Abstract— In this paper, we present an intelligent, state-of-the-art, mobile-based transportation system called SAFAR (Safe and Fast around the Road), which provides dynamic information to Karachi bus commuters concerning any type of violence incident which has occurred farther ahead from their current location on the current bus route. Using named entity recognition techniques, we have trained SAFAR to recognize the location and method of violence incident along with the casualty information (if available) from live Twitter news feeds. Using the well-known A* heuristic search algorithm, SAFAR also recommends alternative routes to reach the destination in case of any violence up ahead. SAFAR has a competitive violence detection accuracy of 80% on a test corpus as well as in an online evaluation with real users. Finally, a subjective evaluation of these users reveals satisfactory performance of SAFAR across several dimensions.

Index Terms— Bus Commuters, Expert System, Android, Violence Detection, Named Entity Recognition, Twitter

I. INTRODUCTION

The city of Karachi (Pakistan) is the third largest city in the world and the eleventh largest urban agglomeration, boasting a population of 23.5 million [1]. The public transport system in Karachi offers diverse types of buses, which are used by millions of citizens of Karachi every day to reach their workplaces located far from residential areas. Unfortunately, Karachi has witnessed a remarkable increase in civilian violence activities over the past decade, which primarily leads to strikes that shut down different parts of the city. On a call for strike, the Karachi transport administration terminates bus services in sensitive locations and consequently, a large majority of bus commuters are hampered from reaching their workplaces on time, which causes a loss in billions to the national exchequer; one strike day in Karachi causes a loss of USD 30 million to Pakistan's economy. This is critical because Karachi's contributes USD 153 million to GDP daily [2]. The commuters also suffer from salary deductions personally, as the workplaces are located in safer areas of the city which remain undisturbed during strikes. The local transport authority is Karachi Urban Transport Corporation (KUTC), and it has not taken any concrete initiative to address the violence-related issues. This leaves the commuters uninformed about the routes of active (running) buses, and delays in bus timings, and forces many to spend hours on over-crowded buses to reach their destinations with a risk to their lives. Considering these problems, there is a clear need for a technological solution to assist Karachi commuters during strike situations.

Machine Learning has been widely applied in many application domains such as banking [3, 4], e-commerce systems [5], software testing [6], biomedical domain [7], transportation systems [8], etc. In this paper, we have applied machine learning and natural language processing techniques to propose an intelligent transportation system called SAFAR which will address the above mentioned need. SAFAR, an abbreviation for “Safe and Fast around the Road” is an intelligent, and android mobile-based transportation system for Karachi bus commuters. Its primary aim is to assist commuters by pre-informing them of violence incidents farther ahead on their current route, and consequently suggesting alternate routes to reach the destination.

II. RELATED WORK

Several research works from the domain of intelligent transportation systems have been carried out which are...
related to SAFAR. Perhaps the work most related to SAFAR is the one linking GPS-based vehicle-location and Twitter analysis [9]. Here, the authors analyze geographically-coded Twitter streams to update information regarding the movement of motorbikes. For validation, a set of 500 electric bikes equipped with GPS were used in the metropolitan area of Stuttgart (Germany) from July 2010 till August 2011. The bike locations were recorded every 30 seconds. The authors accumulate GPS data to identify popular bike routes which display frequently-tweeted words at different locations, hence helping to understand the changing sentiments and opinions of users as they ride the bikes.

Another important work is the re-designing of the public bus transportation system of Abidjan, Ivory Coast by IBM [10]. The system is called AllAboard and it optimizes the transportation network based on mobile usage patterns collected from millions of users of telecommunications provider Orange. Abidjan has around 500 large buses, 5000 minibuses and more than 12000 taxis. IBM collected location data for every phone call usage or text message for a two week period, applied extensive data cleaning operations to filter out the noise and extracted travel sequences from clean data. These sequences were fed into an optimization model to output bus routes which meet the commuters' travel demands and enhance urban mobility.

Besides this, NetBus is a comprehensive application developed for bus commuters in Israel (Susan, 2013). It combines three features: 1) real-time data regarding bus arrival times, 2) monitoring bus travel progress on a map, and 3) dynamic reminders and updates regarding the bus location. Every bus in Israel is equipped with GPS functionality. As the buses move, the location information is updated to the Ministry of Transport every 20 seconds. NetBus uses this data to estimate the arrival times to bus stops farther ahead, taking into account important heuristics such as traffic jam, traffic lights, and road conditions.

In the study [11], authors developed an intelligent transportation system that predicts road traffic condition in London in real-time using Twitter feeds from the Twitter account TfLTraficNews, which provides dynamic updates about London's traffic. The information extracted from tweets is combined with Google's traffic information, and finally displayed using Google Maps (maps.google.com). Traffic incidents which are happening in the same area, or on the same road, are clustered together for better visibility. This system is a cheaper and more cost-effective solution as compared to the current traffic sensor system of London.

III. METHODOLOGY

A. Data Collection and Pre-processing:

To train the SAFAR, we have collected tweets from Twitter. We extracted tweets from seven different Twitter accounts (shown in Table 1) of well-known and authentic news channels of Pakistan. We extracted tweets through Twitter4J API [12, 13]. In total, we collected 29601 from September 1, 2015 till December 31, 2015, and sampled 22000 for training and 7601 for testing. After data collection, we preprocessed collected tweets. In preprocessing, we tokenized the tweets into unigram tokens and then removed the stopwords from the set of tokens. We also removed the word “RT” (Retweet), name of the Twitter user, URLs and timestamp data. Stemming and lemmatization was also performed on tokens to reduce inflectional forms and sometimes derivationally related forms of a word to a common base form.

B. Creation of training set

After preprocessing, we manually labelled each word within the tweet with one of the following four labels to create the training set:

- **MET:** This is annotated to words representing the method of violence activity, e.g., ‘firing’, ‘bomb blast’, ‘gunshot’ etc.

- **CAS:** This is annotated to words representing casualty information, e.g., ‘killing’, ‘killed’, ‘shot dead’, ‘dead’, ‘injured’ etc.

- **LOC:** This is annotated to words representing the particular locations where the violence incident occurred; all annotated locations were a subset of the 2258 bus stops which we have extracted previously.

- **O:** All the tweets which do not annotated as Method, Casualty and Location; these are annotated as others.

An example of annotated tweets is shown in Figure 1. Here, ‘accident’ is labeled as the method (MET), ‘defence’ as locations (LOC) and 'injured' and 'died' as casualty (CAS).

<table>
<thead>
<tr>
<th>Twitter Account</th>
<th>No. Of Collected Tweets</th>
</tr>
</thead>
<tbody>
<tr>
<td>Geo News English</td>
<td>4159</td>
</tr>
<tr>
<td>Samaa TV</td>
<td>4350</td>
</tr>
<tr>
<td>Express News PK</td>
<td>4204</td>
</tr>
<tr>
<td>Aaj TV</td>
<td>4155</td>
</tr>
<tr>
<td>Dawn News</td>
<td>4355</td>
</tr>
<tr>
<td>PK News</td>
<td>4022</td>
</tr>
<tr>
<td>Express Tribune</td>
<td>4356</td>
</tr>
</tbody>
</table>

Table 1: Twitter accounts and frequency of collected tweets
C. Construction of SAFAR classifier

After the creation of training set, classifier was constructed for SAFAR which will classify the tweet as violent or non-violent. For violence recognition, we employed Named Entity Recognition (NER) technique [14] through the Java-based LingPipe library [15][16]. NER is a supervised learning technique and requires annotated text for classification [14].

The SAFAR classifier has the goal of detecting the violence zone, based on detecting MET, LOC and CAS labels from a tweet. For this, we implemented a rule-based expert system which classifies zone according to the pseudo code shown in Figure 2, and works as follows:

- If only LOC is detected, this cannot directly imply that any violence has occurred at that location and hence, this cannot be tagged as a violence zone (variable detect_violence_zone is false).
- If both LOC and MET are identified, then we can confidently imply that some violence has occurred at this location, and hence, this is tagged as a violence zone.
- If CAS is also detected along with MET and LOC, this is tagged as a violence zone (using the previous argument)
- If only LOC and CAS is detected, we consider this information inconclusive and hence, this is not tagged as a violence zone.

For instance, consider the tweet “In North Nazimabad, 25 people were killed last year when a bus overturned into a ditch”. As we don’t classify accidents as violence-related information, this tweet would wrongly classify North Nazimabad as a violence zone if we don’t follow the last rule. Moreover, when SAFAR detects a violence zone, we add it to the violence zone list which is always available to SAFAR users to identify unsafe areas.

```
if (LOC == true AND MET == true AND CAS == true) then
    predict_violence_zone = true
else if (LOC == true AND MET == true AND CAS == false) then
    predict_violence_zone = false
else if (LOC == true AND MET == false AND CAS == false) then
    predict_violence_zone = false
else if (LOC == true AND MET == false AND CAS == true) then
    predict_violence_zone = false
end if
```

Figure 2: Pseudocode of rule based engine to detect violence zone

For NER classifier training, we used the standard 10-fold cross validation process. We used the standard NER approach of first-best named entity chunking, which outputs the first best match for an annotated label as compared to a set of matches with different likelihoods. Here, we used the Hidden Markov Model (HMM) for performing chunking of annotated character sequences. We consider HMM to be appropriate because it can easily construct models for sequences of words through a (seemingly infinite) sequences of (hidden) nodes and links. It also makes training simpler by considering the Markov assumption, i.e., the conditional probability distribution of the future states (tweet words in our case) is dependent only on the current state (word) and not on the sequence of preceding words, which will make the detection of annotated words even more complicated. (for more details on HMMs, refer to [17],[18],[19]).

D. Results:

The results for our classifier training are shown in the second and third columns of Table 2. Although not of our interest, the highest precision is obtained for label ‘O’ (97.9%) because of considerably more O labels in training data. The next best precision is obtained for MET (90.8%) followed by CAS (89.4%) and LOC (88.5%). These values are competitive and acceptable as compared to other works on evaluating NER accuracy across standard NERs [14, 20]. The highest recall is obtained for ‘O’ class (99.10%), which is again not interesting for us. This is followed by CAS (92.5%), MET (83.6%) and LOC (74.3%). A high recall for CAS is important because casualty is a critical information in SAFAR. However, the recall for LOC is less (as compared to others) which is a limitation, and implies that we need to use more LOC-labeled data in the future for learning this concept. For the time being, we continue with our current training set to gauge SAFAR’s performance with live users.

We then evaluated SAFAR on the testing corpus, whose results are shown in the third and fourth column of Table 2. As expected highest precision and recall are obtained for label O. This is followed by MET (84.2%), CAS (80.5%) and LOC (75.5%), which is the same sequence as that of training performance. However, these precision values are, at the most, 15% less than corresponding training value. This demonstrates a minutely reduced performance on testing data. The recall values follow the same trend as above, and are at most 7% less than the corresponding training recall values. As is true for training data, the testing performance is also comparable with NER performances on other standard recognizers [14, 20].

<table>
<thead>
<tr>
<th>Label</th>
<th>Precision (Training)</th>
<th>Precision (Testing)</th>
<th>Recall (Training)</th>
<th>Recall (Testing)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MET</td>
<td>90.8</td>
<td>84.2</td>
<td>83.6</td>
<td>83.7</td>
</tr>
<tr>
<td>CAS</td>
<td>89.4</td>
<td>80.5</td>
<td>92.5</td>
<td>85.4</td>
</tr>
<tr>
<td>LOC</td>
<td>88.5</td>
<td>75.5</td>
<td>74.3</td>
<td>72.2</td>
</tr>
<tr>
<td>O</td>
<td>97.9</td>
<td>98.5</td>
<td>99.1</td>
<td>97.9</td>
</tr>
</tbody>
</table>

Table 2 Precision and recall for MET, LOC, CAS and O for training data and testing data

IV. DEVELOPMENT OF SAFAR- A MOBILE EXPERT SYSTEM

This section discusses the development of SAFAR application. SAFAR users provide their source location, destination location and bus name using GPS-enabled Android phones. SAFAR will then attempt to dynamically
detect violence zones between given source and destinations through a web server. If no violence zone is detected, SAFAR classifies the route as “safe”, and if one or more violence zones are detected, then SAFAR classifies the route as “unsafe”. In case of unsafe route, SAFAR recommends an alternate route to users. The modular view of SAFAR is shown in Figure 3.

![Figure 3: Modular view of SAFAR architecture](image)

Here, the User Interface module (designed in Android Studio) provides the GUI to SAFAR users. Specifically, the home page of SAFAR is shown in Figure 4(a), which offers four services to its users. Particularly, Tweet Extraction service allows users to extract tweets at run-time and the User Manual service displays the necessary help features. The Bus Root Information service provides information about bus routes. Its interface is shown in Figure 4 (b), in which the user can select the particular bus number from a drop-down list (bus N1) and the corresponding route is shown on Google Maps using Google Map Android API [21] where A and B are the starting and destination bus stops respectively as shown in Figure 4 (c). We collected information about the buses routes from several local web references [22-24]. From these, we identified 104 public buses and 2258 unique bus stops (a large number of bus stops are common to more than one bus). Finally, the Violence Detection service serves as the core of Violence Zone Detection module which is responsible for using NER to detect violence zones (as described in previous section). The two possible outputs of this service are shown in Figure 4 (d) and Figure 4 (e); the former shows that the currently selected route (from Al-Asif Square to Jail Chowrangi) is safe, while the latter shows that it is unsafe along with the location and reason of violence. In case of unsafe route, the user has the option to request for an alternate route (shown in Figure 4 (f)), which we calculate in the Alternate Route Recommendation module through the A* search algorithm at run-time and display using Google Map Android API. The task of the Violence Zone Matching module is to compare the violence zones (detected in the training and testing module) with the actual bus stops stored in our bus database. This involves a simple searching procedure to locate the identified violence zone within the list of available bus stops.

For recommending alternative route, we used the A* search algorithm [25] which is widely used in path-finding problems. This module is responsible to dynamically detect violence zones between given source and destinations. If no violence zone is detected, SAFAR classifies the route as “safe”, and if one or more violence zones are detected, then SAFAR classifies the route as “unsafe”. In case of unsafe route, SAFAR recommends an alternate route to users through the A* search algorithm at run-time and display the alternative route using Google Map Android API.

V. ONLINE VALIDATION AND SUBJECTIVE EVALUATION

We evaluated the online (run-time or dynamic) performance of SAFAR with 50 real users over a period of two months, from 1st January 2016 till 31st 29th February 2016. We obviously had no previous information about the violence incidents; they occurred at run-time. To maximize the chance of occurrence of violence, we selected bus routes in the more violent areas of Karachi, i.e., those locations in which a majority of violence incidents were detected during the training and test phases. These areas are Korangi, Lyari, Orangi Town, SITE area and Baldia Town. In each area, we selected 10 resident students of that area who had GPS-enabled Android phones. Thus, the total number of students was 50. These students used SAFAR in their homes mostly in the later part of the day for the two month period. Each student selected a bus route near to her home, and entered the source destination as the starting bus stop of this route, and the destination as the terminus of this route (the student’s home was near to some bus stop between the starting source and destination). During the evaluation period, SAFAR detected several violence incidents which occurred further ahead than the students’ homes. This performance is shown in Figure 5, which displays the frequency of actual and SAFAR-detected incidents in each area. For instance, in Korangi, in the reporting time period, three violent incidents happened, while SAFAR detect 2 incidents out of 3. Hence, 100% incidents were detected in Orangi Town and Baldia Town, 80% in Lyari, 75% in SITE and 67% in Korangi. The average accuracy is ~80%, which is close to performance obtained on test corpus.

We also evaluated the subjective performance of SAFAR by distributing a questionnaire to our 50 real users. This consisted of six questions (shown Figure 6) concerning critical evaluation parameters, specifically SAFAR’s level of innovation, violence detection accuracy, GUI, practicality of alternate routes, users’ overall satisfaction, and feasibility for generic deployment. These are to be rated on the standard 5-point Likert scale from Very Poor (a rating of 1) to Very Good (a rating of 5). The average responses of these questions for our 50 users are shown in Table 3. As can be seen, the highest rating (4.5) is obtained for SAFAR’s innovativeness and practicality for the commuters. The lowest rating is obtained for the SAFAR’s GUI (3.1).
Figure 4: Different Snapshots of SAFAR's GUI

a) Main GUI  
b) List of Buses  
c) Route of selected bus

d) Safe route  
e) Unsafe route  
f) Alternate route from Al-Asif to Jail Chowrangi

Figure 5: Actual and detected violence activities

Figure 6: Questions for SAFAR's subjective evaluation

a) What's the level of innovation of SAFAR?  
b) How was SAFAR’s performance in detecting the violence incidents?  
c) How will you rate the GUI of SAFAR?  
d) What’s the practicality of alternate routes suggested by SAFAR?  
e) What’s your overall level of satisfaction with SAFAR?  
f) What’s the practicality of SAFAR for real-time bus commuter usage in Karachi?
Table 3 Average responses of subjective questionnaire for 50 users

<table>
<thead>
<tr>
<th>Question Topic</th>
<th>Avg. Rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAFAR’s GUI</td>
<td>3.1</td>
</tr>
<tr>
<td>Alternate Route</td>
<td>3.5</td>
</tr>
<tr>
<td>Violence Zone Detection</td>
<td>4.1</td>
</tr>
<tr>
<td>Satisfaction</td>
<td>4.2</td>
</tr>
<tr>
<td>Practicality</td>
<td>4.5</td>
</tr>
<tr>
<td>Innovativeness</td>
<td>4.5</td>
</tr>
</tbody>
</table>

A manual interview with a sample of 20 users revealed that the GUI for alternate route and displaying violent information is not interactive and usable. The alternate route feature also received a comparatively average (3.5), primarily because our users already knew about the shortcut routes (at street level) to reach the specified destination in their residential areas (where the evaluation was taking place). Compared to street dynamics, the alternate A* routes are obviously non-optimal. We don’t consider this a limitation as the A* routes will still be useful for non-residents of a given area where the violence occurred. Nevertheless, an encouraging result is that users were satisfied with the overall performance of SAFAR (4.2).

VI. CONCLUSION AND FUTURE WORK

This paper presented an intelligent, state-of-the-art mobile-based transportation system called SAFAR to aid Karachi bus commuters in the violent situations in the city. SAFAR acquires dynamic information from seven Twitter accounts related to Karachi news. From each tweet, it attempts to dynamically recognize the mentioned location, the method of violence and casualty information by using named entity recognition techniques. It uses an expert system which uses these parameters to determine violence farther ahead on the current bus route of the bus commuters. In case of violence, SAFAR also provides an alternate route to reach the user’s destination safely by using the standard A* path-finding algorithm. Offline and online tests reveal a violence detection accuracy of ~80, which is considered acceptable based on results of other standard named entity recognizers. Moreover, a subjective questionnaire revealed that users considered SAFAR to be an innovative and practical solution and were satisfied with its performance.

In future work, we will improve the accuracy of SAFAR by using deep learning and ontology base concepts. Moreover, we will extend SAFAR for iPhone users and windows phone users. We will also improve the GUI of SAFAR.

REFERENCES