‘The science of words’.
- Miller, G. A. (1995), ‘Wordnet: a lexical database for english’, *Communications of the ACM* **38**(11), 39–41.
- Minaee, S., Kalchbrenner, N., Cambria, E., Nikzad, N., Chenaghlu, M. & Gao, J. (2020), ‘Deep learning based text classification: A comprehensive review’, *arXiv preprint arXiv:2004.03705* .
- Mind (2013), ‘Depression’.  
**URL:** <https://tinyurl.com/y2xhmnf9>
- Minsky, M. (1988), *Society of mind*, Simon and Schuster.
- Minsky, M. (2006), ‘The emotion machine’, *New York: Pantheon* **56**.
- Mishne, G., De Rijke, M. et al. (2006), Moodviews: Tools for blog mood analysis., *in* ‘AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs’, pp. 153–154.
- Miyazaki, T., Makino, K., Takei, Y., Okamoto, H. & Goto, J. (2019), Label embedding using hierarchical structure of labels for twitter classification, *in* ‘Proceedings of the 2019 Conference on Empirical Methods in Natural Language

- Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)', pp. 6318–6323.
- Mohammad, S., Bravo-Marquez, F., Salameh, M. & Kiritchenko, S. (2018), Semeval-2018 task 1: Affect in tweets, *in* 'Proceedings of the 12th international workshop on semantic evaluation', pp. 1–17.
- Mohammad, S. M. (2012), # emotional tweets, *in* 'Proceedings of the First Joint Conference on Lexical and Computational Semantics-Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation', Association for Computational Linguistics, pp. 246–255.
- Mohammad, S. M. (2016), Sentiment analysis: Detecting valence, emotions, and other affectual states from text, *in* 'Emotion measurement', Elsevier, pp. 201–237.
- Mohammad, S. M. (2020), 'Practical and ethical considerations in the effective use of emotion and sentiment lexicons', *arXiv preprint arXiv:2011.03492* .
- Mohammad, S. M. & Kiritchenko, S. (2015), 'Using hashtags to capture fine emotion categories from tweets', *Computational Intelligence* **31**(2), 301–326.
- Mohammad, S. M., Kiritchenko, S. & Zhu, X. (2013), 'Nrc-canada: Building the state-of-the-art in sentiment analysis of tweets', *arXiv preprint arXiv:1308.6242* .
- Mohammad, S. M. & Turney, P. D. (2013), 'Nrc emotion lexicon', *National Research Council, Canada* .
- Montgomery, J. W. (2000), 'Relation of working memory to off-line and real-time sentence processing in children with specific language impairment', *Applied Psycholinguistics* **21**(1), 117–148.
- Montiel-Ponsoda, E., Gracia, J. & Rodríguez-Doncel, V. (2018), Building the legal

- knowledge graph for smart compliance services in multilingual europe, *in* ‘CEUR workshop proc.’.
- Morales, M. R. & Levitan, R. (2016), Speech vs. text: A comparative analysis of features for depression detection systems, *in* ‘2016 IEEE Spoken Language Technology Workshop (SLT)’, IEEE, pp. 136–143.
- Morales, M., Scherer, S. & Levitan, R. (2017), A cross-modal review of indicators for depression detection systems, *in* ‘Proceedings of the Fourth Workshop on Computational Linguistics and Clinical Psychology - From Linguistic Signal to Clinical Reality’, pp. 1–12.
- Mowery, D., Park, A., Conway, M. & Bryan, C. (2016), Towards automatically classifying depressive symptoms from twitter data for population health, *in* ‘Proceedings of the Workshop on Computational Modeling of People’s Opinions, Personality, and Emotions in Social Media’, pp. 182–191.
- Munzero, M., Montero, C. S., Sutinen, E. & Pajunen, J. (2014), ‘Are they different? affect, feeling, emotion, sentiment, and opinion detection in text’, *IEEE transactions on affective computing* **5**(2), 101–111.
- Nair, V. & Hinton, G. E. (2010), Rectified linear units improve restricted boltzmann machines, *in* ‘Proceedings of the 27th international conference on machine learning (ICML-10)’, pp. 807–814.
- Najafabadi, M. M., Villanustre, F., Khoshgoftaar, T. M., Seliya, N., Wald, R. & Muharemagic, E. (2015), ‘Deep learning applications and challenges in big data analytics’, *Journal of Big Data* **2**(1), 1.
- Nakashole, N., Theobald, M. & Weikum, G. (2011), Scalable knowledge harvesting with high precision and high recall, *in* ‘Proceedings of the fourth ACM international conference on Web search and data mining’, pp. 227–236.

- Nakov, P., Ritter, A., Rosenthal, S., Sebastiani, F. & Stoyanov, V. (2016), Semeval-2016 task 4: Sentiment analysis in twitter, *in* 'Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)', pp. 1–18.
- Nathani, D., Chauhan, J., Sharma, C. & Kaul, M. (2019), Learning attention-based embeddings for relation prediction in knowledge graphs, *in* 'Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics', pp. 4710–4723.
- Neelakantan, A., Roth, B. & McCallum, A. (2015), Compositional vector space models for knowledge base inference., *in* 'AAAI Spring Symposia'.
- Neil, D., Pfeiffer, M. & Liu, S.-C. (2016), Phased lstm: Accelerating recurrent network training for long or event-based sequences, *in* 'Advances in Neural Information Processing Systems', pp. 3882–3890.
- Newman, M. L., Pennebaker, J. W., Berry, D. S. & Richards, J. M. (2003), 'Lying words: Predicting deception from linguistic styles', *Personality and social psychology bulletin* **29**(5), 665–675.
- Nguyen, T. D., Nguyen, D. Q., Phung, D. et al. (2018), A novel embedding model for knowledge base completion based on convolutional neural network, *in* 'Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)', pp. 327–333.
- Nguyen, T., Phung, D., Dao, B., Venkatesh, S. & Berk, M. (2014), 'Affective and content analysis of online depression communities', *IEEE Transactions on Affective Computing* **5**(3), 217–226.
- Nickel, M., Rosasco, L., Poggio, T. A. et al. (2016), Holographic embeddings of knowledge graphs., *in* 'AAAI', Vol. 2.

Nickel, M., Tresp, V. & Kriegel, H.-P. (2011), A three-way model for collective learning on multi-relational data., *in* 'Icml', Vol. 11, pp. 809–816.

Nithish, R., Sabarish, S., Kishen, M. N., Abirami, A. & Askarunisa, A. (2013), An ontology based sentiment analysis for mobile products using tweets, *in* 'Advanced Computing (ICoAC), 2013 Fifth International Conference on', IEEE, pp. 342–347.

Niu, F. (1912), Web-scale knowledge-base construction via statistical inference and learning, PhD thesis, UNIVERSITY OF WISCONSIN–MADISON.

Novak, P. K., Smailović, J., Sluban, B. & Mozetič, I. (2015), 'Sentiment of emojis', *PloS one* **10**(12), e0144296.

O'dea, B., Larsen, M. E., Batterham, P. J., Calex, A. L. & Christensen, H. (2017), 'A linguistic analysis of suicide-related twitter posts', *Crisis* .

Ofek, N., Poria, S., Rokach, L., Cambria, E., Hussain, A. & Shabtai, A. (2016), 'Unsupervised commonsense knowledge enrichment for domain-specific sentiment analysis', *Cognitive Computation* **8**(3), 467–477.

Office, I. C. (2020), 'General data protection regulation'.

**URL:** <https://ico.org.uk/for-organisations/guide-to-data-protection/guide-to-the-general-data>

O'neil, C. (2016), *Weapons of math destruction: How big data increases inequality and threatens democracy*, Broadway Books.

Oneto, L., Bisio, F., Cambria, E. & Anguita, D. (2017), 'Semi-supervised learning for affective common-sense reasoning', *Cognitive Computation* **9**(1), 18–42.

Onishi, H. & Manchanda, P. (2012), 'Marketing activity, blogging and sales', *International Journal of Research in Marketing* **29**(3), 221–234.

Osgood, C. E. (1960), 'The cross-cultural generality of visual-verbal synesthetic tendencies', *Systems research and behavioral science* **5**(2), 146–169.

- Osgood, C. E. & Walker, E. G. (1959), ‘Motivation and language behavior: A content analysis of suicide notes.’, *The Journal of Abnormal and Social Psychology* **59**(1), 58.
- Pang, B., Lee, L. & Vaithyanathan, S. (2002), Thumbs up?: sentiment classification using machine learning techniques, *in* ‘Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10’, Association for Computational Linguistics, pp. 79–86.
- Panksepp, J. (2004), *Affective neuroscience: The foundations of human and animal emotions*, Oxford university press.
- Pascanu, R., Mikolov, T. & Bengio, Y. (2012), ‘Understanding the exploding gradient problem’, *CoRR*, *abs/1211.5063* .
- Pascanu, R., Mikolov, T. & Bengio, Y. (2013), On the difficulty of training recurrent neural networks, *in* ‘International conference on machine learning’, pp. 1310–1318.
- Paulheim, H. (2017), ‘Knowledge graph refinement: A survey of approaches and evaluation methods’, *Semantic web* **8**(3), 489–508.
- Pedersen, T., Patwardhan, S. & Michelizzi, J. (2004), Wordnet:: Similarity: measuring the relatedness of concepts, *in* ‘Demonstration papers at HLT-NAACL 2004’, Association for Computational Linguistics, pp. 38–41.
- Peng, N., Poon, H., Quirk, C., Toutanova, K. & Yih, W.-t. (2017), ‘Cross-sentence n-ary relation extraction with graph lstms’, *Transactions of the Association for Computational Linguistics* **5**, 101–115.
- Pennebaker, J. W., Chung, C. K., Frazee, J., Lavergne, G. M. & Beaver, D. I. (2014), ‘When small words foretell academic success: The case of college admissions essays’, *PloS one* **9**(12), e115844.

- Pennington, J., Socher, R. & Manning, C. D. (2014), Glove: Global vectors for word representation, *in* ‘Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)’, pp. 1532–1543.
- Pestian, J., Nasrallah, H., Matykiewicz, P., Bennett, A. & Leenaars, A. (2010), ‘Suicide note classification using natural language processing: A content analysis’, *Biomedical informatics insights* **3**, BII–S4706.
- Pestian, J. P., Matykiewicz, P., Linn-Gust, M., South, B., Uzuner, O., Wiebe, J., Cohen, K. B., Hurdle, J. & Brew, C. (2012), ‘Sentiment analysis of suicide notes: A shared task’, *Biomedical informatics insights* **5**(Suppl 1), 3.
- Peters, M. E., Ammar, W., Bhagavatula, C. & Power, R. (2017), ‘Semi-supervised sequence tagging with bidirectional language models’, *arXiv preprint arXiv:1705.00108* .
- Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K. & Zettlemoyer, L. (2018), ‘Deep contextualized word representations’, *arXiv preprint arXiv:1802.05365* .
- Peters, M. E., Neumann, M., Logan, I., Robert, L., Schwartz, R., Joshi, V., Singh, S. & Smith, N. A. (2019), ‘Knowledge enhanced contextual word representations’, *arXiv preprint arXiv:1909.04164* .
- Petroni, F., Rocktäschel, T., Lewis, P., Bakhtin, A., Wu, Y., Miller, A. H. & Riedel, S. (2019), ‘Language models as knowledge bases?’, *arXiv preprint arXiv:1909.01066* .
- Picard, R. W. (2000), *Affective computing*, MIT press.
- Picasso, A., Merello, S., Ma, Y., Oneto, L. & Cambria, E. (2019), ‘Technical analysis and sentiment embeddings for market trend prediction’, *Expert Systems with Applications* **135**, 60–70.

- Pirina, I. & Çöltekin, Ç. (2018), Identifying depression on reddit: The effect of training data, *in* ‘Proceedings of the 2018 EMNLP Workshop SMM4H: The 3rd Social Media Mining for Health Applications Workshop & Shared Task’, pp. 9–12.
- Plutchik, R. (1984), ‘Emotions: A general psychoevolutionary theory’, *Approaches to emotion* **1984**, 197–219.
- Plutchik, R. (1990), Emotions and psychotherapy: A psychoevolutionary perspective, *in* ‘Emotion, psychopathology, and psychotherapy’, Elsevier, pp. 3–41.
- Plutchik, R. (2001), ‘The nature of emotions: Human emotions have deep evolutionary roots, a fact that may explain their complexity and provide tools for clinical practice’, *American scientist* **89**(4), 344–350.
- Pontiki, M., Galanis, D., Papageorgiou, H., Androutsopoulos, I., Manandhar, S., Mohammad, A.-S., Al-Ayyoub, M., Zhao, Y., Qin, B., De Clercq, O. et al. (2016), Semeval-2016 task 5: Aspect based sentiment analysis, *in* ‘Proceedings of the 10th international workshop on semantic evaluation (SemEval-2016)’, pp. 19–30.
- Ponzetto, S. P. & Strube, M. (2006), Exploiting semantic role labeling, wordnet and wikipedia for coreference resolution, *in* ‘Proceedings of the Human Language Technology Conference of the NAACL, Main Conference’, pp. 192–199.
- Radford, A., Wu, J., Child, R., Luan, D., Amodei, D. & Sutskever, I. (2019), ‘Language models are unsupervised multitask learners’, *OpenAI Blog* **1**(8), 9.
- Rahman, A. & Ng, V. (2011), Coreference resolution with world knowledge, *in* ‘Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies’, pp. 814–824.
- Rajshree, N. (2017), ‘Enterprise knowledge graphs for large scale analytics’, <https://>



- [//cci.drexel.edu/bigdata/bigdata2017/files/Tutorial1-1.pdf](http://cci.drexel.edu/bigdata/bigdata2017/files/Tutorial1-1.pdf). Accessed on 2020-05-20.
- Rakib, T. B. A. & Soon, L.-K. (2018), Using the reddit corpus for cyberbully detection, *in* 'Asian Conference on Intelligent Information and Database Systems', Springer, pp. 180–189.
- Ramteke, J., Shah, S., Godhia, D. & Shaikh, A. (2016), Election result prediction using twitter sentiment analysis, *in* '2016 international conference on inventive computation technologies (ICICT)', Vol. 1, IEEE, pp. 1–5.
- Ravi, K. & Ravi, V. (2015), 'A survey on opinion mining and sentiment analysis: tasks, approaches and applications', *Knowledge-Based Systems* **89**, 14–46.
- Recupero, D. R., Presutti, V., Consoli, S., Gangemi, A. & Nuzzolese, A. G. (2015), 'Sentilo: frame-based sentiment analysis', *Cognitive Computation* **7**(2), 211–225.
- Reed, S. E., Zhang, Y., Zhang, Y. & Lee, H. (2015), Deep visual analogy-making, *in* 'Advances in neural information processing systems', pp. 1252–1260.
- Ren, F., Kang, X. & Quan, C. (2015), 'Examining accumulated emotional traits in suicide blogs with an emotion topic model', *IEEE journal of biomedical and health informatics* **20**(5), 1384–1396.
- Ren, Y., Zhang, Y., Zhang, M. & Ji, D. (2016), Improving twitter sentiment classification using topic-enriched multi-prototype word embeddings, *in* 'Thirtieth AAAI conference on artificial intelligence'.
- Resnik, P., Armstrong, W., Claudino, L., Nguyen, T., Nguyen, V.-A. & Boyd-Graber, J. (2015), Beyond lda: exploring supervised topic modeling for depression-related language in twitter, *in* 'Proceedings of the 2nd Workshop on Computational Linguistics and Clinical Psychology: From Linguistic Signal to Clinical Reality', pp. 99–107.

- Riordan, M. A. & Trichtinger, L. A. (2017), ‘Overconfidence at the keyboard: Confidence and accuracy in interpreting affect in e-mail exchanges’, *Human Communication Research* **43**(1), 1–24.
- Roberts, K., Roach, M. A., Johnson, J., Guthrie, J. & Harabagiu, S. M. (2012), Empatweet: Annotating and detecting emotions on twitter., *in* ‘LREC’, Vol. 12, Citeseer, pp. 3806–3813.
- Rosenblatt, F. (1957), *The perceptron, a perceiving and recognizing automaton Project Para*, Cornell Aeronautical Laboratory.
- Rosenthal, S., Farra, N. & Nakov, P. (2017), Semeval-2017 task 4: Sentiment analysis in twitter, *in* ‘Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)’, pp. 502–518.
- Rosenthal, S. & McKeown, K. (2011), Age prediction in blogs: A study of style, content, and online behavior in pre-and post-social media generations, *in* ‘Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1’, Association for Computational Linguistics, pp. 763–772.
- Rospoche, M., van Erp, M., Vossen, P., Fokkens, A., Aldabe, I., Rigau, G., Soroa, A., Ploeger, T. & Bogaard, T. (2016), ‘Building event-centric knowledge graphs from news’, *Journal of Web Semantics* **37**, 132–151.
- Rothkrantz, L. (2014), Online emotional facial expression dictionary, *in* ‘Proceedings of the 15th International Conference on Computer Systems and Technologies’, ACM, pp. 116–123.
- Rotmensch, M., Halpern, Y., Tlimat, A., Horng, S. & Sontag, D. (2017), ‘Learning a health knowledge graph from electronic medical records’, *Scientific reports* **7**(1), 1–11.

- Rozental, A. & Fleischer, D. (2018), ‘Amobee at semeval-2018 task 1: Gru neural network with a cnn attention mechanism for sentiment classification’, *arXiv preprint arXiv:1804.04380* .
- Rude, S., Gortner, E.-M. & Pennebaker, J. (2004), ‘Language use of depressed and depression-vulnerable college students’, *Cognition & Emotion* **18**(8), 1121–1133.
- Ruder, S. (2019), Neural transfer learning for natural language processing, PhD thesis, NUI Galway.
- Rumelhart, D. E., Hinton, G. E. & Williams, R. J. (1985), Learning internal representations by error propagation, Technical report, California Univ San Diego La Jolla Inst for Cognitive Science.
- Russell, S. & Norvig, P. (2002), ‘Artificial intelligence: a modern approach’.
- Sachan, D. S., Zaheer, M. & Salakhutdinov, R. (2018), ‘Revisiting lstm networks for semi-supervised text classification via mixed objective function’.
- Saeidi, M., Bouchard, G., Liakata, M. & Riedel, S. (2016), ‘Sentihood: Targeted aspect based sentiment analysis dataset for urban neighbourhoods’, *arXiv preprint arXiv:1610.03771* .
- Saif, H., Fernandez, M., He, Y. & Alani, H. (2013), ‘Evaluation datasets for twitter sentiment analysis: a survey and a new dataset, the sts-gold’.
- Sander, D. (2013), ‘Models of emotion’, *The Cambridge handbook of human affective neuroscience* p. 1.
- Savova, G., Pestian, J., Connolly, B., Miller, T., Ni, Y. & Dexheimer, J. W. (2016), Natural language processing: applications in pediatric research, *in* ‘Pediatric biomedical informatics’, Springer, pp. 231–250.
- Sawhney, R., Manchanda, P., Mathur, P., Shah, R. & Singh, R. (2018), Exploring and learning suicidal ideation connotations on social media with deep learning, *in*

- ‘Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis’, pp. 167–175.
- Saxe, A. M., McClelland, J. L. & Ganguli, S. (2013), ‘Exact solutions to the nonlinear dynamics of learning in deep linear neural networks’, *arXiv preprint arXiv:1312.6120*.
- Scarselli, F., Gori, M., Tsoi, A. C., Hagenbuchner, M. & Monfardini, G. (2008), ‘The graph neural network model’, *IEEE Transactions on Neural Networks* **20**(1), 61–80.
- Schler, J., Koppel, M., Argamon, S. & Pennebaker, J. W. (2006), Effects of age and gender on blogging., *in* ‘AAAI spring symposium: Computational approaches to analyzing weblogs’, Vol. 6, pp. 199–205.
- Schlichtkrull, M., Kipf, T. N., Bloem, P., Van Den Berg, R., Titov, I. & Welling, M. (2018), Modeling relational data with graph convolutional networks, *in* ‘European Semantic Web Conference’, Springer, pp. 593–607.
- Schmitz, M., Bart, R., Soderland, S., Etzioni, O. et al. (2012), Open language learning for information extraction, *in* ‘Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational natural language learning’, Association for Computational Linguistics, pp. 523–534.
- Schoene, A. & de Mel, G. (2019), Pooling tweets by fine-grained emotions to uncover topic trends in social media, *in* ‘2019 22th International Conference on Information Fusion (FUSION)’, IEEE, pp. 1–7.
- Schoene, A. M. (2020), Hybrid approaches to fine-grained emotion detection in social media data, *in* ‘Proceedings of the AAAI Conference on Artificial Intelligence’, Vol. 34, pp. 13732–13733.

- Schoene, A. M. & Dethlefs, N. (2016), Automatic identification of suicide notes from linguistic and sentiment features, *in* ‘Proceedings of the 10th SIGHUM Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities’, pp. 128–133.
- Schoene, A. M. & Dethlefs, N. (2018), ‘Unsupervised suicide note classification’.
- Schoene, A. M., Lacey, G., Turner, A. P. & Dethlefs, N. (2019), Dilated lstm with attention for classification of suicide notes, *in* ‘Proceedings of the Tenth International Workshop on Health Text Mining and Information Analysis (LOUHI 2019)’, pp. 136–145.
- Schoene, A. M., Turner, A. & Dethlefs, N. (2020), Bidirectional dilated lstm with attention for fine-grained emotion classification in tweets, *in* ‘Proceedings of the AAAI-20 Workshop on Affective Content Analysis, New York, USA, AAAI’.
- Schoene, A. M., Turner, A. P. & Dethlefs, N. (2019), ‘Dilated lstm with ranked units for classification of suicide notes’.
- Schoene, A. M., Turner, Alexander P, G. d. M. & Dethlefs, N. (n.d.), ‘Learning embedding representations using a linguistically inspired knowledge graph for emotion classification.’.
- Schuck, A. R. & Ward, J. (2008), ‘Dealing with the inevitable: strategies of self-presentation and meaning construction in the final statements of inmates on texas death row’, *Discourse & society* **19**(1), 43–62.
- Schuff, H., Barnes, J., Mohme, J., Padó, S. & Klinger, R. (2017), Annotation, modelling and analysis of fine-grained emotions on a stance and sentiment detection corpus, *in* ‘Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis’, pp. 13–23.

- Schuster, M. & Paliwal, K. K. (1997), 'Bidirectional recurrent neural networks', *IEEE Transactions on Signal Processing* **45**(11), 2673–2681.
- Sedding, J. & Kazakov, D. (2004), Wordnet-based text document clustering, *in* 'Proceedings of the 3rd workshop on RObusT Methods in Analysis of Natural Language Data (ROMAND 2004)', pp. 104–113.
- Shahreen, N., Subhani, M. & Rahman, M. M. (2018), Suicidal trend analysis of twitter using machine learning and neural network, *in* '2018 International Conference on Bangla Speech and Language Processing (ICBSLP)', IEEE, pp. 1–5.
- Shapero, J. J. (2011), The language of suicide notes, PhD thesis, University of Birmingham.
- Sharma, S., Kiros, R. & Salakhutdinov, R. (2015), 'Action recognition using visual attention', *arXiv preprint arXiv:1511.04119* .
- Shin, B., Lee, T. & Choi, J. D. (2017), 'Lexicon integrated cnn models with attention for sentiment analysis', *EMNLP 2017* p. 149.
- Shioiri, T., Nishimura, A., Akazawa, K., Abe, R., Nushida, H., Ueno, Y., KOJIKI-MARUYAMA, M. & Someya, T. (2005), 'Incidence of note-leaving remains constant despite increasing suicide rates', *Psychiatry and Clinical Neurosciences* **59**(2), 226–228.
- Shneidman, E. S. & Farberow, N. L. (1956), 'Clues to suicide', *Public health reports* **71**(2), 109.
- Socher, R., Chen, D., Manning, C. D. & Ng, A. (2013), Reasoning with neural tensor networks for knowledge base completion, *in* 'Advances in neural information processing systems', pp. 926–934.

- Socher, R., Lin, C. C., Manning, C. & Ng, A. Y. (2011), Parsing natural scenes and natural language with recursive neural networks, *in* ‘Proceedings of the 28th international conference on machine learning (ICML-11)’, pp. 129–136.
- Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A. Y. & Potts, C. (2013), Recursive deep models for semantic compositionality over a sentiment treebank, *in* ‘Proceedings of the 2013 conference on empirical methods in natural language processing’, pp. 1631–1642.
- Sowa, J. F. (2014), *Principles of semantic networks: Explorations in the representation of knowledge*, Morgan Kaufmann.
- Stanford (2017), ‘Large movie review dataset’, <http://ai.stanford.edu/~amaas/data/sentiment>. Accessed on 2017-02-10.
- Steiner, T., Verborgh, R., Troncy, R., Gabarro, J. & Van de Walle, R. (2012), Adding realtime coverage to the google knowledge graph, *in* ‘11th International Semantic Web Conference (ISWC 2012)’, Citeseer.
- Strapparava, C. & Mihalcea, R. (2007), SemEval-2007 task 14: Affective text, *in* ‘Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)’, Association for Computational Linguistics, Prague, Czech Republic, pp. 70–74.  
**URL:** <https://www.aclweb.org/anthology/S07-1013>
- Strapparava, C. & Mihalcea, R. (2008), Learning to identify emotions in text, *in* ‘Proceedings of the 2008 ACM symposium on Applied computing’, ACM, pp. 1556–1560.
- Strapparava, C., Valitutti, A. et al. (2004), Wordnet affect: an affective extension of wordnet., *in* ‘Lrec’, Vol. 4, Citeseer, pp. 1083–1086.

- Sukhbaatar, S., Weston, J., Fergus, R. et al. (2015), End-to-end memory networks, *in* ‘Advances in neural information processing systems’, pp. 2440–2448.
- Sun, C., Huang, L. & Qiu, X. (2019), Utilizing bert for aspect-based sentiment analysis via constructing auxiliary sentence, *in* ‘Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)’, pp. 380–385.
- Sun, C., Qiu, X., Xu, Y. & Huang, X. (2019), How to fine-tune bert for text classification?, *in* ‘China National Conference on Chinese Computational Linguistics’, Springer, pp. 194–206.
- Sun, Y., Wang, S., Li, Y., Feng, S., Chen, X., Zhang, H., Tian, X., Zhu, D., Tian, H. & Wu, H. (2019), ‘Ernie: Enhanced representation through knowledge integration’, *arXiv preprint arXiv:1904.09223* .
- Sun, Y., Wang, S., Li, Y.-K., Feng, S., Tian, H., Wu, H. & Wang, H. (2020), Ernie 2.0: A continual pre-training framework for language understanding., *in* ‘AAAI’, pp. 8968–8975.
- Sutskever, I. & Hinton, G. (2010), ‘Temporal-kernel recurrent neural networks’, *Neural Networks* **23**(2), 239–243.
- Sutskever, I., Vinyals, O. & Le, Q. V. (2014), Sequence to sequence learning with neural networks, *in* ‘Advances in neural information processing systems’, pp. 3104–3112.
- Suttles, J. & Ide, N. (2013), Distant supervision for emotion classification with discrete binary values, *in* ‘International Conference on Intelligent Text Processing and Computational Linguistics’, Springer, pp. 121–136.



- Suykens, J. A. & Vandewalle, J. (1999), ‘Least squares support vector machine classifiers’, *Neural processing letters* **9**(3), 293–300.
- Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J. & Wojna, Z. (2016), Rethinking the inception architecture for computer vision, *in* ‘Proceedings of the IEEE conference on computer vision and pattern recognition’, pp. 2818–2826.
- Tai, K. S., Socher, R. & Manning, C. D. (2015), ‘Improved semantic representations from tree-structured long short-term memory networks’, *arXiv preprint arXiv:1503.00075* .
- Tang, D., Qin, B. & Liu, T. (2015), Document modeling with gated recurrent neural network for sentiment classification, *in* ‘Proceedings of the 2015 conference on empirical methods in natural language processing’, pp. 1422–1432.
- Tang, D., Qin, B. & Liu, T. (2016), ‘Aspect level sentiment classification with deep memory network’, *arXiv preprint arXiv:1605.08900* .
- Tang, D., Wei, F., Qin, B., Yang, N., Liu, T. & Zhou, M. (2015), ‘Sentiment embeddings with applications to sentiment analysis’, *IEEE transactions on knowledge and data Engineering* **28**(2), 496–509.
- Tang, D., Wei, F., Yang, N., Zhou, M., Liu, T. & Qin, B. (2014), Learning sentiment-specific word embedding for twitter sentiment classification, *in* ‘Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)’, pp. 1555–1565.
- Tausczik, Y. R. & Pennebaker, J. W. (2010), ‘The psychological meaning of words: Liwc and computerized text analysis methods’, *Journal of language and social psychology* **29**(1), 24–54.
- Tensorflow (2020), ‘Tensorboard embedding projector’, [https://www.tensorflow.org/tensorboard/tensorboard\\_projector\\_plugin](https://www.tensorflow.org/tensorboard/tensorboard_projector_plugin). Accessed on 2020-10-10.

Texas Department of Criminal Justices (2019), ‘Texas death row executions info and last words’, [https://www.tdcj.state.tx.us/death\\_row/dr\\_executed\\_offenders.html](https://www.tdcj.state.tx.us/death_row/dr_executed_offenders.html).

Thakor, P. & Sasi, S. (2015), ‘Ontology-based sentiment analysis process for social media content’, *Procedia Computer Science* **53**, 199–207.

The Kernel (2013), ‘What suicide notes look like in the social media age’.

**URL:** <https://kernelmag.dailydot.com/features/report/6451/what-suicide-notes-look-like-in-the-social-m>

Tian, H., Gao, C., Xiao, X., Liu, H., He, B., Wu, H., Wang, H. & Wu, F. (2020), ‘Skep: Sentiment knowledge enhanced pre-training for sentiment analysis’, *arXiv preprint arXiv:2005.05635* .

Tibshirani, R. (1996), ‘Regression shrinkage and selection via the lasso’, *Journal of the Royal Statistical Society: Series B (Methodological)* **58**(1), 267–288.

Townsend, L. & Wallace, C. (2016), ‘Social media research: A guide to ethics’, *University of Aberdeen* .

Trouillon, T., Welbl, J., Riedel, S., Gaussier, É. & Bouchard, G. (2016), Complex embeddings for simple link prediction, *in* ‘International Conference on Machine Learning (ICML)’.

Tumblr (2013), ‘Suicide notes’.

**URL:** <http://suicide-notes.tumblr.com/>

Turk, A. M. (2012), ‘Amazon mechanical turk’, *Retrieved August 17, 2012*.

Twitter (2017a), ‘Api’, <https://developer.twitter.com/en/docs>. Accessed on 2017-10-10.

Twitter (2017b), ‘General’, <https://twitter.com/?lang=en-gb>. Accessed on 2017-10-10.

- Twitter (2017c), ‘Terms of services’, <https://twitter.com/en/tos>. Accessed on 2017-10-10.
- Twitter (2018a), ‘Counting characters’, <https://developer.twitter.com/en/docs/basics/counting-characters.html>. Accessed on 2018-11-11.
- Twitter (2018b), ‘Docs’, <https://developer.twitter.com/en/docs.html>. Accessed on 2018-11-11.
- Ugander, J., Karrer, B., Backstrom, L. & Marlow, C. (2011), ‘The anatomy of the facebook social graph’, *arXiv preprint arXiv:1111.4503* .
- UK, N. (2020), ‘Clinical depression’.  
**URL:** <https://www.nhs.uk/conditions/clinical-depression/causes/>
- Van Den Oord, A., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A. W. & Kavukcuoglu, K. (2016), Wavenet: A generative model for raw audio., *in* ‘SSW’, p. 125.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł. & Polosukhin, I. (2017), Attention is all you need, *in* ‘Advances in neural information processing systems’, pp. 5998–6008.
- Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P. & Bengio, Y. (2017), ‘Graph attention networks’, *arXiv preprint arXiv:1710.10903* .
- Vezhnevets, A. S., Osindero, S., Schaul, T., Heess, N., Jaderberg, M., Silver, D. & Kavukcuoglu, K. (2017), ‘Feudal networks for hierarchical reinforcement learning’, *arXiv preprint arXiv:1703.01161* .
- Vossen, P., Caselli, T. & Kontzopoulou, Y. (2015), Storylines for structuring massive streams of news, *in* ‘Proceedings of the First Workshop on Computing News Storylines’, pp. 40–49.

- Wang, B., Liakata, M., Zubiaga, A. & Procter, R. (2017), Tdparse: Multi-target-specific sentiment recognition on twitter, *in* ‘Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers’, pp. 483–493.
- Wang, J., Wang, Z., Zhang, D. & Yan, J. (2017), Combining knowledge with deep convolutional neural networks for short text classification, *in* ‘Proceedings of the 26th International Joint Conference on Artificial Intelligence’, pp. 2915–2921.
- Wang, J., Yu, L.-C., Lai, K. R. & Zhang, X. (2016), Dimensional sentiment analysis using a regional cnn-lstm model, *in* ‘Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)’, Vol. 2, pp. 225–230.
- Wang, K., Shen, W., Yang, Y., Quan, X. & Wang, R. (2020), ‘Relational graph attention network for aspect-based sentiment analysis’, *arXiv preprint arXiv:2004.12362* .
- Wang, Q., Huang, P., Wang, H., Dai, S., Jiang, W., Liu, J., Lyu, Y., Zhu, Y. & Wu, H. (2019), ‘Coke: Contextualized knowledge graph embedding’, *arXiv preprint arXiv:1911.02168* .
- Wang, Q., Mao, Z., Wang, B. & Guo, L. (2017), ‘Knowledge graph embedding: A survey of approaches and applications’, *IEEE Transactions on Knowledge and Data Engineering* **29**(12), 2724–2743.
- Wang, W., Chen, L., Tan, M., Wang, S. & Sheth, A. P. (2012), ‘Discovering fine-grained sentiment in suicide notes’, *Biomedical informatics insights* **5**, BII–S8963.
- Wang, X., Liu, Y., Sun, C.-J., Wang, B. & Wang, X. (2015), Predicting polarities of tweets by composing word embeddings with long short-term memory, *in*

- ‘Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)’, pp. 1343–1353.
- Wang, Z., Zhang, J., Feng, J. & Chen, Z. (2014), Knowledge graph embedding by translating on hyperplanes., *in* ‘Aaai’, Vol. 14, pp. 1112–1119.
- Wei, Y., Xia, W., Lin, M., Huang, J., Ni, B., Dong, J., Zhao, Y. & Yan, S. (2015), ‘Hcp: A flexible cnn framework for multi-label image classification’, *IEEE transactions on pattern analysis and machine intelligence* **38**(9), 1901–1907.
- Weiss, K., Khoshgoftaar, T. M. & Wang, D. (2016), ‘A survey of transfer learning’, *Journal of Big data* **3**(1), 9.
- Werbos, P. J. (1988), ‘Generalization of backpropagation with application to a recurrent gas market model’, *Neural networks* **1**(4), 339–356.
- Weston, J., Bordes, A., Yakhnenko, O. & Usunier, N. (2013), ‘Connecting language and knowledge bases with embedding models for relation extraction’, *arXiv preprint arXiv:1307.7973* .
- Weston, J., Chopra, S. & Bordes, A. (2014), ‘Memory networks’, *arXiv preprint arXiv:1410.3916* .
- WHO (2019), ‘Sustainable development goal 3’.  
**URL:** <https://sustainabledevelopment.un.org/sdg3>
- Widen, S. C., Russell, J. A. & Brooks, A. (2004), Anger and disgust: Discrete or overlapping categories, *in* ‘2004 APS Annual Convention, Boston College, Chicago, IL’.
- Wiegrefe, S. & Pinter, Y. (2019), ‘Attention is not not explanation’, *arXiv preprint arXiv:1908.04626* .

Wiki, A. (2019), ‘Semeval portal’.

**URL:** *https://aclweb.org/aclwiki/SemEval\_Portal*

Winograd, T. (1972), ‘Understanding natural language’, *Cognitive psychology* **3**(1), 1–191.

Wu, Z., Zheng, H., Wang, J., Su, W. & Fong, J. (2019), Bnu-hkbu uic nlp team 2 at semeval-2019 task 6: Detecting offensive language using bert model, *in* ‘Proceedings of the 13th International Workshop on Semantic Evaluation’, pp. 551–555.

Xiao, H., Huang, M., Hao, Y. & Zhu, X. (2015*a*), ‘Transa: An adaptive approach for knowledge graph embedding’, *arXiv preprint arXiv:1509.05490* .

Xiao, H., Huang, M., Hao, Y. & Zhu, X. (2015*b*), ‘Transg: A generative mixture model for knowledge graph embedding’, *arXiv preprint arXiv:1509.05488* .

Xiao, H., Huang, M. & Zhu, X. (2015), ‘From one point to a manifold: Knowledge graph embedding for precise link prediction’, *arXiv preprint arXiv:1512.04792* .

Xie, R., Liu, Z. & Sun, M. (2016), Representation learning of knowledge graphs with hierarchical types., *in* ‘IJCAI’, pp. 2965–2971.

Xu, H., Liu, B., Shu, L. & Yu, P. S. (2020), ‘Dombert: Domain-oriented language model for aspect-based sentiment analysis’, *arXiv preprint arXiv:2004.13816* .

Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R. & Bengio, Y. (2015), Show, attend and tell: Neural image caption generation with visual attention, *in* ‘International conference on machine learning’, pp. 2048–2057.

Xue, B., Fu, C. & Shaobin, Z. (2014), A study on sentiment computing and classification of sina weibo with word2vec, *in* ‘2014 IEEE International Congress on Big Data’, IEEE, pp. 358–363.

- Yang, B., Yih, W.-t., He, X., Gao, J. & Deng, L. (2014), ‘Embedding entities and relations for learning and inference in knowledge bases’, *arXiv preprint arXiv:1412.6575* .
- Yang, H., Willis, A., De Roeck, A. & Nuseibeh, B. (2012), ‘A hybrid model for automatic emotion recognition in suicide notes’, *Biomedical informatics insights* **5**(Suppl 1), 17.
- Yang, Z., Yang, D., Dyer, C., He, X., Smola, A. & Hovy, E. (2016), Hierarchical attention networks for document classification, *in* ‘Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies’, pp. 1480–1489.
- Yao, L., Mao, C. & Luo, Y. (2019a), Graph convolutional networks for text classification, *in* ‘Proceedings of the AAAI Conference on Artificial Intelligence’, Vol. 33, pp. 7370–7377.
- Yao, L., Mao, C. & Luo, Y. (2019b), ‘Kg-bert: Bert for knowledge graph completion’, *arXiv preprint arXiv:1909.03193* .
- Yaqi, X., Xu, Z., Meel, K. S., Kankanhalli, M. & Soh, H. (2019), Embedding symbolic knowledge into deep networks, *in* ‘Advances in Neural Information Processing Systems’, pp. 4235–4245.
- Yin, D., Meng, T. & Chang, K.-W. (2020), ‘Sentibert: A transferable transformer-based architecture for compositional sentiment semantics’, *arXiv preprint arXiv:2005.04114* .
- Yin, W., Schütze, H., Xiang, B. & Zhou, B. (2016), ‘Abcnn: Attention-based convolutional neural network for modeling sentence pairs’, *Transactions of the Association for Computational Linguistics* **4**, 259–272.

- You, Q., Jin, H., Wang, Z., Fang, C. & Luo, J. (2016), Image captioning with semantic attention, *in* ‘Proceedings of the IEEE conference on computer vision and pattern recognition’, pp. 4651–4659.
- Young, T., Hazarika, D., Poria, S. & Cambria, E. (2017), ‘Recent trends in deep learning based natural language processing’, *arXiv preprint arXiv:1708.02709* .
- Young, T., Hazarika, D., Poria, S. & Cambria, E. (2018), ‘Recent trends in deep learning based natural language processing’, *iee Computational intelligence magazine* **13**(3), 55–75.
- Yu, L.-C., Wang, J., Lai, K. R. & Zhang, X. (2017), Refining word embeddings for sentiment analysis, *in* ‘Proceedings of the 2017 conference on empirical methods in natural language processing’, pp. 534–539.
- Zayats, V. & Ostendorf, M. (2018), ‘Conversation modeling on reddit using a graph-structured lstm’, *Transactions of the Association for Computational Linguistics* **6**, 121–132.
- Zhang, C., Li, Q. & Song, D. (2019), ‘Aspect-based sentiment classification with aspect-specific graph convolutional networks’, *arXiv preprint arXiv:1909.03477* .
- Zhang, D. (2018), Big data security and privacy protection, *in* ‘8th International Conference on Management and Computer Science (ICMCS 2018)’, Atlantis Press.
- Zhang, X., Wu, J. & Dou, D. (2019), Delta embedding learning, *in* ‘Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics’, pp. 3329–3334.
- Zhang, Y., Liu, Q. & Song, L. (2018), ‘Sentence-state lstm for text representation’, *arXiv preprint arXiv:1805.02474* .
- Zhang, Z., Han, X., Liu, Z., Jiang, X., Sun, M. & Liu, Q. (2019), Ernie: Enhanced



- language representation with informative entities, *in* ‘Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics’, pp. 1441–1451.
- Zhao, X., Lin, S. & Huang, Z. (2018), Text classification of micro-blog's "tree hole" based on convolutional neural network, *in* ‘Proceedings of the 2018 International Conference on Algorithms, Computing and Artificial Intelligence’, pp. 1–5.
- Zhao, Z., Chen, W., Wu, X., Chen, P. C. & Liu, J. (2017), ‘Lstm network: a deep learning approach for short-term traffic forecast’, *IET Intelligent Transport Systems* **11**(2), 68–75.
- Zhou, J., Cui, G., Zhang, Z., Yang, C., Liu, Z., Wang, L., Li, C. & Sun, M. (2018), ‘Graph neural networks: A review of methods and applications’, *arXiv preprint arXiv:1812.08434* .
- Zhou, X., Wan, X. & Xiao, J. (2016), Attention-based lstm network for cross-lingual sentiment classification, *in* ‘Proceedings of the 2016 conference on empirical methods in natural language processing’, pp. 247–256.
- Zirikly, A., Resnik, P., Uzuner, O. & Hollingshead, K. (2019), Clpsych 2019 shared task: Predicting the degree of suicide risk in reddit posts, *in* ‘Proceedings of the Sixth Workshop on Computational Linguistics and Clinical Psychology’, pp. 24–33.