Incident Detection From Social Media Targeting Indian Traffic Scenario Using Transfer Learning

Priyambada Ambastha
Department of Computer Science and Engineering
Indian Institute of Technology Hyderabad
Hyderabad, India
cs18mtech11025@iith.ac.in

Maunendra Sankar Desarkar
Department of Computer Science and Engineering
Indian Institute of Technology Hyderabad
Hyderabad, India
maunendra@cse.iith.ac.in

***Abstract*—Road traffic congestion is one of the most challenging problems in densely populated cities. This paper aims to address this problem by developing a system to detect traffic congestion in India using Twitter. Twitter has been gaining momentum for research in congestion event detection for past several years because many commuters, as well as traffic authorities, tend to post traffic-related updates in real-time. There is no such traffic-tweet dataset for the Indian traffic scenario. We develop one such dataset that contains traffic-related posts concerning different Indian regions. The dataset contains posts that talk about traffic incidents such as accidents, infrastructure damage, and also about future planned events that can impact traffic flow. We call our dataset as L-TWITS (Labelled-TWeets for Indian Traffic Scenario).**

Basic practice in literature for traffic event detection problems is to collect a large amount of data, its annotation and then further analysis for event extraction. Such approaches often require a considerable amount of time for labelling the data. To address this shortcoming the proposed method uses a Transfer learning-based classifier that generally performs well even with less data. ULMFiT model has been used as a Transfer Learning approach for classifying the tweet samples into "Traffic incident related" or "Non-Traffic incident related" category. Experimental results on our labelled dataset show that ULMFiT outperforms other classification models making our model a convenient one for extracting traffic-related information targeting Indian scenario.

***Index Terms*—Road traffic detection, social media analysis, deep learning, transfer learning, keyword extraction**

I. Introduction

Road traffic congestion is a significant issue in India. This problem is induced by incidents or events like road accidents, infrastructure damages, rallies, protests, adverse weather events, disabled vehicles, roadway debris, daily rush hours etc. Detection of these incidents on time or ahead of time wherever possible will help the traffic authorities to alleviate road congestion problem, and also help the commuters as they can pre-plan their trip accordingly.

Historically, researchers have been using live data from different sensors such as loop detectors, GPS probe vehicles, cameras etc. [16] installed on transportation network to detect traffic congestion. But, due to rapid growth of transportation networks the cost of procurement, installation and maintenance of these sensors also increases. In this work, we aim to detect traffic congestion events using Twitter data. Civic authorities as well as general people or groups often publish traffic related data in social media sites like Twitter. There are few such existing datasets [13], [17] that contain traffic related information generated by the human sensors. [13] contains labelled data from US for traffic incident detection. [17] comprises of labelled tweets for incident detection from 10 major English speaking cities. There is no such dataset for developing countries like India. It is anticipated that the nature of the traffic-related tweet data for different countries would be different due to multiple reasons such as writing style, usage of location names, and type of incidents that affect the traffic etc. Furthermore, the information in the existing datasets are mostly concerned about current traffic conditions and contain posts that are extracted after the traffic incident has happened. However, posts that talk about future events (like rallies, protest marches, public gatherings) and their impact on traffic at different regions are also important. These information help people to be well-informed in advance and hence make route or travel planning accordingly. This paper introduces a dataset that we have created for India-specific traffic scenario using a combination of multiple retrieval strategies. The dataset is named as L-TWITS (Labelled-TWeets for Indian Traffic Scenario).

One of the main challenges faced by researchers for incident related classification problem is annotating a huge corpus of data for the target country of interest. In this paper, we demonstrate how transfer learning based methods trained on existing datasets can be used on a target dataset having limited or no label information. We classify tweets in L-TWITS dataset using the knowledge obtained from existing US traffic-related dataset. The main contributions of this paper are:

- We develop a dataset L-TWITS that contains traffic-related tweets for Indian scenario. We make special consideration to capture posts that talk about future events and their impact on traffic.
- We present an architectural framework for identifying traffic congestion events from Twitter data.
- We perform detailed experimental analysis to verify the applicability and usefulness of transfer learning for Twitter-based traffic event detection.

II. Related Works

In this section, we review the use of social media for incident detection purposes, for traffic congestion detection and then discuss on existing text classification problems involving transfer learning approaches.

A. Twitter-based incident detection

A lot of work has been done for Twitter-based incident detection for various purposes. Researchers have used social media data for detecting events in the following topics: drug abuse, flu trend, disaster, first story detection, mass emergencies, sentiment analysis, controversies, traffic incidents etc.

Abel et al. [1] presented Twitcident to filter, search and analyze real-time event information monitored from emergency broadcasting services. On detecting incident, a query is initiated for profiling that incident which is then used to extract messages from Twitter. Semantic enrichment like Named Entity Recognition (NER) and classification is applied to these messages to identify relevant tweets and to provide summarized information related to the incident. Sarah et al. [2] proposed a system that may contribute to situational awareness by analyzing information generated via Twitter during two natural hazard events in North America. Li et al. [3] developed a system to detect Crime and Disaster related Events (CDE) from Twitter. A classifier with 80% accuracy performed CDE detection which is followed by prediction of geo-location of CDE tweets by using a user's historical messages. Schulz et al. [4] introduced automatic classification of tweets related to small scale incidents in real time. They extracted features from Linked Open Data on original tweets and applied spatio-temporal filtering on tweets for filtering irrelevant content before passing it to classifier.

B. Traffic analysis using Twitter

Gu et al. [5] proposed an iterative adaptive data acquisition technique which aims to collect tweets using initial keywords and then create a dictionary of traffic related words. In each iteration they manually labeled the tweets extracted, updated keyword dictionary and crawled tweets using new queries from this updated dictionary. Semi-Naive-Bayes classifier is used to classify tweets into Traffic Incident (TI) or Non Traffic Incident (NTI) class. All TI tweets are then geocoded to determine location entities from tweet text. Das et al. [6] developed a framework that uses SVM model to categorize traffic related tweets and a hybrid georeferencing model which uses supervised methods and spatial rules to detect location entities from tweet content in Greater Mumbai. They collected data using the premium service of DiscoverText and used crowdsourcing service to annotate 3548 tweets. Shekhar et al. [7] performed data analysis to identify congestion cause using traffic related historical data from Twitter. They predicted user's emotion using a sentiment classifier to determine level of congestion and for tweets with negative sentiment, location identification is done and is fed to Google Maps API with instructions to avoid those locations to provide optimized route to commuters. Salas et al. [8] presented a simple approach for real-time detection of incidents in the UK. They manually annotated around 13000 tweets extracted from Twitter API which were then processed, and classified using SVM. Dabiri et al. [9] manually labelled more than 51000 tweet samples for 2-class and 3-class traffic related classification tasks and applied deep learning models.

C. Transfer learning on classification task

Generally, task specific datasets are not easily available. Researches have started applying transfer learning approach to solve various NLP related problems where the available labeled data is small. Since past several years pre-trained word vectors (Word2vec and Glove embeddings) have drawn great interest to improve the performance in various downstream tasks, such as POS tagging, question answering etc. Many pre-trained language models which are trained on very large corpus like ULMFiT, OpenAI GPT and BERT have emerged recently. They enable robust transfer learning for fine-tuning NLP tasks with little labeled data. While working on Social Media Mining for Health Applications (SMM4H) Shared Tasks, Mahata et al. [11] demonstrated the effectiveness of BERT and ULMFiT fine-tuning for heavily skewed classification datasets. Xiao J. [12] experimented with transfer learning using different pre-trained language models and fine-tuned them for contextual emotion detection in SemEval-2019 Task 3. Their experimental results showed that ULMFiT outperforms models trained from scratch due to its superior fine-tuning techniques.

III. Dataset Preparation and Analysis

Although there are tweet datasets containing traffic information for developed countries like USA, UK, Australia etc., there are no such datasets for developing countries. As a part of this work, we try to come up with a dataset: L-TWITS that focuses on Indian traffic context, and contains tweets discussing incidents and scenarios in Indian traffic. One of the contributions of this study is a collection of Indian tweets that have been labelled into two classes- Traffic Incident(TI) and Non-traffic Incident(NTI). Moreover, the existing datasets contain posts about *current* traffic incidents such as vehicle breakdown, high congestion etc. While preparing our dataset, we make special attempt to retrieve posts that talk about future events and their possible impact on traffic. This additional aspect make our dataset different from other existing datasets. We now discuss the data collection mechanism in detail.

A. Data acquisition

The geocode parameters are specified to the boundaries of India. We used three parallel retrieval strategies for identifying the tweets: (a) seed keyword based, (b) from pre-identified twitter handles, and (c) seed event based. Fig. 1 shows the percentages of unique tweets (no retweets) in L-TWITS that were collected using each of these individual retrieval strategies.

1) Traffic related seedwords: An initial set of seed words were created using the US Traffic related dataset [13]. A graph based technique namely TextRank is used in this study to extract keywords from this dataset. The idea of TextRank for ranking keywords is similar to how PageRank is for webpage ranking. Each word in the vocabulary serves as a vertex for graph. If two words co-occur within a pre-defined window size anytime in the data, an undirected edge with weight 1 is added between the corresponding vertices. We have built the graph with window size set to 5.

$$TR(v_i) = (1-\alpha)+\alpha \sum_{v_j \in In(v_i)} \left(\frac{w_{ji}}{\sum_{v_k \in Out(v_j)} w_{jk}} TR(v_j) \right) \quad (1)$$

Here $TR(v_i)$ represents the weight of the word v_i, α is the damping factor, $In(v_i)$ is the set of nodes that point to v_i, $Out(v_j)$ is the set of nodes pointing from v_j, and w_{ij} is the edge weight between node v_i and node v_j. Example seed words extracted using TextRank include *highway, lane, traffic, crash, incident, accident, block etc.* A total of 1263 Tweets were extracted over a period of 7 days using these seed words as a keyword based filter. After labeling we observed that the count of TI tweets is 262 and that of NTI tweets is 1001. It has been observed that seed keyword based techniques generally fetch many irrelevant tweets as the same words may be used in different contexts, all of which might not be related to the context (here context is traffic) under consideration.

2) Twitter handles of traffic departments from major cities: We noticed that the Twitter handles of Traffic Police of a few major cities in India namely Delhi, Kolkata and Hyderabad are highly active on Twitter. Hence we extracted data from these handles. It was observed that many-a-times their tweeting patterns are mostly repetitive. A total of 624 tweets were extracted out of which 454 were traffic related.

3) Using social events as keywords: Many social events like rallies, protests etc. lead to huge traffic congestion problem. A study was done about such kind of recent events in India that led to protests. Those event names were used as keywords to extract tweets using Twitter Streaming API during the happenings of such events. Fig. 2 shows the tweet counts in the dataset extracted for the different events considered.

TABLE I: Sample tweets from L-TWITS dataset

Tweet	**Actual Class**
One broken down vehicle on EM Bypass near Patuli Connector has obstructed the traffic partially.	TI
Traffic Advisory for tomorrow 26th Dec due to CAA agitation. Plan your travel accordingly avoid get caught in @mumbaitraffic	TI
That's the spirit we need on Hyderabad roads. #HYDT-Pwecareforyou #hydcitypolice #HYDTraffic #HYDTP	NTI
Caught by traffic police without helmet	NTI

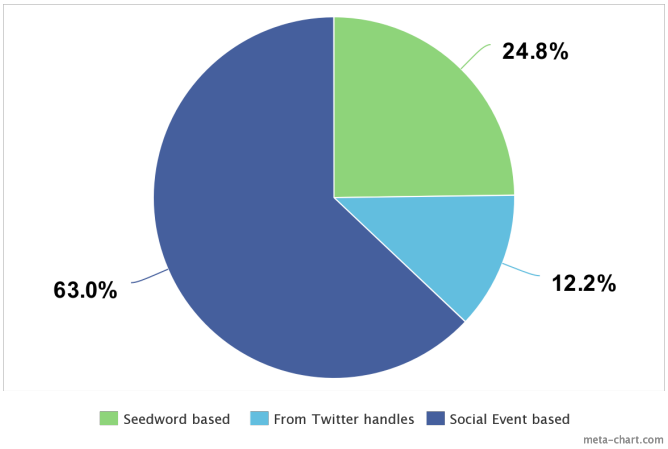

Fig. 1: Percentage of tweets contributed by the individual methods in the L-TWITS dataset

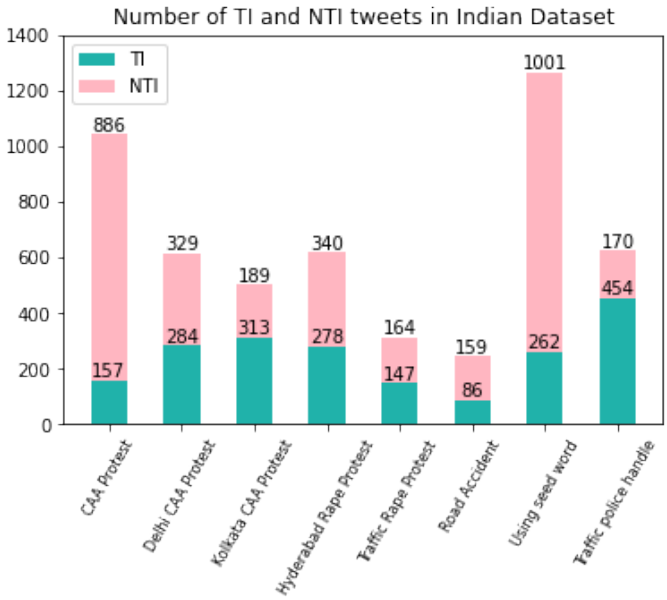

Fig. 2: Tweet counts in the dataset for different events

B. Tweet annotation

The tweets collected using the strategies discussed above are labeled by a set of volunteers into one of the following two classes: (a) Traffic Incident (TI) and (b) Non-Traffic Incident (NTI). Tweets that contain information regarding events that lead to increase in traffic and congestion are labeled with "Traffic Incident (TI)" class. Examples of such traffic related scenarios are rallies, infrastructure damages, accidents, social events, vehicle breakdown etc. Tweets that do not belong to the TI class are marked as "Non-Traffic Incident (NTI)". A sample of tweets along with their labels is shown in Table I. 500 Tweets of the TI class are annotated manually with the incident location information, which are then geocoded into coordinates. Fig. 3 indicates that the collected tweets have wide spatial coverage throughout India. The map is created with Python Folium library.

Fig. 3: Map with markers representing incident location

C. Dataset analysis

We now present some statistical analysis of the dataset: L-TWITS that we have developed. We also compare L-TWITS with the US traffic dataset [13] popularly used for works related to incident detection from tweets. The distribution of tweet lengths in terms of words for the two datasets is shown in Fig 4. Clearly, the distributions differ from each other as 95% of the tweet samples in US dataset have less than 30 words whereas for L-TWITS dataset most of the tweets have word count between 10 to 60. Fig. 5(a) and Fig. 5(b) show top 200 words as word clouds in the US and L-TWITS datasets respectively. Words with bigger font have more weightage (frequency). We can observe that factors affecting Indian traffic are road jam, congestion, protests, accidents, broken down vehicles etc. and for US, words like crash, block, roadworks, construction, disabled vehicles etc. are heavily used, this shows that eminent keywords in both the datasets share similar context since words like vehicle breakdown and disabled vehicles, jam and block are semanticaly similar in traffic context. However, the usage of location mentions like highway, lane, street etc. in US tweets and marg, road, flyover etc. in Indian tweets show the difference in nature of tweets. Prominent keywords in L-TWITS Dataset are shown in Table II.

TABLE II: Top 10 keywords related to traffic in L-TWITS dataset

Keyword	Frequency	Keyword	Frequency
traffic	1370	road	649
accident	598	affected	595
breakdown	482	jam	376
obstructed	249	protesters	236
slow	184	stretch	47

IV. Traffic tweet classification: Methodology

For representing the input tweets we have used both bag of words and word embeddings based techniques. SVM, Naive Bayes, CNN are used as classification models. We then use ULMFiT [14], a transfer learning based model to improve the classification performance. ULMFiT is trained to retain knowledge in the domain of traffic from US tweets and this knowledge is then adapted for the text classification task of L-TWITS tweets. To the best of our knowledge, none of the existing methods have utilized the power of Transfer Learning in the domain of traffic congestion event detection.

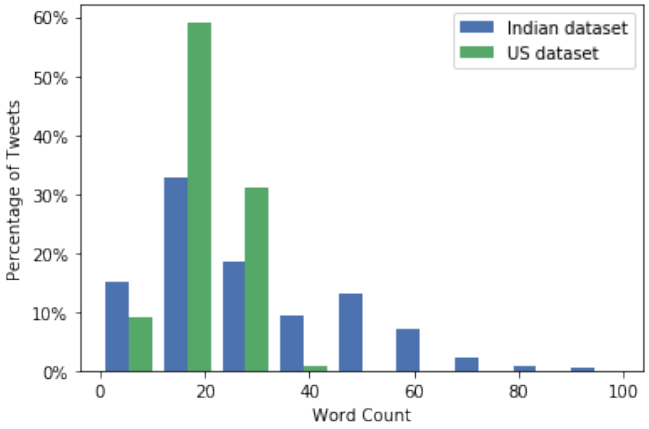

Fig. 4: Word count distribution of tweets

(a) Word-cloud for US dataset

(b) Word-cloud for L-TWITS dataset

Fig. 5: Prominent words from the US and L-TWITS datasets.

A. Input representation

We used two different approaches for representing the tweets for classification: (a) TF-IDF which is a Bag-of-Words based method and (b) Word2Vec which is a word embedding based method.

1) TF-IDF: It is a word-occurrence based representation of tweets. Each tweet is represented as a vector in the vocabulary space. For each word in a tweet, the corresponding TF-IDF value is stored in the vector. The TF value comes from the tweet whereas the IDF value comes from the corpus level statistics and is used to boost the importance of the rare words. The TF-IDF vectors obtained were fed as input to SVM and Naive Bayes models.

2) Word2Vec: Word2Vec is an embedding method to represent the words in a compact vector space based on distributional semantics. It uses a two-layer neural network to learn the representations of the words from large corpus. It is a well-known pre-trained embedding used to reconstruct linguistic context of words. Using Word2vec we map each tweet in the dataset to a fixed length embedding matrix which is given as input to the CNN model in the input layer.

B. Classification methods

We use different classical machine learning based (SVM, Random Forest and Naive Bayes) and deep learning based methods (CNN, LSTM and ULMFiT) for developing the classification models. We briefly describe the CNN. LSTM, and ULMFiT methods below.

1) Convolutional Neural Networks (CNN): Input to CNN is an embedding matrix of tweets where each row is a word vector. This matrix is passed to convolution layer where different sized filters (2-5 words) are applied across the matrix to detect patters in the input. The output of convolution operator is the resulting feature map which is sent to max pooling layer to condense the features extracted into a representative number. This output is then passed to a fully connected layer followed by an activation function on the outputs that gives probability values for each class.

2) Long Short-Term Memory (LSTM): The cell state, various gates and backpropagation through time are the main concepts in LSTMs. Cell state holds significant information during the processing of a sequence and can be thought of as memory of the network. The gates are responsible to process and control cell state. LSTMs are capable to remember relevant information for long periods of time.

3) ULMFiT: Universal Language Model Fine-tuning (ULMFiT) [14] comprises of two parts, the pre-trained encoder language model and the classifier model. The language model is pre-trained on Wikitext-103 general domain corpus containing 28,595 preprocessed Wikipedia articles and 103 million words. The model learns general features of the language in different layers. Overfitting doesn't occur as it doesn't require training a classification model from scratch. Researchers have shown that ULMFiT works very well for smaller datasets.

The implementation requires three distinct steps, that includes (a) Language Model (LM) pre-training, (b) LM fine-tuning to learn task-specific features from target data, and (c) transferring the fine tuned LM to a task-specififc head (here classification task) which is fine-tuned on the target task labelled data.

V. Experimental Results

A. Performance Metrics

The performance of classification models have been evaluated using the following popular metrics: Precision, Recall and F1-Score. We also use Macro-F1-Score that computes the F1 score independently for each class and then takes the average.

B. Datasets Used

The number of labeled tweets in the Indian dataset are less in number. Hence Transfer Learning approach is used to perform classification on this small dataset. The following two datasets have been used in this study.

- US Traffic Dataset [13]: It contains 40879 tweets out of which 20439 are traffic-related tweets (i.e., TI) and 20440 are non traffic-related tweets (i.e., NTI).
- L-TWITS: This dataset is a contribution of this paper to analyze Indian Traffic Scenario using social media. This dataset has 5094 labelled tweets out of which 1981 are traffic related and 3113 are non-traffic related.

C. Classification results

L-TWITS dataset is split into Train set and Test Set. Train Set has 2594 and Test Set has 2500 tweet samples. Table III shows the performance of various classification models which were both trained and tested on L-TWITS dataset. It can be observed that the results of RandomForest and LSTM are almost similar and they performed well with F1-Score of 0.82 in TI class. Other methods couldn't really learn well might be because of the fact that the training data was limited.

TABLE III: Performance comparison of models when trained on L-TWITS train set

Model	Precision	Recall	F1-Score (TI Class)	Macro Avg F1-Score
Naive Bayes	0.78	0.67	0.72	0.80
SVM	0.82	0.72	0.77	0.83
CNN	0.76	0.79	0.77	0.83
RandomForest	0.81	0.83	0.82	0.86
LSTM	0.80	0.85	0.82	0.87

Since the performances of the above methods were not satisfactory, we experimented with ULMFiT model using the transfer learning framework. We created multiple training datasets by augmenting x% of L-TWITS train set with the US data. We experimented with different values of x. Fig. 6 shows that the performance of ULMFiT model on L-TWITS test set increased on increasing samples from L-TWITS train set in Train Set. In Table IV we show performances of CNN, LSTM and ULMFiT trained on a mix of US data and 80% L-TWITS train set (since the performance of ULMFit trained on US data augmented with either 80% L-TWITS train set or 100% set is similar). It can be seen that knowledge transfer using ULMFiT results in significant improvement over other approaches. Due to lack of space, we could not include complete results of all the methods for all training set configurations, and have only reported the best values for the closest competitors, which are CNN and LSTM based

model. It can be observed that by only using 10% of the training data from the target (L-TWITS) dataset along with the US data, ULMFiT is able to outperform the competitor methods.

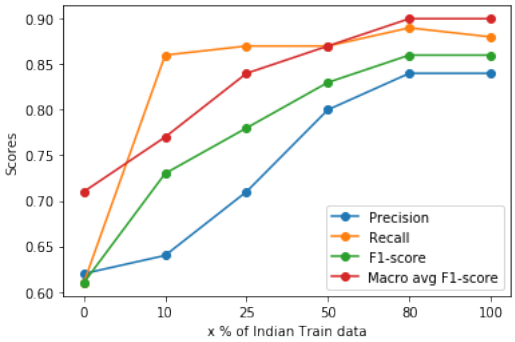

Fig. 6: Performance of ULMFiT on L-TWITS test set. x-axis denotes % of L-TWITS train set combined with the US train set.

TABLE IV: Performance comparison of models when trained on US dataset + 80% of L-TWITS train set

Model	Precision	Recall	F1-Score (TI Class)	Macro Avg F1-Score
CNN	0.77	0.64	0.70	0.78
LSTM	0.79	0.74	0.76	0.82
ULMFiT	**0.84**	**0.89**	**0.86**	**0.90**

TABLE V: Prediction probabilities of tweets misclassified by ULMFiT model trained on US data + 80% of L-TWITS train set

Tweet	Prob (TI)	Prob (NTI)	Actual Class
@CYBTRAFFIC excellent traffic management, just took 40mins to cross hi-tech signal.	0.427	0.573	TI
If Traffic will do their work properly and RTO will do their work properly nearly 90% Road accident will reduce.	0.555	0.445	NTI
Hyderabad university students protest for better security on campus	0.861	0.139	NTI

A validation dataset is created which contains tweets associated with heavy rains, floods and religious events from India. ULMFiT performed with 0.87 F1-Score for TI class in validation dataset which indicates that the model is not biased towards protest like traffic events and performed similarly on random data sample of different events.

However, there are cases where ULMFiT fails to predict the correct class of the test set samples. We include in Table V few tweets for which the model errors in prediction. An interesting observation was that for sarcastic traffic tweets like the first sample in the table, the model gave a higher probability of it being in NTI class. Also, few irrelevant tweets talking about traffic but not related to traffic incident also had higher probability of being classified to TI class.

VI. Conclusion and Future Work

In this paper, we have used Twitter as an information source for detecting traffic related incidents in India. We have created a dataset L-TWITS for this purpose. The tweets were extracted through three efficient strategies using Twitter APIs and were manually labelled into two classes TI and NTI. We then applied a transfer learning approach by using US data and a portion of the labelled L-TWITS train set to fine tune the language model. We could achieve 90% macro-avg F1-score on L-TWITS test set by this approach. Future work will include traffic location detection from tweet content and enhancement of this dataset to increase incident coverage by acquiring more tweets related to various events that may cause traffic like severe rains, floods, religious events etc.

Acknowledgment

This work is a part of the project “M2Smart: Smart Cities for Emerging Countries based on Sensing, Network and Big Data Analysis of Multimodal Regional Transport System”, supported by JST/JICA SATREPS, Japan (Grant Number: JPMJSA1606).

References

[1] Abel, Fabian, et al. ”Twitcident: fighting fire with information from social web streams.” Proceedings of the 21st International Conference on World Wide Web. 2012.

[2] Vieweg, Sarah, et al. ”Microblogging during two natural hazards events: what twitter may contribute to situational awareness.” Proceedings of the SIGCHI conference on human factors in computing systems. 2010.

[3] Li, Rui, et al. ”Tedas: A twitter-based event detection and analysis system.” 2012 IEEE 28th International Conference on Data Engineering. IEEE, 2012.

[4] Schulz, Axel, Petar Ristoski, and Heiko Paulheim. ”I see a car crash: Real-time detection of small scale incidents in microblogs.” Extended semantic web conference. Springer, Berlin, Heidelberg, 2013.

[5] Gu, Yiming, Zhen Sean Qian, and Feng Chen. ”From Twitter to detector: Real-time traffic incident detection using social media data.” Transportation research part C: emerging technologies 67 (2016): 321-342.

[6] Das, R. D., & Purves, R. S. (2019). Exploring the Potential of Twitter to Understand Traffic Events and Their Locations in Greater Mumbai, India. IEEE Transactions on Intelligent Transportation Systems.

[7] Shekhar, Himanshu, Shankar Setty, and Uma Mudenagudi. ”Vehicular traffic analysis from social media data.” 2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI). IEEE, 2016.

[8] Salas, Angelica, Panagiotis Georgakis, and Yannis Petalas. ”Incident detection using data from social media.” 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC). IEEE, 2017.

[9] Dabiri, Sina, and Kevin Heaslip. ”Developing a Twitter-based traffic event detection model using deep learning architectures.” Expert systems with applications 118 (2019): 425-439.

[10] Mikolov, Tomas, et al. ”Efficient estimation of word representations in vector space.” arXiv preprint arXiv:1301.3781 (2013)

[11] Mahata, Debanjan, et al. ”MIDAS@ SMM4H-2019: Identifying Adverse Drug Reactions and Personal Health Experience Mentions from Twitter.” Proceedings of the Fourth Social Media Mining for Health Applications (# SMM4H) Workshop & Shared Task. 2019.

[12] Xiao, Joan. ”Figure Eight at SemEval-2019 Task 3: Ensemble of Transfer Learning Methods for Contextual Emotion Detection.” Proceedings of the 13th International Workshop on Semantic Evaluation. 2019.

[13] Dabiri, Sina (2018), “Tweets with traffic-related labels for developing a Twitter-based traffic information system.”, Mendeley Data, v1 http://dx.doi.org/10.17632/c3xvj5snvv.1

[14] Howard, Jeremy, and Sebastian Ruder. ”Universal language model fine-tuning for text classification.” arXiv preprint arXiv:1801.06146 (2018).

[15] Merity, Stephen, Nitish Shirish Keskar, and Richard Socher. ”Regularizing and optimizing LSTM language models.” arXiv preprint arXiv:1708.02182 (2017).

[16] Zhang, Junping, et al. ”Data-driven intelligent transportation systems: A survey.” IEEE Transactions on Intelligent Transportation Systems 12.4 (2011): 1624-1639.

[17] Schulz, A., Guckelsberger, C.,& Janssen, F. (2017). Semantic Abstraction for generalization of tweet classification: An evaluation of incident-related tweets. Semantic Web, 8(3), 353-372.