Harnessing Social Interactions on Twitter for Smart Transportation Using Machine Learning

Narayan Chaturvedi(✉), Durga Toshniwal, and Manoranjan Parida

Centre for Transportation Systems, Indian Institute of Technology, Roorkee, India
narayanchaturvedi@gmail.com, {durga.toshniwal,m.parida}@ce.iitr.ac.in

Abstract. Twitter is generating a large amount of real-time data in the form of microblogs that has potential knowledge for various applications like traffic incident analysis and urban planning. Social media data represents the unbiased actual insights of citizens' concerns that may be mined in making cities smarter. In this study, a computational framework has been proposed using word embedding and machine learning model to detect traffic incidents using social media data. The study includes the feasibility of using machine learning algorithms with different feature extraction and representation models for the identification of traffic incidents from the Twitter interactions. The comprehensive proposed approach is the combination of following four steps. In the first phase, a dictionary of traffic-related keywords is formed. Secondly, real-time Twitter data has been collected using the dictionary of identified traffic related keywords. In the third step, collected tweets have been pre-processed, and the feature generation model is applied to convert the dataset eligible for a machine learning classifier. Further, a machine learning model is trained and tested to identify the tweets containing traffic incidents. The results of the study show that machine learning models built on top of right feature extraction strategy is very promising to identify the tweets containing traffic incidents from micro-blogs.

Keywords: Machine learning · Twitter data analysis · Traffic incident detection

1 Introduction

Rapid urbanization and an increased number of social media users attract the researchers for harnessing social media data to resolve the problems raised due to rapid urbanization. Recently, the government of developing countries like India has also been focusing on the need for smart cities to resolve the problems of urbanization. Smart cities also demand smart transportation and utilization of continuously generating social media data to improve city transportation. Detection of transport services disruptions, travel complains resolutions, seasonal messages to commuters, and more importantly transport event or incident detection

Published by Springer Nature Switzerland AG 2020
I. Maglogiannis et al. (Eds.): AIAI 2020, IFIP AICT 584, pp. 281–290, 2020.
https://doi.org/10.1007/978-3-030-49186-4_24

are the typical applications of social media data harnessing for the field of transportation. In [17], the authors evaluated the potential of Twitter data for transit customer satisfaction and in [10], the authors proposed an approach for traffic incidents detection using social media data.

People generally express their daily activities using different social media platforms. However, the microblogging platform, like twitter, is most popular among all present microblogging websites. Microblogging service, Twitter has 328 million active users every month [16] who are tweeting at 230000 messages per minute [1]. Twitter is a cost-effective solution for generating a vast amount of real-time data. Relevant information extraction from Twitter data is the primary requirement to characterize the traffic events. In text mining, words of short messages are the tokens which are the basic unit of feature, and these tokens are the key to knowledge discovery from short messages. Since written tweet text cannot be directly utilized in machine learning models for tweet discrimination. Therefore, features of the tweets need to be converted into a numeric sparse matrix representation. The selection of an effective approach for feature extraction will improve the overall accuracy of the traffic event identification task.

The main objective of this study is to present and compare the text feature representation models and state of the art embedding to built a powerful machine learning based framework for the detection of traffic incidents. On the basis of the results acquired, we have proposed a comprehensive approach that works on social media data to extract patterns containing traffic incidents. The main contributions of our work are summarized as following: (1) To collect traffic-related social media data, a dictionary containing prominent traffic related keywords that are popular in social media and day to day communication of citizens, has been created. (2) Keyword filtration and manual approaches are applied to label the collected tweets for the training and testing of machine learning models. The paper proposes a computational framework based on a machine learning model to detect non-recurrent traffic causes.

Rest of the paper is organized as follows: Sect. 2 discusses the work done related to social media and transportation. Section 3 covers the details of data collection and labeling processes, and Sect. 4 explains the steps taken for preprocessing of collected data. Section 5 presents the methodology proposed and Sect. 6 contains the results of the study. Finally, the paper is concluded in Sect. 7.

2 Literature Review

Recently, several studies in the literature have been come out that utilizes social media data in various applications. Studies applied machine learning techniques on top of various non-numeric representation of text features and numeric sparse matrix feature representation to classify social media data containing traffic incidence/traffic-related information and data not related to traffic. A tweet contains many attributes like coordinates, creation time, language, place, timestamp, tweet text, etc. In [4], the authors analyzed not only tweet text but also other attributes of the tweet to demonstrate user activities and their moving

pattern. Along with other attributes, the tweet text has been given more importance in twitter-based studies. The traffic related microblogging messages were retrieved using Support Vector Machine in [3]. The studies in [6,7,14] utilized machine learning models for the detection of tweets containing traffic-related information.

Several studies have been performed to classify the tweets containing traffic information using machine learning (ML) models with numeric feature representation. To know the effect of feature extraction on ML models, there is a need for this study which focuses on a feasibility study of bag-of-words, TF-IDF and word2vec all three on three different machine learning models.

3 Data Collection

Domain-specific data collection is the first challenge in social media data based studies in order to identify relevant tweets. The objective of labeling is to allocate a class identity to every user posted tweet, as related to traffic incident or not. Real-time publicly published tweets are collected using the Twitter streaming API. The streaming API collects and filter tweets based on language, hash-tags, keywords, and geographic bounding box.

In this study, we have collected twitter data based on geographical location and identified general keywords matching specific hashtags such as road, accident, injury, potholes, congestion, jam related to different transportation services for 26 March 2018 to 26 April 2018. The keywords have chosen from the newspaper and research articles and also validated with transportation experts. Our work focuses on tweets related to transportation services with the geographic location of densely populated capital region Delhi of India for our study.

3.1 Dictionary Formation and Labeling

In order to collect traffic incident related tweets using Twitter streaming Api, a dictionary of related keywords has been created. In the first phase of dictionary formation, traffic incident related keywords have been collected manually from the related literature, research papers and news articles. The manually collected keywords have been validated by transport experts. Further, one synonym of every keyword has been fetched from the wordnet dictionary database [12] using python library. Adding synonyms doubles the number of keywords in the dictionary. The keyword dictionary is used to collect Twitter data.

To train a machine learning model, we need to label each tweet with a class name. In this study, we have collected geo-tagged tweets based on traffic related keyword but still, there are tweets collected that do not contain any traffic related information. For example, some keywords like *accident* is a popular keyword for road accident but many times people may use such keywords in other references also. Because of this use of the same keyword in several different references, those tweets that contain the keywords, but still are irrelevant for our study get collected. For example, tweet: *jiocare is taking follows up from last2 years.*

But no improvement in connectivity and congestion at m... is irrelevant to our study but collected due to keyword *congestion*. To train a machine learning model to classify such tweets, tweets are manually labelled in to two classes: i. t - tweets that contain traffic event/incident ii. n - non-traffic tweets. Traffic related and non-traffic tweet classes are abbreviated as t and n, respectively.

The interesting factor of this study is the data collection strategy in which a dictionary of traffic-related keywords is formed and used to collect geo-tagged tweets to train the classification model. This approach of data collection represents that collected Twitter data-set contains both types of tweets: traffic-related and non-traffic related but most of the non-traffic tweets have different reference/domains containing some of the similar keywords of other class.

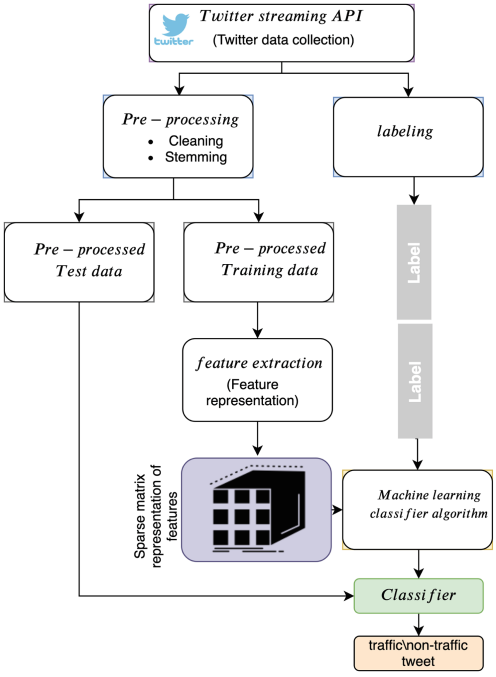

Fig. 1. Detailed steps of the proposed methodology.

4 Preprocessing

The main objective of the pre-processing is to make the tweets eligible for the classification task. Special characters like punctuation and stop words are frequently used in tweets and these stop words do not have any meaning. Therefore, the removal of such special characters and hyperlinks are the primary step of pre-processing. We have removed the # symbol, "@" symbol from the tweets. Tweets have been split into keywords (tokens) using Natural language Toolkit

library. This process is known as tokenization of tweets. Further, upper case keywords could not be interpreted same so all keywords have been changed to lowercase. We performed Stemming to reduce the words to their stem. Thus, we have been changed each tweet into a bag of tokens eligible for text mining task.

5 Methodology

K Nearest Neighbour, Naive Bayes and Support Vector Machine are popular machine learning techniques that have been applied by researchers in text mining task [3,6,8,14]. The comprehensive methodology for detecting tweets consist of traffic events have shown in Fig. 1 with all the steps taken, starting from tweets crawling to built classifier. The Bag-of-words, TF-IDF and Word2Vec embedding have been applied in conjunction with machine learning models to construct an efficient framework for traffic incident detection.

5.1 Feature Extraction Model

The section gives the brief idea about the feature extraction model and word embedding applied in the study.

Bag-of-words (BOW) [2] is a simple technique to represent written text document with machine learning algorithms. BOW keeps the frequency of terms present in a text document and this term frequency is used to represent text in numeric form. We have used CountVectorizer from sklearn library to count the occurrences of words and further represent them in form of sparse matrix for machine learning technique implementation.

Term frequency-inverse document frequency (TF-IDF) is a statistical measuring technique to calculate the weight which decides the significance of the word in the document. Term frequency of a word w in a document of tweets T_n, $TF(w, T_n)$ can be calculated as per Eq. 1. IDF score of a word indicates the rareness of word in a written text document. IDF score of a word w which is present in T number of tweets out of total T_n collected tweets can be calculated as per Eq. 2. Further, TF-IDF weight of a word w is calculated to know the final normalized significance value as per Eq. 3.

$$TF(w, T_n) = \frac{count(w)}{word\_count(T_n)} \tag{1}$$

where count(w) is the frequency of word w in text document and $word\_count(T_n)$ is the total word count in text document.

$$IDF(w, T_n) = log_e \frac{T}{T_n} \tag{2}$$

$$TF - IDF(w, T_n) = TF(w, T_n) * IDF(w, T_n) \tag{3}$$

Word2Vec (W2V) is neural network-based word embedding tool in natural language processing to create a n-dimensional vector corresponding to every word of a text sentence. Continuous bag of words (CBOW) and skip-gram models are the two main approaches to train W2V model [11]. In the process of W2V model training, a vocabulary from tokenized training data creates in first phase. In second phase, similar text features are grouped based on cosine similarity distance. In this study, we have created separate W2V word vectors from the training and test dataset to train and set the machine learning classifier. To implement the feature extraction model and generate the numeric feature vector, gensim [15] python library has been used.

5.2 Machine Learning Model

The section covers the brief idea about the machine learning classifiers used in the study.

K-nearest neighbors is a non-parametric machine learning classifier which scores its nearest neighbors in training data and k-top-scored neighbors' class is used to classify the new input data [5,9]. The decision rule can be written as Eq. 4.

$$Score(t, c) = \sum_{t_n \epsilon KNN(t)} similarity(t, t_n) D(t_n, c) \tag{4}$$

$D(t_n, c)$ is the categorization for tweet t_n with respect to category c that is defined as Eq. 5.

$$D(t_n, c) = \begin{cases} 1, & \text{if } t_n \in c \\ 0, & \text{if } t_n \notin c \end{cases} \tag{5}$$

Naive Bayes (NB) classifier is based on an assumption that probability of one feature to be in a class is independent to the presence of any other feature in the class. Naive Bayes model is based on Bayes' theorem and performs well on big datasets. It is used in text classification [13] that uses a method of estimating the possibility of different classes based on different features of a written text document. MultinomialNB has been used to implement the classifier with BOW and TF-IDF model while the Gaussian distribution NB has been applied with W2V vectors because W2V vectors contain negative values and Gaussian NB works well with such values.

Support Vector Machine (SVM) is the supervised classification technique which categorizes the data. The basic SVM model training is performed as follows: 1. Plot features in a dimensional space where each plotted point co-ordinates are known as support vectors. 2. A separating line needs to be found in such a way so that each point should be farthest from the separating line. 3. In the testing phase, depending on the test, where the data is found on either side of the line, we can categorize new data.

6 Results and Analysis

This section evaluates the performance of machine learning classifiers with the combinations of feature representation models. The machine learning model training and the confusion matrix are also investigated.

6.1 Performance Metrics

In this study, mainly two popular classification metrics have been used to measure the performance of the classifier.

- Accuracy - the percentage/fraction of rightly classified twitter messages is known as the accuracy of trained model classifier.

$$Accuracy = \frac{TN + TP}{TN + TP + FN + FP} \tag{6}$$

  Where TN is true negative, TP is true positive, FN is false positive and FP is false positive.
- Precision and Recall - Precision is the percentage/fraction of relevant tweets while recall is the percentage of actual positives rightly identified.

$$Precision = \frac{TP}{TP + FP} \tag{7}$$

$$Recall = \frac{TP}{TP + FN} \tag{8}$$

- F-measure - harmonic average of precision and recall. F-measure is the function of precision P and recall R and can be defined as in Eq. 9.

$$F - measure = \left(\frac{P^{-1} + R^{-1}}{2}\right)^{-1} \tag{9}$$

6.2 Classifier Performance

Table 1 shows the accuracy of various combinations of machine learning models with Bag-of-words, TF-IDF and W2V feature extraction models to categorize tweets containing traffic incidents. To find the value of number of neighbors K for the KNN classifier, the variation between the number of neighbors and misclassification error has been calculated. The value of K at which minimum value of misclassification error has been obtained is assumed to be the optimal value of K. The variation of Misclassification error and K has been shown in Fig. 2 for W2V. The optimal value of K = 3 in case of BOW, K = 5 in case of TF-IDF and K = 9 in case of W2V has been used to train the KNN classifier. The accuracy results in Table 1 clearly depicts the competative results of all three machine learning models. However, SVM with the combination of W2V

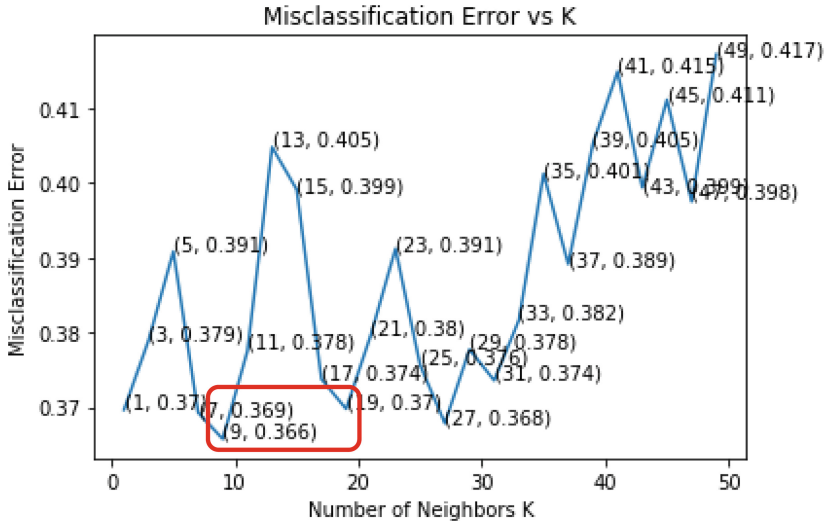

Fig. 2. optimal value of K in KNN classifier with W2V.

embedding technique outperforms other combinations of feature extraction and machine learning models by almost 3% and It successfully brings about the Precision, recall and F-measure of 77%, 84% and 80% for the class of tweets containing traffic incidents, respectively.

Since the SVM model with the W2V has achieved the best accuracy, this combination is treated as the representative of traffic incident detection approach from social interactions.

Table 1. Accuracy of the machine learning models (ML models), trained and tested on top of bag-of-words, TF-IDF and W2V techniques

ML model	Accuracy (%)		
	Bag-of-Words	TF-IDF	Word2Vec
KNN Classifier	62	64	73
Naive Bayes	61	64	71
Support vector machine	69	74	76

6.3 Real World Case Assessment and Confusion Matrix

In this study, we have collected geo-tagged twitter data for the capital region Delhi of India to evaluate the performance of machine learning based incident detection framework. However, our assessment has not been taken the case of real world in to consideration. The final objective of this study is to identify traffic incidents from the citizen's posted messages on micro-blogging platform

Table 2. Classification report(Y_{test}, *prediction*)

	Precision	Recall	F1-score
n	.75	.68	.71
t	.77	.84	.80
Micro avg.	.76	.76	.76
Macro avg.	.76	.76	.76
Weighted avg.	.76	.76	.76

like twitter. In order to detect traffic incidents from twitter streams in real time, the classifiers need to trained on labeled historical tweets and then future tweets can be analyzed for identifying traffic events. Since real time evaluation is a need for such studies, 70% of labeled tweets have been selected to train the model while remaining labeled tweets worked as test dataset. The classifier's performance is examined on the test data-set to conclude the optimal comprehensive approach.

Confusion matrix is the best way to analyse the predictions of both classes of tweets. Table 2 represents the precision, recall, F1-score, and confusion matrix for the tweets test set. Further, the effectiveness of the classifying model is directly proportional to precision, recall, and F1-score for both the classes. Some studies reported the lower precision and recall for traffic class tweets. The reason behind this lower value may be a lower number of traffic class tweets. However, in our case, we have different data collection strategy as described in Sect. 3 due to that we have collected a high number of traffic class tweets t in comparison to non-traffic class tweets number. Therefore, as depicted in Table 2, the performance metrics for non-traffic class n is obtained lower.

7 Conclusion

The study proposes a computational framework that includes a machine learning model built with word embedding technique to detect tweets that contain information related to traffic incidents. The method uses a text feature extraction and its transformation into numeric sparse matrix representation to implement text features on top of machine learning classifiers. Further, a dictionary of traffic-related keywords has been formed which is used to collect the tweets containing traffic incidents. Collected tweets are analyzed, pre-processed, and labeled to model a classifier for detecting tweets containing traffic incidents.

The results of the study present that neural network-based W2V shows slightly better results than TF-IDF and BOW for traffic incident detection. However, as one of the limitations of the W2V model, training of the model is much more complex which requires more processing and memory resource in comparison to TF-IDF model. A tradeoff occurs in choosing TF-IDF and W2V model in different applications. The study clearly identifies that the combination of W2V model with SVM classifier gives a best fit computational framework to detect traffic incidents from social media data.

References

1. Ashtari, O.: The super tweets of# sb47. Twitter. com Blog (2013)
2. Berry, M.W., Castellanos, M.: Survey of text mining. Comput. Rev. **45**(9), 548 (2004). https://doi.org/10.1007/978-1-4757-4305
3. de Carvalho, S.F.L., et al.: Real-time sensing of traffic information in twitter messages (2010)
4. Chaniotakis, E., Antoniou, C.: Use of geotagged social media in urban settings: Empirical evidence on its potential from twitter. In: 2015 IEEE 18th International Conference on Intelligent Transportation Systems, pp. 214–219. IEEE (2015)
5. Cover, T.M., Hart, P.E., et al.: Nearest neighbor pattern classification. IEEE Trans. Inf. Theory **13**(1), 21–27 (1967)
6. D'Andrea, E., Ducange, P., Lazzerini, B., Marcelloni, F.: Real-time detection of traffic from twitter stream analysis. IEEE Trans. Intell. Transp. Syst. **16**(4), 2269–2283 (2015)
7. Fu, K., Nune, R., Tao, J.X.: Social media data analysis for traffic incident detection and management. Technical report (2015)
8. Gu, Y., Qian, Z.S., Chen, F.: From twitter to detector: real-time traffic incident detection using social media data. Transp. Res. Part C Emerg. Technol. **67**, 321–342 (2016)
9. Han, X., Liu, J., Shen, Z., Miao, C.: An optimized k-nearest neighbor algorithm for large scale hierarchical text classification. In: Joint ECML/PKDD PASCAL Workshop on Large-Scale Hierarchical Classification, pp. 2–12 (2011)
10. Mai, E., Hranac, R.: Twitter interactions as a data source for transportation incidents. Technical report (2013)
11. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in neural information processing systems, pp. 3111–3119 (2013)
12. Miller, G.A., Beckwith, R., Fellbaum, C., Gross, D., Miller, K.J.: Introduction to wordnet: An on-line lexical database. Int. J. Lexicogr. **3**(4), 235–244 (1990)
13. Mitchell, T.: Machine learning. mccraw hill, 1996. 93 d. moniere et d. labbé. essai de stylistique quantitative. In: JADT, pp. 561–569 (2002)
14. Pereira, J., Pasquali, A., Saleiro, P., Rossetti, R.: Transportation in social media: an automatic classifier for travel-related tweets. In: Oliveira, E., Gama, J., Vale, Z., Lopes Cardoso, H. (eds.) EPIA 2017. LNCS (LNAI), vol. 10423, pp. 355–366. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-65340-2_30
15. Rehurek, R., Sojka, P.: Software framework for topic modelling with large corpora. In Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks. Citeseer (2010)
16. Sadam, R.: Twitter reports 6 pct increase in monthly active users. https://www.reuters.com/article/twitter-results/twitter-reports-6-pct-increase-in-monthly-active-users-idUSL4N1HY48L. Accessed 23 Jun 2019
17. Wu, B., Idris, A.O.: Measuring and visualizing transit customers' satisfaction using twitter data. Technical report (2018)