Real-Time Detection and Visualization of Traffic Conditions by Mining Twitter Data

Sonia Khetarpaul(✉), Dolly Sharma, Jackson I. Jose, and Mohith Saragur

Department of Computer Science and Engineering, Shiv Nadar University, NCR, Greater Noida, India
{sonia.khetarpaul,dolly.sharma,jj779,ms207}@snu.edu.in

Abstract. There have been various attempts to leverage the massive amount of data generated from social media websites. The real-time nature of social media platforms can help detect events, especially in a metropolitan city. In this paper, a system is proposed, that detects traffic-related events and road conditions in real-time from tweets by using classification algorithms and custom-trained named entity recognition model (NER) to classify and extract contextual information and visualise it on a map to get an overall picture of the traffic conditions in a city. The proposed system is versatile and can be applied to other use cases such as detecting calamities, social unrest, etc.

Keywords: Social media mining · Text mining · Traffic analysis · Classification · Named entity recognition · Data visualization

1 Introduction

The ever-growing popularity of social networking websites like Facebook, Instagram, Twitter, etc. causes more and more people to use these networks to connect with others and as a source of information. These websites have become a very rich source of unstructured data. The users of these websites share real-time information on a variety of topics. These websites allow users to find more people with similar interests and help in the easy flow of information among their social groups/communities. Users can also share their personal or public experiences, express different ideas and thoughts, report different problems and events using these websites in real-time.

Twitter has more than 330 million active users and produces more than 500 million tweets each day. This makes it one of the largest social networking portals in the world, generating more than 12 terabytes of data every day. The sheer volume of data produced every day, if mined properly could prove to be a very constructive source of information. Twitter's question of "What's happening?" captures the primary objective of the portal very well, i.e., to share what's happening around you. In this paper, data generated by Twitter is used to detect

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2022
W. Hua et al. (Eds.): ADC 2022, LNCS 13459, pp. 141–152, 2022.
https://doi.org/10.1007/978-3-031-15512-3_11

events in real-time with a greater level of precision as compared to other studies. The study is limited to events related to traffic, such as congestion, accidents, diversions, etc.

Since tweets are highly unstructured, various text mining techniques like preprocessing, classification, named entity recognition are applied to transform it into structured information which can be used for further analysis. The main objective of this paper is to create a system, which consumes unstructured data to detect real-time events and produce visualisations on the city map, thus making the model usable in other domains as well.

1.1 Motivation

In most metropolitan cities, traffic is a serious issue. Many people travel to work using the same route every day and the traffic can get very severe during peak hours. Events like road blockage, diversions due to accidents, construction work, etc. adds on to the traffic. Detecting such events as they occur and notifying commuters can help save time, fuel and manpower. Google Maps detects real-time traffic but it does so based on the data it receives from mobile devices: the number of devices in a particular area and their speed is used to determine the density of the traffic. Suppose a road is obstructed and the amount of traffic in the road is also sparse or certain routes are blocked by the police on a particular day, Google Maps may fail to detect and provide an alternate route.

The proposed system will be useful in detecting traffic events and can be extrapolated to detect areas of stress in events such as civil unrest, calamities, etc.

1.2 Contributions

The following are the main contributions of the paper:

- After preprocessing, the tweets are classified based on whether they contains traffic related information with very high accuracy.
- Custom trained named entity recognition (NER) model to extract information such as event type, location, reason and advice is used.
- The extracted information is further processed and is converted into coordinates, which is then consumed by an application that visualizes the information.

2 Literature Review

Twitter activity increases during events of stress such as natural disasters, riots, wars, etc. In [3], the authors have described people as a social sensor. Their work provides the base for event detection on Twitter. They formed a system that detects and reports earthquakes by monitoring Twitter. This allowed them to detect an earthquake and then report it much quicker than traditional broadcast announcements in Japan.

Aya Ishino et al. in [4] proposed methods to retrieve transportation information and traffic problems from tweets written in Japanese and posted during a disaster. The data sets used by them were quite small, which reflected in the performance of the models they trained. They were able to obtain a precision of 77.7% and recall of 70.7% in identifying tweets with traffic-related content.

In [5], the authors analyzed tweets related to traffic in the city of Jeddah and found the most congested roads and time of the day. They monitored Tweets related to traffic in the city of Jeddah and using temporal information present in the tweets they identified which roads are generally congested at different hours of the day. In [1], authors classified traffic related tweets into point or link where link information is the traffic information that has all the attributes: Road, Start point and End point, and point information is the traffic information that has Road and Start or Stop point attribute or only Road or Start point attribute or Stop attribute.

In [6], the authors created a system that performs event classification and location prediction from tweets during disasters. They used Markov model to predict the location of the tweet when geolocation data was not present. They achieved a classification accuracy of 81% and predicted location with an accuracy of 87%. In [7] the authors propose a method for automatic detection of road traffic-related events. They also used big data technologies like Spark to perform analytics. Their system used multiple classifiers to detect various kinds of events like accidents, roadwork, road closure, road damage, etc. Their study is unique in the sense that they use big data technologies for event detection of road traffic events from tweets in Arabic.

In [15] the authors have developed a big data tool over Apache Spark for traffic-related event detection from Twitter data in Saudi Arabia. They have used 3 Machine Learning algorithms to build multiple classifiers that can detect 8 different types of events. In [14] the authors propose a technique, "Transaction-based Rule Change Mining" to extract hashtag keywords present in tweets. This provides a rule dynamics approach to event detection, they demonstrated its application in the domain of sports and politics to detect events.

Authors in [16] have used the text data from social multimedia websites like YouTube and Flickr (title, description, tags, time, location etc.), they have devised a clustering-based approach to detect sub-events or smaller events during a crisis. In [2], the authors broadly categorize events into Large-Scale events (e.g. - earthquakes, tornadoes etc.) and Small-Scale events (e.g. - traffic, accidents etc.). Detecting small scale events is difficult since there is a small number of tweets about them and they belong to a small geographic location. On the other hand, Large-Scale events have a large number of tweets and belong to a wider location. They performed classification to detect events belonging to both these categories.

3 Proposed Approach

In this section, first the high-level view of the proposed system is discussed followed by each component. It is important to note that all of the following

components work on run time, i.e., dynamically, as our objective is to detect events as they happen and not at a later point in time.

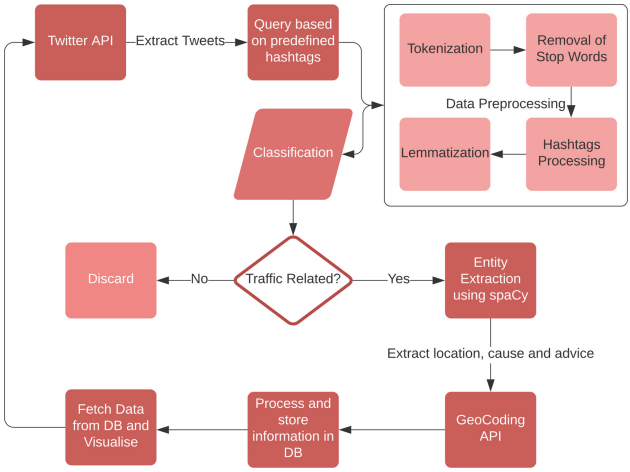

Fig. 1. The architecture of the proposed system

3.1 The Proposed Architecture

The architecture of the proposed model is shown in Fig. 1. The model starts with a Twitter streaming API, whenever a tweet satisfying certain conditions is found, it enters our pipeline. The first step in the process is to clean the tweets because the tweets are unstructured texts, they may contain abbreviations, misspelt words and sometimes grammatical errors. Twitter users commonly use trending slang, emoticons, shorthand, etc. If the tweets are not pre-processed, it would hamper the accuracy levels of our models. After this, the cleaned tweets are classified based on whether the tweet contains information related to traffic or not. If it is found to be a traffic-related, the tweet is sent to the next stage of the model. The next step is using a named entity recognition (NER) model to extract information such as event type, location, reason and advice. A table containing this information is created. The location information is then converted into coordinates and sent to a cloud database, which then gets consumed by an application that visualizes the retrieved information.

3.2 Data Collection

Two different data sets are used to build the models. The first one is obtained by crawling Twitter's publicly available API to query tweets. Tweets tweeted by the Delhi Traffic Police and certain other handles which regularly post updates related to traffic were queried, and then manually labelled to 0 or 1 depending

on whether it had traffic related information or not. The second data set is from Mendeley Data [22]. This data set has tweets labelled in three classes, 0, 1 and 2. The label 0 represents tweets that are not related to traffic. Label 1 represents tweets that report non-recurring events which generate an abnormal increase in traffic demand or reduces transportation infrastructure's capacity. Label 2 represents the type of tweets that report traffic flow conditions such as daily rush hours, traffic congestion, traffic delays due to high traffic volume or jammed traffic. Classes 1 and 2 are combined since both of them contain traffic-related information. Both of these data sets are used in different combinations in different parts of the process.

The reason two data sets are used is to capture both types of tweets - those from organizations such as Delhi Traffic Police, whose tweets are structured and also those which are tweeted by individual users.

An example to demonstrate their difference:
Official Delhi Police Tweet:
"Traffic Alert Obstruction in traffic in the carriageway from Nangloi chowk towards Peera Garhi due to breakdown of a Cluster bus near Nangloi flyover. Kindly avoid the stretch".
User Tweet:
"Disabled vehicle, right lane blocked in #VanAlystneGraysonCo.Line on NB at County Line Rd, stop and go traffic back to FM- #DFWTraffic".

As evident, the tweet by Delhi Traffic Police is both formal and well structured, hence using both these data sets are necessary to train the models to capture information from both types of tweets. The data set included roughly 100k tweets. It was split in 80:20 ratio for training and testing sets. There was no class imbalance, both the classes were approximately equal in size, both in training and testing set.

3.3 Data Preprocessing

This is a very crucial step in the pipeline as the quality of the data plays an important role in improving the overall performance of machine learning models. Precision and recall are two important metrics for the evaluation of any model. If the recall becomes low the model would miss certain information which might have been useful, or if the precision becomes low the model might include irrelevant data points.

Once a tweet enters the system, the following steps are taken:

1. All user mentions are removed. User mention begins with '@', hence is easy to identify. They do not provide any useful information in this use case.
2. A common trend in most social networks is that people embed information in hashtags. For example consider these tweets "Accident cleared in #OxonHill on Beltway Local Lanes Outer Loop on Woodrow Wilson Memorial Brg,

jammed back to highway, delay of mins" and "Disabled Train Causes Rush-Hour NJ Transit Delays #NewYorkCity", both of these tweets have location information embedded in hashtags, which should not be disregarded. Hence while preprocessing hashtags are converted to proper words, "#OxonHill" is replaced by "Oxon Hill" and "#NewYorkCity" is replaced by "New York City".
3. All hyperlinks are removed from tweets as they also do not provide much useful information.
4. All stop words are removed and each word is lemmatized.
5. The tweet is deleted if its length becomes less than 3 tokens (words).

3.4 Classification of Tweets

Once a tweet enters into this part of the pipeline it gets classified into two classes depending on whether it contains information related to traffic or not. Existing machine learning models like Logistic Regression [8], Random Forest Classifier [9], Support Vector Machines [10] and Naive Bayes Classifier [11] are employed on our data set and their performance is compared.

Class 1 represents tweets with traffic-related information, and class 0 represents tweets without traffic-related information. The performance of each model employed is described and shown in Table 1, and comparison of **mean square error** is shown in Table 2.

Table 1. Performance comparison of different classifiers on the 2-class datasets

Classifier	Class	Precision	Recall	F1-score	Support
Logistic Regression	1	0.98	0.99	0.99	5239
	0	0.99	0.98	0.99	5250
Naive Bayes	1	0.98	0.98	0.98	5239
	0	0.98	0.98	0.98	5250
Support Vector Machine	1	0.99	0.99	0.99	5239
	0	0.99	0.98	0.99	5250
Random Forest Classifier	1	1	0.99	0.99	5239
	0	0.99	1	0.99	5250

The most notable performance was of Support Vector Machine and Random Forest Classifier with the mean squared error of just **0.0084** and **0.0056** respectively. Random Forest Classifier has low bias and moderate variance, this is due to each tree in the forest having low bias, and as the variance of each tree is averaged out, the overall variance turns out to be moderate. These characteristics, along with high accuracy scores made it the ideal choice. The tweets which get labelled as Class 1 are sent forward into the pipeline, the rest are discarded.

Table 2. Comparison of mean squared errors

Classifier	Mean squared error
Logistic Regression	0.0135
Naive Bayes	0.0226
Support Vector Machine	0.0084
Random Forest Classifier	0.0056

3.5 Named Entity Recognition

Named entity recognition (NER) [13] is a process where a sentence is parsed to find entities that can be categorized into names, locations, quantities, monetary values, places, etc. In the system, NER is being used to detect location, event, cause of the event, time and suggestions in a particular tweet. "spaCy" [12], an open-source software library with built in NER models was used. Users generally do not mention the time/date of the incident which they are tweeting about, but that is not a problem, as information related to time can be extracted from the tweet itself.

To create the data set for the NER, 100 dissimilar tweets which contained information related to traffic were identified. Then parts of the text which belonged to classes such as event type, location, reason and advice were marked in each tweet, and then the data set was supplied to train a blank NER model using spaCy.

Results of NER Extraction. The tweets which enter this part of the pipeline are cleaned and most probably (99% accuracy) contain traffic information. Some examples can be seen below. Examples of the output of the NER pipeline is shown in Table 3. The text shown is after preprocessing.

1. "Traffic is affected on Signature Bridge due to demonstration near Bhajanpura. Motorists are advised to use NH-24, Geeta Colony Flyover & NH-1 to cross Yamuna River."
2. "Traffic Alert Traffic be affect near Indraprastha Gas, Rohini due to water logging."
3. "Traffic Alert Traffic be slow near Kabutar Market, Please avoid the stretch"
4. "Traffic Alert Obstruction in traffic in the carriageway from Nangloi chowk towards Peera Garhi due to breakdown of a Cluster bus near Nangloi flyover. Kindly avoid the stretch."

The NER extracts these entities from each tweet in this pipeline which is then passed onto the next stage, Geocoding.

The evaluation of the custom trained NER is different from the way traditional models are measured. Entity level and token level evaluation are two different methods to measure a NER model. Entity level evaluation considers

Table 3. Output of NER

Tweet	Event/Reason	Locations	Advice
1	Demonstration	Signature Bridge Bhajanpura NH-24, Geeta Colony Yamuma River	Use
2	Water logging	Indraprastha Gas, Rohini	
3		Kabutar Market	Avoid
4	Breakdown of cluster bus	Nangloi Chowk Peera Garhi Nanngloi Chowk	Avoid

Table 4. Performance of named entity recognition

Class	Precision	Recall	F1-score	Support
Location	0.80	0.85	0.83	39
Reason	0.73	0.58	0.65	19
Event	0.84	0.84	0.84	19
Advice	0.64	0.60	0.62	15
Other	0.84	0.84	0.84	87
Macro avg	0.81	0.79	0.80	179
Macro avg	0.80	0.79	0.80	179

Overall Accuracy : 88.42%

only the correctly labelled named entities. For example "Shaheen Bagh" is a location and it is considered as correctly labelled if the entire word is being marked as "LOCATION" (as one named entity). Token level evaluation checks the label of each token. For example "Shaheen Bagh" is treated as two tokens, and if both the tokens get marked as "LOCATION" it is treated as correctly labelled, but if one gets marked as "LOCATION" and the other does not, it is considered as a partial match. In this paper, entity level evaluation is used to evaluate the NER model as partial matches are not beneficial. Based on the number of correctly recognised named entities the accuracy score is calculated in Table 4.

3.6 GeoCoding

This part of the pipeline processes the location data, into a more manageable format, i.e., coordinates. The NER model identifies all the different locations in a tweet which needs to be converted into coordinates. Forward Geocoding is the process of taking input text, such as an address or the name of a place and returning a latitude/longitude. Forward GeoCoding is used to obtain the

latitude and longitude of the locations found by the NER. Google's GeoCoding API is used for this.

3.7 Data Storage and Traffic Events Visualization

Firebase is popular backend support for modern applications, it can host websites, provide real-time databases, safe and secure authentication for mobile and web apps. Cloud Firestore, a flexible NoSql database is used to store the details of the events, like coordinates for all locations, the reason behind the event and advice for the public. This information is fetched by an application which then visualises it.

The application constantly scans the database for all the information posted and plots them on Google Maps. Single points of focus are depicted using a heat map surrounding that point. When a road or route is identified, it is then highlighted with red or green depending on whether to avoid the particular route or use it instead of some other route.

4 Results

The proposed system can detect, analyse extract information accurately, and visualize the information from tweets on a map. The information extracted could be useful to both traffic police operators who would be aided with the ease of use of this application which identifies stress locations and to travellers who can take alternate routes when their usual path is obstructed. Figure 2 contains plots made based on tweet 1 from Sect. 3.5, the NER identified four different locations, and corresponding to the second pair of locations, i.e., NH-24, "Geeta Colony" and "Yamuna River" found the "Advice" to be "use". Hence the second pair of locations has its route marked in the colour green, signifying an alternate route to cross the river.

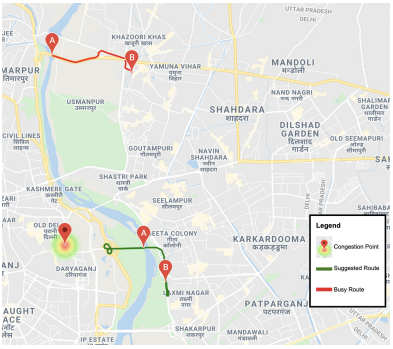

Fig. 2. Real-time visualization of advised routes

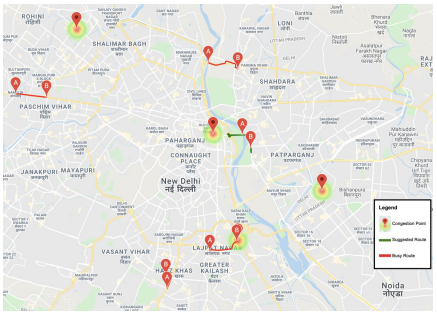

Fig. 3. Real-time visualization at a given point of time

When a tweet highlights a traffic problem in some specific location and not a route, a heat map is generated around that location. Figure 3 shows an overall view of all the information at a given point of time.

4.1 Comparison with Related Studies

In [2], the authors created a system that fetches tweets based on various search criteria, processes them by applying text mining techniques, and then performs classification on it. They solve a multi-class classification problem to determine what kind of event occurred. They obtained an accuracy value of 88.89% for the multi-class problem of detecting different kinds of events and 95.75% for the binary classification problem (traffic versus non-traffic tweet).

The system created has significant differences when compared with [2,19–21]:

1. A higher accuracy is obtained in the binary classification problem (traffic versus non-traffic tweet). This improves the overall accuracy of the proposed system.
2. NER extracts exact contextual information from a tweet. The accuracy of the NER was 88.42%, which is comparable to the accuracy obtained in [2] (88.89%), which uses several different classification models to extract entities.
3. The proposed approach visualises and captures events that lead to traffic congestion, such as protests, social unrest in certain parts of the city or any other form of large scale event that can affect the traffic.

5 Future Work

The proposed end-to-end model is one of its kind. The system shows promising results which can be further extended into other domains. The following ideas can be considered in the future:

1. The NER also needs to understand the local/regional language to be able to detect locations better.
2. Adding real-time trusted information to maps can help people on the road avoid blocked/closed routes and unnecessary re-routing on the way. This feature can enhance the experience of Google Maps.
3. This service can be extended to user tweets too, so that people can update about their locality but this can make the system prone to a wide variety of adversarial attacks which has to be addressed [18].
4. Extrapolate the system to detect other types of events such as civil unrest, calamities, etc. which could especially be useful in identifying areas of stress.

6 Discussion and Conclusion

The ability to identify/detect traffic events in a metropolitan city can be of great use for the daily commuters and the authorities, saving time, fuel and

manpower. Through this paper, we have used social platforms to detect traffic conditions and visualise them. This use case is not just limited to traffic events in the city but can also be used to identify others mass events that might need urgent attention from the authorities. This paper stands out in a number of ways compared to previous work in this area as mentioned in the above section. In the initial classification task (traffic vs non-traffic) random forest classifier was the most effect model. Next NER was able to detect 4 different entity classes with an accuracy of 88.42%. The visualisation was able to intuitively summarise all the extracted information on the map. The model can be used to help authorities better manage traffic by visualising all major traffic events on the map. By incorporating real-time information into maps, it can help users save time and fuel by avoiding congested or blocked routes.

In this paper, we demonstrated incorporating information extracted from Twitter data into internet services. The application of these techniques is countless, this study aimed to demonstrate one such use case which utilises the massive pool of data generated each day by social media.

References

1. Wanichayapong, N., Pruthipunyaskul, W., Pattara-Atikom, W., Chaovalit, P.: Social-based traffic information extraction and classification. In 2011 11th International Conference on ITS Telecommunications, pp. 107–112. IEEE, August 2011
2. 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)
3. Sakaki, T., Okazaki, M., Matsuo, Y.: Tweet analysis for real-time event detection and earthquake reporting system development. IEEE Trans. Knowl. Data Eng. **25**(4), 919–931 (2012)
4. Ishino, A., Odawara, S., Nanba, H., Takezawa, T.: Extracting transportation information and traffic problems from tweets during a disaster. In: Proceedings of the Institute of Mathematics and Mechanic, pp. 91–96 (2012)
5. Alomari, E., Mehmood, R.: Analysis of tweets in Arabic language for detection of road traffic conditions. In: Mehmood, R., Bhaduri, B., Katib, I., Chlamtac, I. (eds.) SCITA 2017. LNICST, vol. 224, pp. 98–110. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-94180-6_12
6. Singh, J.P., Dwivedi, Y.K., Rana, N.P., Kumar, A., Kapoor, K.K.: Event classification and location prediction from tweets during disasters. Ann. Oper. Res. **283**(1), 737–757 (2019)
7. Alomari, E., Mehmood, R., Katib, I.: Road traffic event detection using twitter data, machine learning, and apache spark. In 2019 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI), pp. 1888–1895. IEEE, August 2019
8. Svensén, M., Bishop, C.M.: Pattern Recognition and Machine Learning, Springer, New York (2007)
9. Liaw, A., Wiener, M.: Classification and regression by randomForest. R. News **2**(3), 18–22 (2002)

10. Tong, S., Koller, D.: Support vector machine active learning with applications to text classification. J. Mach. Learn. Res. **2**(Nov), 45–66 (2001)
11. Chen, J., Huang, H., Tian, S., Qu, Y.: Feature selection for text classification with Naïve Bayes. Exp. Syst. Appl. **36**(3), 5432–5435 (2009)
12. Partalidou, E., Spyromitros-Xioufis, E., Doropoulos, S., Vologiannidis, S., Diamantaras, K.I.: Design and implementation of an open source Greek POS Tagger and Entity Recognizer using spaCy. In: 2019 IEEE/WIC/ACM International Conference on Web Intelligence (WI), pp. 337–341. IEEE, October2019
13. Nadeau, D., Sekine, S.: A survey of named entity recognition and classification. Lingvis. Investig. **30**(1), 3–26 (2007)
14. Adedoyin-Olowe, M., Gaber, M.M., Dancausa, C.M., Stahl, F., Gomes, J.B.: A rule dynamics approach to event detection in twitter with its application to sports and politics. Exp. Syst. Appl. **55**, 351–360 (2016)
15. Alomari, E., Katib, I., Mehmood, R.: Iktishaf: a big data road-raffic event detection tool using Twitter and spark machine learning. Mob. Netw. Appl. **21**, 1–16 (2020)
16. Pohl, D., Bouchachia, A., Hellwagner, H.: Social media for crisis management: clustering approaches for sub-event detection. Multim. Tools Appl. **74**(11), 3901–3932 (2013). https://doi.org/10.1007/s11042-013-1804-2
17. Li, W.J., Yen, C., Lin, Y.S., Tung, S.C., Huang, S.: JustIoT Internet of Things based on the Firebase real-time database. In: 2018 IEEE International Conference on Smart Manufacturing, Industrial & Logistics Engineering (SMILE), pp. 43–47. IEEE, February 2018
18. Ravi, V., Alazab, M., Srinivasan, S., Arunachalam, A., Soman, K.P.: Adversarial defense: DGA-based botnets and DNS homographs detection through integrated deep learning. IEEE Trans. Eng. Manag. Early Access (2021)
19. Jones, A.S., Georgakis, P., Petalas, Y., Suresh, R.: Real-time traffic event detection using Twitter data. Infrastr. Asset Manag. **5**(3), 77–84 (2018)
20. Zulfikar, M.T.: Detection traffic congestion based on Twitter data using machine learning. Procedia Comput. Sci. **157**, 118–124 (2019)
21. Dabiri, S., Heaslip, K.: Twitter-based traffic information system based on vector representations for words. arXiv preprint arXiv:1812.01199 (2018)
22. Dabiri, S.: Tweets with traffic-related labels for developing a Twitter-based traffic information system. Mendeley Data V1 (2018). https://doi.org/10.17632/c3xvj5snvv.1. Accessed 15 Feb 2020.