A Navigation System for Safe Routing

Ramneek Kaur*, Vikram Goyal*, Venkata M. V. Gunturi†, Aakanksha Saini*,
Kaushal Sanadhya*, Ritvik Gupta* and Siftee Ratra‡
*Indraprastha Institute of Information Technology Delhi, †Indian Institute of Technology Ropar,
‡ Indira Gandhi Delhi Technical University for Women
Email: [ramneekk, vikram, aakanksha19056, kaushal19133, ritvik18085]@iiitd.ac.in,
gunturi@iitrpr.ac.in, siftee088btit19@igdtuw.ac.in

***Abstract*—Globally, women are cautious when planning their routine travel routes. In a recent survey on street harassment, 82% of international respondents reported taking a different route to their destination than the conventional route due to fear of harassment. Such studies indicate an increasing need for '*Safe Routing*', especially in developing nations where the lack of infrastructure such as street lights, may contribute to higher crime rates. However, to the best of our knowledge, no state-of-the-art navigation system provides the option of '*Safe Routing*'. In this work, we propose a novel system that recommends "Safe Routes". Routes recommended by our system balance the conflicting requirements of *increasing the safety* and *constraining the total length* of the path to be within a reasonable limit (as desired by the user). From a theoretical perspective, the problem of '*Safe Routing*' can be modeled as the Arc Orienteering Problem, which is a well-known NP-hard combinatorial optimization problem.**

I. INTRODUCTION

Safety in public places has become an essential aspect of route planning these days, especially for women travelers in developing nations. A recent international survey on street harassment shows that several women prefer taking longer routes to their destination that are safer than the otherwise shorter routes suggested by navigation applications like Google Maps and Bing Maps [1]. The problem of finding safe routes becomes even more relevant in developing nations where the lack of infrastructure such as street lights, may contribute to higher crime rates. The annual report by the National Crime Records Bureau (NCRB) of India records 3672 cases of women abduction in the capital city of Delhi in 2019. Furthermore, safe navigation would also benefit tourists in planning their travel itineraries, by helping them avoid travelling through areas with higher criminal activity.

In this work, we put together a system that suggests "Safe Routes" to travelers. Our objective is to find a route between the source and the destination which has a high "safety level". This is in stark contrast to the state-of-the-art navigation systems that are focused on computing the shortest or the fastest route to the destination. It is important to note that the goal of maximizing "safety" may lead to fetching of longer routes. However, this may not be desirable as a traveler would not be willing to compromise on the length of the route indefinitely. Therefore, we model our problem as a constrained maximization problem defined as follows.

Problem formulation: Given as input, (1) a graph representing the road network, where each edge is associated with a *cost* value and a *safety score*, (2) a source s and a destination d (3) a threshold on the total cost of the solution, the *budget* value; the goal is to find a route from s to d that maximizes the sum of *safety scores* of its edges, where the sum of *costs* of edges of the route is no more than the input *budget* value.

Our problem can be reduced to an instance of the Arc Orienteering Problem, which is a well known NP-hard problem [2]. The problem is challenging due to the conflicting requirements of maximizing the safety score while constraining the cost. As explained in [3], trivial modifications of minimization based strategies (e.g., Dijkstra's algorithm and its variants) would not be suitable for our problem. Some researchers (e.g., [4] and [5]) have started exploring the concept of safe routing. However, they do not model the problem as a constrained maximization problem, which is essential for our application domain.

Contributions: We build a navigation system for computing "Safe Routes", that is suited for vehicular navigation. For our study, we targeted the city of Delhi. An essential input to a "Safe Routing" system is data that can be reliably used as a proxy for public safety at the street level. To compute the street level safety, we extracted data from various publicly available resources like government websites, news websites, and social media platforms. To transform the data into a usable form for safe routing, we applied processing techniques using tools from machine learning, natural language processing, and geocoding. For computing the "Safe Routes", we apply the algorithms proposed in an earlier work [3].

II. SYSTEM OVERVIEW

A. Data sources

The data sources we use can broadly be classified into four categories: crime data, public transport data, social media data, and image data (as shown in Figure 1). More specifically, we extract the following data: (1) **Crime data from news websites**: We scrape news articles for crime incidents for the period 2016 to 2020 for the region of Delhi from the Times Of India (TOI) website [6]. (2) **Crime statistics data from the NCRB website**: We consider statistical data from the 2019 report for district-wise crime rates in Delhi [7]. (3) **Public transport data**: We collect data from the Delhi Integrated Multi-Modal Transit System Ltd. website to get the live locations of the public transport buses in Delhi [8]. This data is processed to find areas with a poor GPS connectivity

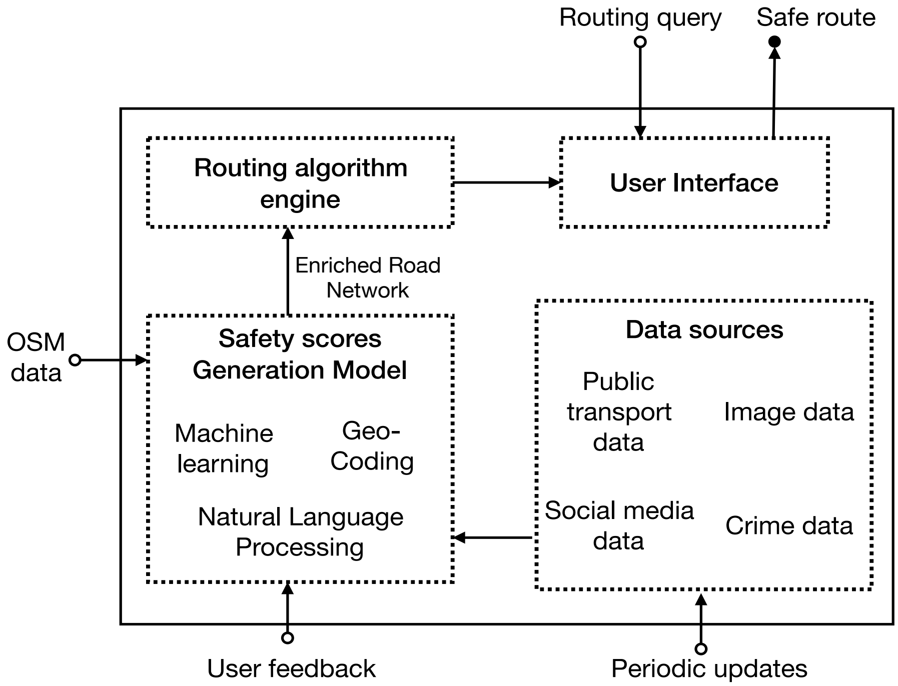
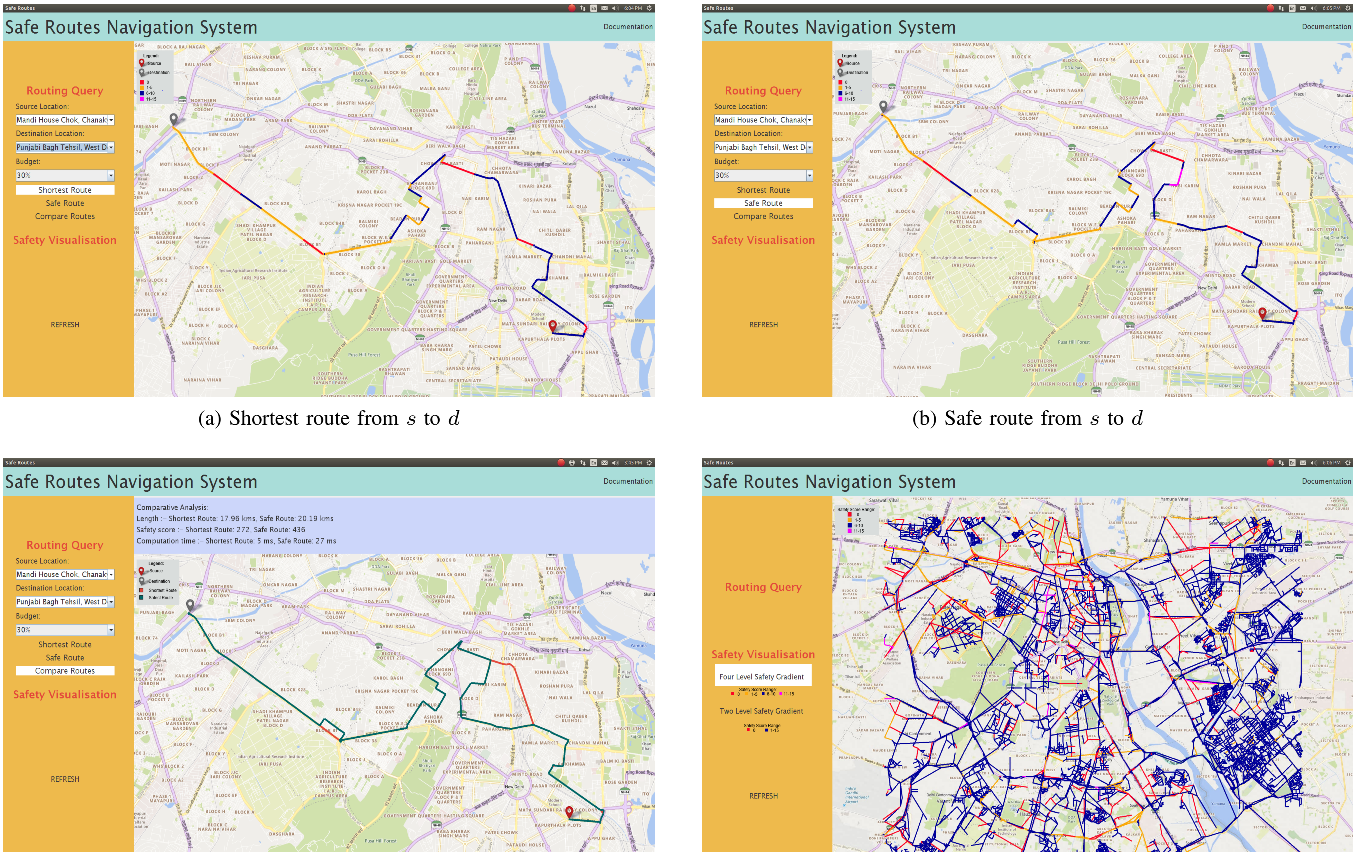


Fig. 1: System Overview

(4) **Traffic data**: We scrape tweets from the Twitter handle of Delhi Traffic Police [9] to get the dynamic congestion-related information on a daily basis. (5) **Image data**: We fetch images for streets across Delhi (uploaded mostly by the general public) using the Google Static Street View API. (6) **Road quality data**: This data was downloaded from the Public Works Department (PWD) website [10] to gather the complaints registered by citizens regarding the quality of streets, which is updated every month. We extracted complaints related to the following safety-related issues: malfunctioning street lights, poor sanitation (open potholes and drain covers) and poor quality road segments (broken flyovers, roads and footpaths). (7) **Police stations coordinates**: We also consider the locations of Police stations across the city as one of the parameters for safety score computation.

B. Data processing

To gauge the level of safety on the streets, we process the extracted data using different techniques. The details of this phase are as follows: (1) **Image data**: The street view images were first classified into two categories, indoor and outdoor images. The outdoor images were further processed to predict their scenic attributes, such as whether the site is a residential area, a construction site, a bus stop, a slum area, etc., from a fixed list of attributes. These tasks were done using the Places-365 CNN [11]. (2) **Road quality data**: We identified two patterns in the complaints registered on the PWD website, complaints that refer to a single location (e.g., the street light is not working at location X), and complaints that refer to more than one locations (example: street lights are not working from location X to location Y). To extract the location(s) from the text, we performed Named Entity Recognition using the SpanBERT model [12] trained on addresses of Delhi. The model generates a tag for each word of the sentence, and identifies the words representing spatial locations. Finally, we used HERE Geocoding Search API [13] to find the longitude and latitude of the specified location. (3) **Public transport data**: We process the live feed coming from the public transport buses in Delhi to find areas with a poor GPS connectivity. Buses (belonging to Delhi Municipality) send signals (their current location) to a central server at periodic time intervals using a GPS system which is mounted on them by the Delhi Muncipality authorities. We track the areas where the buses do not send signals by evaluating the time lapse between successive signals. The areas that are reported by at least ten buses are considered as having a poor GPS connectivity. (4) **Traffic data**: Most of the tweets scraped from the Twitter handle of Delhi Traffic Police were not geotagged. We geotag these tweets using the SpanBERT model and the HERE Geocoding Search API, to compute the geo-coordinates of congested locations from the textual description. (5) **Crime data from TOI**: To process this data, we first lemmatise the text in the news articles using the spaCy library [14]. We identify the location of the crime incident using the SpanBERT model and the HERE Geocoding Search API. We also extract the type of crime incident using the Smote classifier [15]. The crime incidents in Delhi were found to belong to fourteen categories, viz. riot, rape, murder, kidnapping, trafficking, assault, scam, robbery, cybercrime, harassment, missing person, roadrage, gangwars and terrorism. In our model for safety scores generation, we do not use the information regarding the crime type. We only



(a) Shortest route from s to d

(b) Safe route from s to d

(c) Comparative analysis of the two routes

(d) Four Level Safety Gradient Visualization

Fig. 2: Safe Routes Navigation System Demonstration

use the information regarding the presence or absence of a crime incident at a specific location.

C. Safety scores generation

The input to this module includes the road network data (from OpenStreetMap (OSM)) and the safety-related data after the processing phase described in Section II-B. In this phase, we compute the impact of the previously mentioned data sources on the safety scores of the road segments. For e.g., proximity to a police station or being a part of a residential area positively affects the safety score of a road segment. In contrast, if the road segment passes through deserted regions or does not have proper lighting, its safety score decreases. The details of the computation of the safety scores of the road segments are described next.

First, each relevant data point from the previously mentioned data sources is map-matched to the road segments present in its spatial neighborhood. For example, in case of Police stations data source, we attach the locations of all the police stations (e.g., Pahar Ganj Police station located at 28.6456° N, 77.2089° E) to their nearby road segments. After all the relevant data points have been mapped to their corresponding road segments present in their spatial neighborhood, we compute the safety score of each road segment.

TABLE I: Dataset Statistics

City	Vertices	Edges	Safety score range
Delhi	11,399	24,803	[0,15]

Initially, we assign a safety score of eight to all the road segments (median in the range 1 through 15). Following this, we consider the impacts of the data points mapped to the road segment. For example, if a road segment has a police station nearby (mapped to it), its safety score value is incremented by value three. On the contrary, if the road segment passes through a deserted region (if indicated by one of the attributes of image data post the processing phase), its safety score value is decremented by value two. Likewise, we compute the impacts of all the data points on a road segment to compute its safety score. If the computed score is less than zero, we set its value to zero. This is done to ensure that the safety scores are non-negative (a requirement of our route generation algorithm). Note that we directly assign a score value of zero to all the road segments that have some crime incident data point mapped to them (least safety score value). In such a case, we don't consider the other data sources (e.g., police stations, etc.) for determining the safety score of the particular road segment. The range of the computed safety scores is [0,15]. Table I details the statistics of the dataset constructed.

D. Route generation

The road network enriched with the safety scores is passed to the Routing algorithm engine module. "Safe Routes" are computed using the **Multiple Segment Replacement** algorithm proposed in [3]. The algorithm employs a four-step framework as depicted in Figure 3: (i) finding a seed path from s to d, (ii) estimating the potential of segments of the seed path for safety score improvement, (iii) selecting a set of segments for replacement with the objective of improving the *safety score*, (iv) replacing the chosen segments in the seed path while checking the *budget* constraint. The query parameters (s , d, algorithm and *budget*) are specified by the user through the User Interface (UI). The UI then shows a visualization of the resulting routes, the Shortest Route and the "Safe Route".

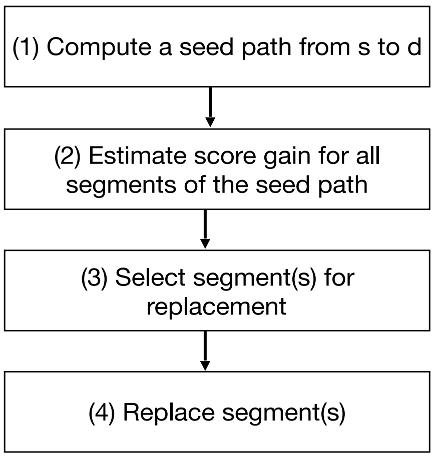


Fig. 3: Algorithmic framework

III. DEMONSTRATION OUTLINE

Figure 2 demonstrates the UI of the Safe Routes Navigation System, designed in Java. The features offered by the system include routing queries and safety visualization. As part of the **Routing Query** feature, the user can submit a routing query and view the shortest route and the "safe route" between the desired source and destination. The budget parameter can be varied as per the user requirement. We denote the budget parameter as a percentage of the length of the shortest route. A budget value of 10% implies the constraint that the length of the safe route should not exceed the length of the shortest route by a factor of 10%. The shortest and the safe routes for a sample query are shown in Figures 2a and 2b respectively. The road segments in these figures are coloured to denote their level of safety, as per the colour coding scheme shown in the legend (top-left corner of the map region). The user can also view a comparative analysis of the two routes (as depicted in Figure 2c). This option allows the user to view the lengths of the two routes, their safety scores, and the computation times for computing the shortest and the safe routes.

Under the **Safety Visualization** feature, the user can view the distribution of safety levels across the road network. The two level safety gradient shows a two-color visualization with different colours for edges with zero and non-zero scores respectively. Similarly, the four level safety gradient illustrates a four-colour visualization of the edges in the road network (as depicted in Figure 2d). Here, the road segments are coloured to denote their level of safety, as per the colour coding scheme shown in the legend. Lastly, the user can **Refresh** the map to clear previous results, and also view the **Documentation** regarding application usage.

ACKNOWLEDGEMENT

This work was in part supported by the Infosys Centre for Artificial Intelligence at IIIT-Delhi, Visvesvaraya Ph.D. Scheme for Electronics and IT, and DST-SERB(ECR/2016/001053).

REFERENCES

[1] B. Livingston, M. Grillo, and R. Paluch. (2014) Cornell international survey on street harassment. [Online]. Available: https://www.ilr.cornell.edu/worker-institute/news/ilr-and-hollaback-release-largest-analysis-street-harassment-date
[2] D. Gavalas and et al., "Approximation algorithms for the arc orienteering problem," *Info. Processing Letters*, vol. 115, no. 2, pp. 313 – 315, 2015.
[3] R. Kaur, V. Goyal, and V. M. V. Gunturi, "Finding the most navigable path in road networks," *GeoInformatica*, vol. 25, pp. 207–240, Jan. 2021.
[4] J. Kim, M. Cha, and T. Sandholm, "Socroutes: Safe routes based on tweet sentiments," ser. WWW '14 Companion. ACM, 2014, p. 179–182.
[5] A. M. de Souza, T. Braun, L. C. Botega, R. Cabral, and L. A. Garcia, Islene C. and. Villas, "Better safe than sorry: a vehicular traffic re-routing based on traffic conditions and public safety issues," *Journal of Internet Services and Applications*, vol. 10, Sep. 2019.
[6] Times of india: Delhi crime news. [Online]. Available: https://timesofindia.indiatimes.com/city/delhi
[7] NCRB. Crime in india table contents. [Online]. Available: https://ncrb.gov.in/en/crime-in-india-table-addtional-table-and-chapter-contents?field_date_value[value][year]=2019&field_select_table_title_of_crim_value=All&items_per_page=All
[8] Open transit data: Delhi. [Online]. Available: https://opendata.iiitd.edu.in/documentation/
[9] "Delhi traffic police: Delhi." [Online]. Available: https://twitter.com/dtptraffic
[10] Public works department: Govt. of nct of delhi. [Online]. Available: https://www.pwddelhi.gov.in/
[11] B. Zhou and et al., "Learning deep features for scene recognition using places database," in *Proc. of NIPS*, 2014, p. 487–495.
[12] M. Joshi, D. Chen, Y. Liu, D. S. Weld, L. Zettlemoyer, and O. Levy, "Spanbert: Improving pre-training by representing and predicting spans," 2020.
[13] "Here geocoder." [Online]. Available: https://developer.here.com/c/geocoding
[14] "spacy." [Online]. Available: https://spacy.io/
[15] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: Synthetic minority over-sampling technique," *J. Artif. Int. Res.*, vol. 16, no. 1, p. 321–357, Jun. 2002.