Real-Time Citizen Problem Detection From Twitter Data Using Naive Bayes Classifier

Nidhi Vanjare [a], Nikita Sarode [b], Rakshita Tantry [c], Amruta Koshe [d], Ramya RB [e]

[a] Nidhi Vanjare, Thane, India, nidhivanjare13@gmail.com

[b] Nikita Sarode, Thane, India, sarode.nikita06@gmail.com

[c] Rakshita Tantry, Thane, India, rakshitatantry2825@gmail.com

[d] Amruta Koshe, Thane, India, amruta.koshe@gmail.com

[e] Ramya RB, Thane, India, ramyarb@apsit.edu.in

Abstract

Urbanization has increased dramatically in recent years, resulting in many city issues, such as pollution, accidents, and traffic. Many people daily face such problems and share about them on social media platforms like Twitter, Facebook, etc. In this paper, the problems faced by residents of a city or their requests are detected by analyzing tweets posted by users containing different localities in a city. Tweets are extracted from Twitter that contain keywords denoting areas, places or localities present in a city (Thane). A Naive Bayes model is trained to detect whether a tweet signifies a problem faced by a citizen or not. The problems are analyzed to get information about any event, citizens' complaints, or requests, and stored in the database along with the location in real-time. The tweets denoting citizens' problems and the locations are obtained from the database and displayed on the front-end interface to make it available to the authorities, to form a smart city management system.

Keywords

Twitter Stream Analysis, Machine Learning, Real-time Extraction, Smart City, Citizen Problem Detection

1. Introduction

In the past few years, the urban population has been growing multifold. Such population growth along with rapid urbanization gives rise to huge challenges for any city. For instance, residents of many Indian cities face a number of problems on a daily

basis including heavy traffic, power outages, water shortages, road accidents, pollution etc. For the sake of this study, the city of 'Thane,' which is located in the Indian state of Maharashtra, is considered.

Simultaneously, with a huge rise in the use of social media, people find it very convenient to share their experiences with others and voice their opinions. Hence, oftentimes, residents of a city use social media platforms to complain or inform about the difficulties that they are experiencing in their localities so that these problems will be noticed and resolved as soon as possible. Various government officials present on these platforms are tagged in many such posts. Twitter is one such platform where people tweet not only about trending affairs but also about problems and issues that they face. Hundreds of tweets are posted daily on Twitter that describe various challenges faced by citizens from different localities of a city. Twitter is considered to be a major area of research as well as an important mode of collecting data. Twitter currently has 396.5 million users and 206 million daily active users. India is the 3rd largest user of Twitter with 22.1 million users where 59.2% of users are between the ages of 25 and 49. Twitter is used by 48% percent of users to stay updated about current news and events happening around the world. Further, tweets are limited to 280 characters, which enhances real-time use of the platform, since people get most information about events in a fast-paced manner as and when they occur. Moreover, tweets are public and contain additional meta-information. Hence, this makes Twitter a suitable platform to be considered for this project's dataset creation for real-time event detection.

Each tweet posted on Twitter is regarded as a Status Update Message (SUM) and contains additional meta-information apart from the text, such as the name of the user, timestamp, geographic coordinates (latitude and longitude), mentions, hashtags, retweet count, and links to other resources. If tweets relating to a certain time period in a limited geographic region, or tweets containing certain hashtags are extracted and correctly analyzed, they provide us with great deal of information about a topic or an event taking place.

However, since thousands of tweets on a variety of topics are posted per second on Twitter, it is very difficult to manually sort out and identify tweets that contain citizen problems faced by the residents of a city. Further, tweets can be considered as irregular and unstructured data, which contains grammatical errors, slangs, and abbreviated words. For this reason, tweets become an incomplete source of information and contain a lot of meaningless data that has to be filtered out. To solve this problem, this study incorporates Data Mining, Natural Language Processing and Machine Learning to build a classifier that will detect tweets posted on Twitter containing problems faced by the citizens of Thane city.

Data mining is the process of sorting through large data sets to identify patterns and relationships in order to retrieve some meaningful information from the source. Twitter API is used for mining text from Twitter. It allows extraction of data from

Twitter, like obtaining every tweet about a certain topic within the last year, or pulling out a particular user's non-retweeted tweets. Tweepy is a Python library which is used for accessing the Twitter API. Mining data from Twitter requires tweepy to create an API object using which we can extract data.

With huge advancements in Natural Language Processing, its utilization can be extended to countless applications for text analysis of various languages. Natural language processing systems can analyze unlimited amounts of text-based data in a consistent, unbiased manner while deciphering ambiguities of language to extract key facts and relationships, or to provide summaries. For this study, the English language is considered. Tweets containing location keywords from Thane region are extracted from Twitter and are processed using Python's Natural Language Processing library - NLTK.

Further, this project incorporates the use of Apache Kafka and Zookeeper for real-time streaming of data in order to extract and detect tweets as and when posted by users on Twitter. Real-time event detection is important for this study due to the nature of the information that needs to be collected. Events taking place in a city like accidents, traffic, and power cut-offs are urgent and require immediate attention. People instantly post about such occurrences on Twitter. Such tweets when collected together can be easily viewed by the authorities responsible for the city, who would be notified of these problems faced by inhabitants of the city and take required actions to resolve these issues.

2. Literature Survey

Extensive research work is done in the field of real-time extraction and analysis of tweets. This is mainly because real-time analysis of data makes it possible to reduce the gap between decision-makers and real-time processing tools. A research paper published by Wladdimiro et al. used zookeeper for real-time extraction. This paper provides a platform based on stream processing systems for analyzing enormous streams of data released in the event of natural disasters. This platform allows for the deployment of disaster-specific apps, provides elasticity in the event of traffic fluctuations, and allows for relocation to improve robustness. Using message brokers such as Apache Kafka to deal with errors and data loss [5].

D'Andrea et al., in their research, proposes a system which retrieves tweets from Twitter based on a variety of search criteria and then analyses them using text mining algorithms before categorizing them. It uses a support vector machine as a classification model, and by solving a binary classification issue, an accuracy of 95.75 percent (traffic versus non-traffic tweets) was obtained. The Java API is supplied by Weka (Waikato Environment for Knowledge Analysis), which we mostly used for data pre-processing and text mining elaboration, and Twitter's API, which allows direct access to the public stream of

tweets. The system, which is based on an SOA, can retrieve and classify streams of tweets as well as alert users to the presence of traffic incidents. Furthermore, the algorithm can determine whether or not a traffic incident is caused by an external factor, such as a football match, procession, or manifestation [4].

The paper published by Goel et al. features an implementation of Naive Bayes using sentiment140 training data from the Twitter database, as well as a strategy to increase classification efficiency. Instead of using only Naive Bayes, they combined the results of Naive Bayes and SentiWordNet, which will be useful in subcomponent technology such as detecting antagonistic, heated language in emails, context-sensitive information detection, spam detection, and determining consumer attitudes and trends [6].

Social networking sites like Twitter have become important sources of current news and events information, and hence are a significant source for data extraction for event detection. The paper published by M. Krstajic et al presents an online method for real-world event detection like natural disasters by analyzing Twitter data. Different textual and frequency components are combined in their study to express semantic features of an event. Visualization is utilized to see which components are important and what effect the parameters have on the event detection algorithm's findings. [11]

The concept of Event Detection is based on the identification of occurrences of a particular event in any system or microblog. Event detection in Twitter is particularly a tedious task since tweets posted on Twitter are flooded with meaningless messages and contain abbreviated words, slangs, emojis, and grammatical errors. Further, more than 15,000 tweets are posted on Twitter per minute which is why a scalable approach is desired for event detection in real-time. The study conducted by Jianshu Weng et al. focuses on these issues and tackles them with Event Detection with Clustering of Wavelet-based signals [10].

A Naive Bayes classifier is based on conditional probability and it assumes that all features are independent of a given class. The paper published by Irina Rush et al analyzes the impact of the distribution entropy on classification error, and it shows that low-entropy feature distributions yield good performance of naive Bayes. The study also concludes that the accuracy of Naive Bayes is not directly correlated with the degree of feature dependencies, but a better predictor of naive Bayes accuracy is the amount of information about the class that is lost because of the independence assumption. Naive Bayes works best with completely independent features and for two opposite cases. [12]

There are numerous methods for extracting information from various types of unstructured data, which include text, video, social media, and predictive analytics. Real-time applications rely on instant input and rapid analysis to make a decision in a short and very specific time frame. A research paper published by Babak Yadranjiaghdam suggests a framework that includes data ingestion, stream processing, and data visualization components, as well as the Apache Kafka messaging system, which is used to perform data ingestion tasks and creates an infrastructure for performing more sophisticated and complicated analytics on streaming data. Spark enables the execution of sophisticated data processing and machine learning algorithms in real-time. The proposed framework provides an infrastructure for real-time processing as well as the ability to extend the analytical capability [9].

A significant amount of tweets are tweeted addressing citizens' problems which are faced by citizens in their day to day lives. By analyzing these tweets a notable amount of problems, city events can be grappled efficiently. Detecting Citizen Problems and Their Locations Using Twitter Data, a paper published by Gizem Abalı et al., extracted tweets along with their locations through tweet texts. Naive Bayes classifier is used to train the machine learning model with two datasets, one with tweets that has event information and the other one with tweets which does not have any event information. Further to this, any web application implementing this model will surely help citizens to speak about their problems collectively [1].

The paper, Social Smart City: A Platform to Analyze Social Streams in Smart City Initiatives, reports on a case study in the real-time acquisition of crime detection information from social media messages, built on top of a platform for fast processing and visualization of Twitter data. This paper presented a platform that uses social media as a data source to help policymakers make decisions in the context of smart city initiatives. The Social Smart City platform collects and analyses real-time Twitter posts in order to improve citizens' experiences. The topology responsible for collecting, processing, and storing Twitter posts was detailed in this paper. Some of the platform's data was analyzed to demonstrate how the platform can be used by a smart city initiative [3].

For various purposes, information can be obtained from social media platforms where people keep on updating about their situation. These microblogs can be of great help in difficult times of disaster. Through the case study, Kumar et al. proposed a tool that can analyze tweets with locations. Using Twitter Streaming API, tweets will be obtained which will further be stored

in a datastore for analysis where data visualization and analysis will be done. Users can navigate through the tweets with various filters like date, locations, etc. Humanitarian Aid and Disaster Relief (HADR) organizations can use this data to get valuable insights and spread awareness [2].

Social media platforms have become easy targets for spammers to trap users in malicious activities. Tools like Google SafeBrowsing can detect and block spam tweets, but these tools cannot help users in real-time. The paper published by Gupta et al. proposes a tool that is able to detect and block spam tweets in real-time with the help of tweet text. This tool would be able to detect and block spam tweets even if spammers create new accounts since this tool uses tweet texts. With four different machine learning algorithms namely - Support Vector Machine, Neural Network, Random Forest and Gradient Boosting results were evaluated for the proposed tool. With Neural Network, an accuracy of 91.65% was obtained [7].

Near real-time Twitter spam detection with machine learning techniques by Nan Sun et al, this paper proposes a near real-time Twitter spam detection system based on an empirical study in which they collected nine mainstream machine learning algorithms and studied their performance in terms of performance, stability, and scalability using different tweet datasets. This system collects tweet data in near real-time, extracts light-weight features from a specific Twitter account, trains a detection model, and visualizes detection results online. In this paper, account-based and content-based features are extracted to aid spam detection. The experiment results show that Random Forest and C5.0 stand out due to their superior detection accuracy, the TPR, the FPR, and the F-measure, and Random Forest performs more consistently than other algorithms. [8].

3. Experimental Setup

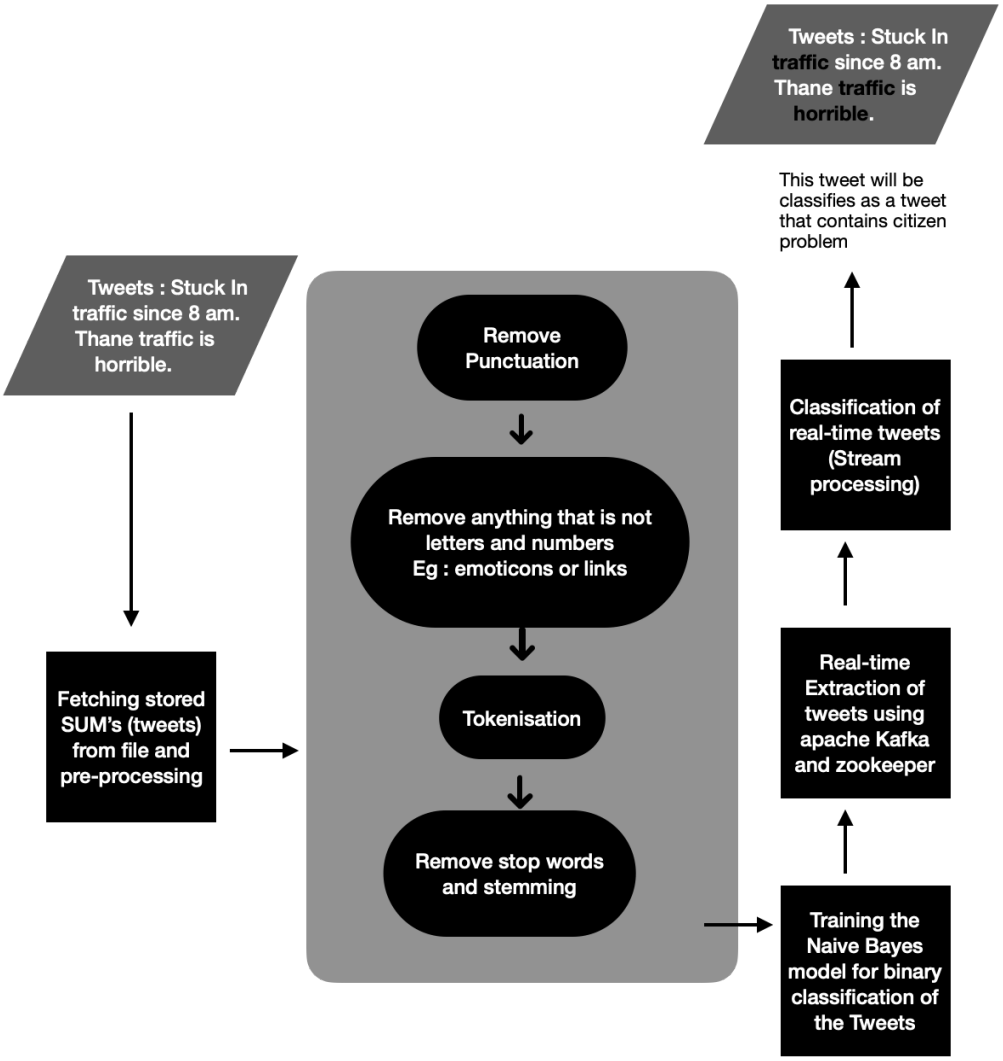


Fig. 1. System architecture for citizen problem detection using twitter in real-time

Tweets which are extracted are pre-processed before model training. Data preprocessing is essential before its actual use because real world data is generally noisy, incomplete, and inconsistent. The quality of data and the meaningful information that can be obtained from data has a direct impact on our model's ability to learn, so preprocessing data before feeding it into our model is critical.

Firstly, we removed punctuation from the tweets as they do not contribute to the analysis of the tweets. The removal of punctuation marks, which are used to divide text into sentences, paragraphs, and phrases, has an impact on the results of any text processing approach, particularly those that rely on the occurrence frequencies of words and

phrases, because punctuation marks are frequently used in text. Before any NLP procedure, stop-words, which are the most regularly used terms in language, are deleted. Stop-words are a group of words that are regularly used without any extra information, such as articles, determiners, and prepositions. We can focus on the important terms instead by deleting these frequently used words from the text.

Similarly, emoticons, characters other than numbers, and letters also do not contribute to text processing as they do not hold any relevant information. Tokenization is also an important process of breaking down text into meaningful chunks. Stemming is a natural language processing approach that reduces inflection in words to their root forms, allowing text, words, and documents to be preprocessed for text normalization. We did face certain limitations and problems such as human error, grammatical mistakes, idioms, vagueness, sarcasm, irony, negation, word ambiguity, multipolarity etc.

Tweet data is a collection of information about tweets (including tweet text, images, retweet counts, location information, date, time, user name etc.). By developing a Twitter application and using token and access ids, data can be retrieved. We used twitter data gathered in Thane city. There are various text mining techniques that work with unstructured data. Twitter API (Tweepy) is used to collect data. Nearly 1% of tweet data from the given area is available via the API itself.

To classify tweets containing problems in a city, a system was trained using a Naive Bayes Classifier, built with Sklearn. Naive Bayes classifiers are a collection of classification algorithms based on Bayes' Theorem. It is a family of algorithms where all of them share a common principle, i.e. every pair of features being classified is independent of each other. This model comes under binary text classification as the tweets were classified into two parts only i.e tweets containing citizen problems and all the other unrelated tweets.

Tweet data is sent to the model using apache kafka and zookeeper to make it real-time. Kafka is an open-source, massively distributed streaming platform. Zookeeper is an Apache centralized service that manages name and configuration data and provides flexible and consistent synchronization inside dispersed systems. It keeps track of

the Kafka cluster nodes' state and includes features like a shared configuration service that allows several clients to read and write at the same time.

The tweets which are classified by our Machine Learning model in real-time are then saved in a mongodb cluster. MongoDB is used due to its various advantages like scalability, cost effectiveness, powerful querying and analytics, and its ability to save unstructured data. This cluster is then connected to the front-end interface which is a website that displays all the problematic tweets. This is done using mongodb queries and node.js. Various filters are also provided that will facilitate the analysis of problematic tweets by a user or government officials in a specific manner.

4. Proposed System

4.1 Explanation

People are using twitter to tweet about problems that they are facing in their locality. Hashtags like #TMC, #trafficsignals, #waterproblems #powercutting, etc are often used in the tweets of citizens of Thane city. We are proposing this website as a platform to display all the tweets that contain citizen problems. This platform can be used as a mode of direct communication between the citizens and the government officials. We are using twitter data to create a platform where all the citizen problems are displayed at one place. Due to this, people will be aware of problems faced by the citizens of their city which may create a force that works as a catalyst and helps them find a solution or get help from relevant authorities.

In this study, tweets were extracted using tweepy, which is a python API, and then manually classified into two sections as mentioned in system architecture . This was done for the purpose of training a dataset. Then, a naive bayes model is trained for classification of tweets into the previously mentioned categories. Using apache kafka, the tweets were then extracted in real time from various locations in Thane. These tweets were classified correctly using the Naive Bayes model and then stored in the back-end which is a mongoDB cluster, which then will be displayed on a website.

4.2 Block Diagram

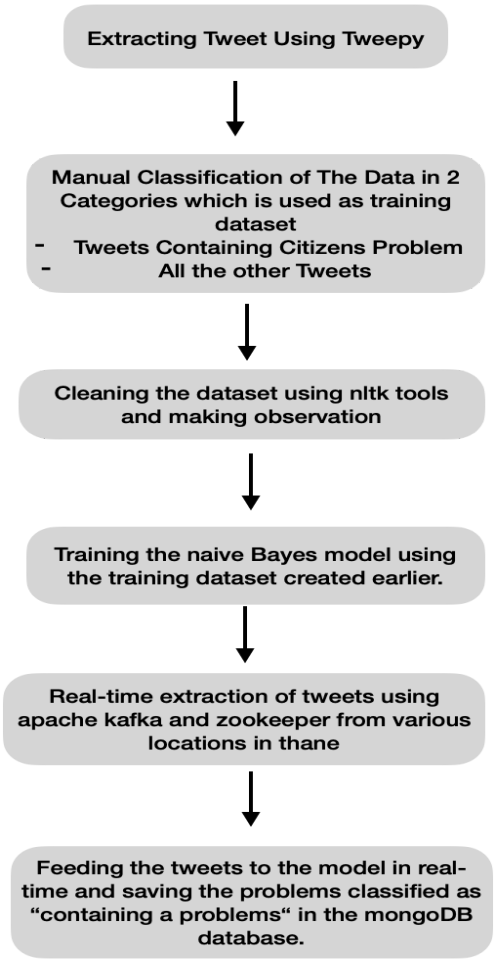


Fig. 2. System architecture for traffic detection from Twitter stream analysis.

4.3 Methodology

STEP 1: Extracting tweets using tweepy from various locations in Thane city.

STEP 2: Manually classifying the data for training purposes into two parts i.e. one with tweets containing citizens' problems and one with all the other unrelated tweets.

STEP 3: Cleaning the data using nltk tools like removing stopwords, stemming, lemmatization etc and making important observations.

STEP 4: Training Naive Bayes model with the training dataset created in step 2.

STEP 5: Saving the trained model in pickel.

STEP 6: Real-time extraction of tweets, specific to our needs, using apache Kafka and zookeeper.

STEP 7: Feeding the tweets to the previously saved naive Bayes model (step 5) in real-time.

STEP 8: Saving the tweets in MongoDB database on cloud after classification of the tweets by the model.

4.4 Naive Bayes Model

The Naive Bayes model assigns class labels to problem instances. There are 2 class labels in our study - “tweet containing problems” and “unrelated tweet”. The Naive Bayes model is based on the Bayes theorem and is used for solving classification problems. Given the class variable, all naive Bayes classifiers assume that a feature's value is independent of the value of any other feature.

Naive Bayes is a conditional probability model, which is decomposed as -

$$posterior = \frac{prior \times likelihood}{evidence}$$

Bayes Rule is based on the concept of deriving a hypothesis (H) from the given evidence (E). It is connected with two ideas: the probability of the hypothesis before getting the evidence, P(H), and the probability of the hypothesis after getting the evidence, P(H|E). Bayes rule is given by the following equation:

$$P(H|E) = \frac{P(E|H) \times P(H)}{P(E)}$$

Using the following Naive Bayes equation, we calculate the probability of our input text belonging to two different classes - “tweet containing problem” and “unrelated tweet”. Naive Bayes model works faster on huge datasets and performs well in case of categorical input and output. These features are advantageous for a real-time detection system, which is why we have opted for a Naive Bayes classifier for our study.

4.5 Expected Output

The end result is a website that will display the tweets containing citizen problems that users have posted along with the location. Furthermore, we are also going to provide various filters so that people can use the website for specific research areas. For example, if someone is looking for electricity problems faced by societies in 'Vasant Vihar', then "location" and "type of the problem" filters can be applied to quickly retrieve such information. If TMC wants to look into problems faced by people in Thane in the last 5 years, the by the "Date-time" filter. This study facilitates bridging the communication gap between the citizens and the government and helping them in finding solutions to the problems.

5. Conclusion

In this research, we present a method for detecting citizen problems - related events in real-time using Twitter stream analysis. Detected Tweets are successfully displayed on a front-end interface from the cloud. Kafka and zookeeper have aided in the real-time extraction of tweets which were then fed to the Naive Bayes Model. This research will help people and respective authorities to keep track of citizen problems in the locality. This real-time analysis allows authorities to take rapid action, enabling them to be proactive by grabbing opportunities or averting problems before they occur, to react quickly, and to be aware of what problems they are dealing with at any given time. It can also assist them in anticipating future requirements of an area so that they can obtain a better understanding and help avert problems before they occur.

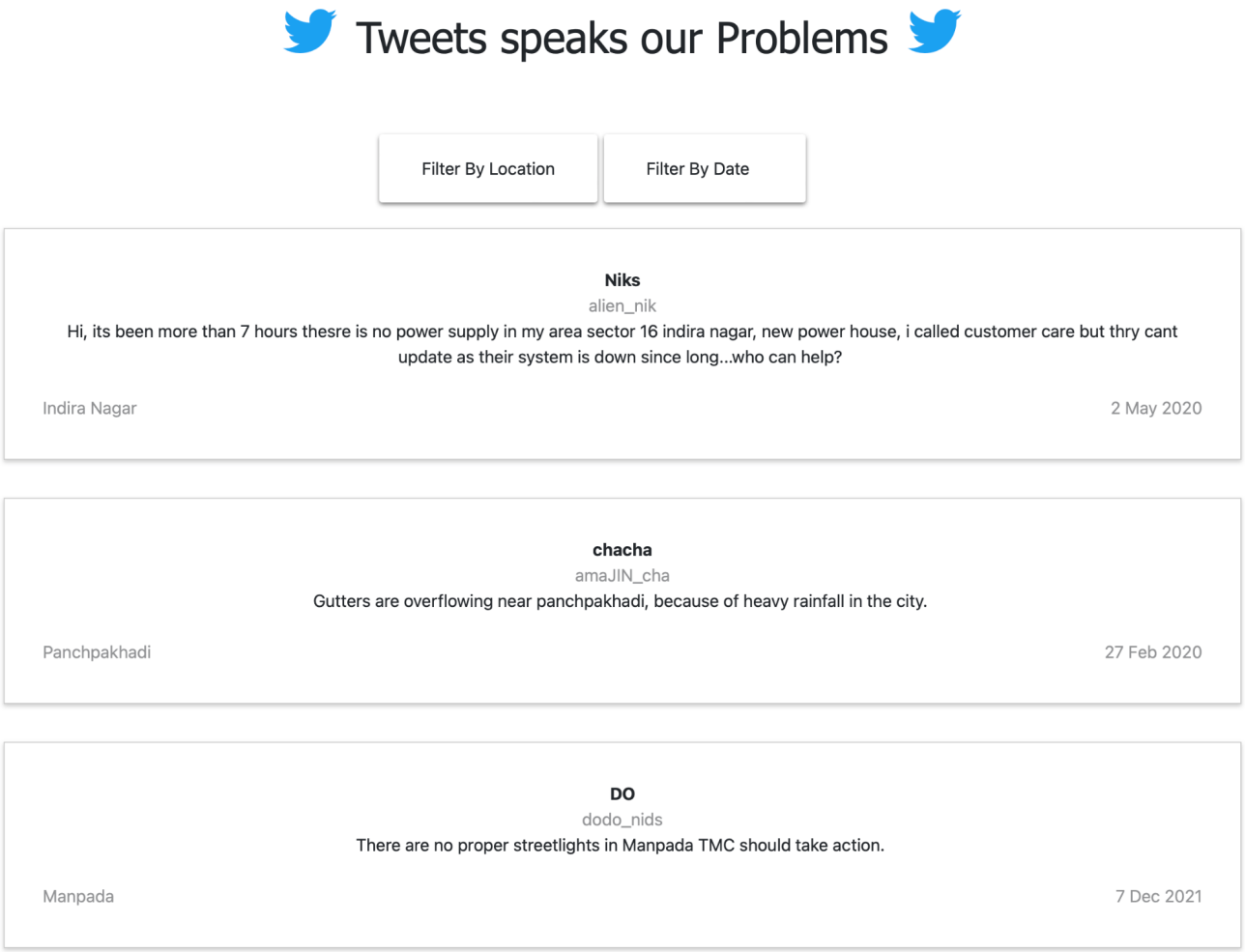
6. Results

We have trained our naive Bayes model on approximately 700 tweets which we extracted and manually classified into 2 types. We have obtained an accuracy of 88%, after a lot of tuning of the dataset. The recall of the model is 90% which means most of the tweets containing the citizen problems are detected. Here the focus is more on recall than precision as we don't want any tweets containing citizen related problems to go undetected by the model. The percentage of problematic tweets that are relevant is referred to as precision which is approximately 80% in case of the ML model [Table 1].

Accuracy score	0.8846153846153846
Precision score	0.7934782608695652
Recall score	0.9012345679012346
F1 score	0.8439306358381502

Table 1. Accuracy, Precision, Recall and F1 Scores

All the classified problematic tweets saved in MongoDB cluster are then displayed on the webpage and get updated in real-time.



7. REFERENCES

[1] G. Abalı, E. Karaarslan, A. Hürriyetoğlu, F. Dalklç, “Detecting Citizen Problems and x`Their Locations Using Twitter Data,” in 6th International Istanbul Smart Grids and Cities Congress and Fair (ICSG), pp. 1-4.

[2] Kumar, S., Barbier, G., Abbasi, M., & Liu, H. (2021). TweetTracker: An Analysis Tool for Humanitarian and Disaster Relief. *Proceedings of the International AAAI Conference on Web and Social Media*, *5*(1), 661-662.

[3] A. Souza, M. Figueredo, N´elio Cacho, Daniel Ara´ujo, Jazon Coelho and Carlos A. Prolo, “Social Smart City: A Platform to Analyze Social Streams in Smart City Initiatives,”

[4] D’Andrea, E., Ducange, P., Lazzerini, B., & Marcelloni, F. (2015). Real-Time Detection of Traffic From Twitter Stream Analysis. IEEE Transactions on Intelligent Transportation Systems, 16(4), 2269–2283.

[5] Wladdimiro, D., Gonzalez-Cantergiani, P., Hidalgo, N., & Rosas, E. (2016). Disaster management platform to support real-time analytics. 2016 3rd International Conference on Information and Communication Technologies for Disaster Management (ICT-DM).

[6] Goel, A., Gautam, J., & Kumar, S. (2016). Real time sentiment analysis of tweets using Naive Bayes. 2016 2nd International Conference on Next Generation Computing Technologies (NGCT).

[7] Gupta, H., Jamal, M. S., Madisetty, S., & Desarkar, M. S. (2018). A framework for real-time spam detection in Twitter. 2018 10th International Conference on Communication Systems & Networks (COMSNETS).

[8] Sun, N., Lin, G., Qiu, J., & Rimba, P. (2020). Near real-time twitter spam detection with machine learning techniques. International Journal of Computers and Applications, 1–11.

[9] Yadranjiaghdam, B., Yasrobi, S., & Tabrizi, N. (2017). Developing a Real-Time Data Analytics Framework for Twitter Streaming Data. 2017 IEEE International Congress on Big Data (BigData Congress). doi:10.1109/bigdatacongress.2017.49

[10] J. Weng and B.-S. Lee, “Event detection in Twitter,” in Proc. 5th AAAI ICWSM, Barcelona, Spain, 2011, pp. 401–408.

[11] M. Krstajic, C. Rohrdantz, M. Hund, and A. Weiler, “Getting there first: Real-time detection of real-world incidents on Twitter” in Proc. 2nd IEEE Work Interactive Vis. Text Anal.—Task-Driven Anal. Soc. Media IEEE VisWeek,” Seattle, WA, USA, 2012.

[12] Irina Rish, "An Empirical Study of the Naïve Bayes Classifier," no. January 2014.