SOCIAL NETWORK ANALYSIS OF LOCAL TRAFFIC POLICE DEPARTMENT

A report submitted in partial
fulfilment of the requirements for the degree of
Bachelor of Technology
in

Computer Science & Engineering

by

MOHAMMED ALLAMA HOSSAIN
RASHIKA DAGA

Under the supervision of

PROF. DR. SAPTARSI GOSWAMI

Assistant Professor

A.K. Choudhury School of Information Technology

Calcutta University

Department of Computer Science & Engineering
Institute of Engineering & Management
Gurukul, Y-12, Block –EP, Sector-V, Salt Lake Electronics Complex
Kolkata-700091, West Bengal, India

MAY, 2018

ACKNOWLEDGEMENT

We should like to take this opportunity to extend our gratitude to the following revered persons without whose immense support, completion of this project wouldn't have been possible.

We are sincerely grateful to our advisor and mentor **Prof Dr. Saptarsi Goswami** of the **A.K. Choudhury School of Information Technology**, **Calcutta University**, for his constant support, significant insights and for generating in us a profound interest for this subject that kept us motivated during the entire duration of this project.

We would also like to express our sincere gratitude to **Prof. Dr. Satyajit Chakrabarti** (Director**,**IEM)**, Prof. Dr. Amlan Kusum Nayak (**Principal,IEM) and **Prof. Dr. Debika Bhattacharyya**, HOD of **Computer Science and Engineering** and other faculties of Institute of Engineering & Management, for their assistance and encouragement.

Last but not the least, we would like to extend our warm regards to our families and peers who have kept supporting us and always had faith in our work.

Mohammed Allama Hossain
Reg. No: 141040110087
Dept. of Computer Science & Engineering
Institute of Engineering & Management

Rashika Daga
Reg .No: 141040110122
Dept. of Computer Science & Engineering
Institute of Engineering & Management

Table of Contents

ABSTRACT

With the increasing social media buzz, connectivity, interaction and power, it is highly important to utilize the insights generated from these sites for better performance, expansion and opportunities. Police departments are leveraging these online platforms to spread awareness, connect to the mass, know their concerns and opinions to use their potential accordingly for better satisfaction. We have considered the official Facebook page maintained by Kolkata Traffic Police Department for analysis. The empirical study was conducted by extracting page owner posts, visitor posts, comments on page owner posts and user reviews by executing a scheduled R script on a daily basis and storing and updating the records in a database. We analyze the user engagement statistics, popular post types, trends in traffic updates of Kolkata, major concerns of the citizens and sentiments of the public. Our results provide practical useful insights for the users, as well as the community who are working on the policing frameworks.

Keywords: Social Media Analysis · User Engagement · Kolkata Police Department · Sentiment Analysis

INTRODUCTION

According to the report generated in 2017 by Statista – an online statistics, market research and business intelligence portal, social media users comprised of 71% of the overall internet users and it's estimated that the number of social media users will reach 2.77billion (bn) from 2.46bn users [1]. The portal also presented statistic displaying the digital population in India as of January 2018. During this period of time, it was found that the country had 250 million active social media users during the measured time period. Internet penetration, ease of use and increase worldwide usage of smartphones has been responsible for this virtual connectivity among people. Online connectivity and user engagement provides opportunities to extract meaningful information about personality [2, 3], sentiments [4-6], and opinions [6].

Facebook (FB) has the current lead in social media market with approximately 2.19 bn monthly active users as of the first quarter of 2018[1]. Apart from the marketing campaigns [7], publicity of movies [8], and current news updates [10], Facebook has been widely used by government bodies, police departments for community outreach, and safety concerns [9]. It is important to harness this powerful tool that supports quick two way interactions between the authority and the public quickly, establishing a mutual beneficial net. The interactions have been proved very effective at the times of emergencies [11, 13], and political elections as well [12].

In our research work, data was mined from the official Facebook page maintained by Police Department of Kolkata, Kolkata Traffic Police (KTP), which is mainly a forum for Traffic Alerts and Public Grievances. With approximately 1,54,000 followers, the officials have been quite active in the last 2 years to increase the public engagement, effective problem solving via complaints on Facebook. Our project work aims to convert the unstructured data (data from Facebook posts) to structured data for better understanding, visualization of data and summarize the results in an effective manner. We also intend to highlight the views of the public via the reviews made on the KTP. This will help the viewers to get the crunch of the matter quickly, and more importantly, this will be of a great help to the Kolkata Police Department and other concerned authorities in analyzing the information that was being generated directly or indirectly via the page. It will help in increasing the quality of their content, and this brief overview will help in highlighting the shortcomings and positive results from the page. Besides the work of extracting information from preprocessed data after applying various techniques, there were other challenges that needed to be overcome to produce approximately accurate results.

BACKGROUND STUDY

Government organizations are quickly adopting social media because of the ease with which they can communicate with a large section of the citizens. Government officials seek to leverage these resources to improve services and communication with citizens, especially segments of the population that previously were difficult to reach and underrepresented [14]. The easy availability of smart devices and the increase in the number of people using the Internet has resulted in this change.

Analyzing public opinions and grievances is a task relevant and crucial to every police department, and the advent of social media has only made it easier. Traditional methods which included on-ground survey, phone or mail survey was cost prohibitive and it could not capture the views of a large section of the society over a long period [15].

Social media has also made it easier to report and manage routine and critical conditions. According to Craig Fugate (the administrator of the US Federal Emergency Management Agency), during the 2010 Haiti earthquake, even when an area's physical infrastructure was destroyed, the cellular tower bounced back quickly, allowing survivors to request help from local first responders and emergency managers to relay important disaster related information via social media sites [16].

However, even though the organizations have these powerful tools at their disposal, they are still learning how to use them effectively. One problem that the organizations often face is the lack of engagement with the citizens. Government agencies are also overwhelmed by the amount of data that is generated on social media sites and they are unable to channel and use the data effectively. Algorithms that could summarize the data in terms of keywords or visual representation would come in very handy to the institutions.

Government agencies and police departments also post a significant amount of information each day which include new policies and schemes, traffic updates and cause of congestion, important events in the city, critical and emergency warnings, but these information are again not utilized properly by the citizens due to lack of engagement strategies.

Our project work aims to solve these problems by representing large amounts of information in terms of visualizations and finding important insights from social media sites of government bodies which could help both the authorities as well as the citizens.

LITERATURE SURVEY

Government bodies are adopting social media at a fast rate because of its ability to engage in crisis communication, to manage a crisis, to enhance the communication between the government and the citizens, to promote openness and transparency which in turn reduces corruption and increases credibility and to improve the efficiency of the organization [17-20].

Denef et. al. illustrated the benefits and challenges of the instrumental strategy as adopted by the London Metropolitan Police (MET) and expressive strategy as adopted by the Greater Manchester Police (GMP). The instrumental approach followed the primary policing functions and had lower maintenance as compared to the other approach but it suffered from loose relations with the public, lower following and hence lower potential to acquire information. The expressive approach on the other hand created closer relations with the public, higher following and hence greater outreach but it suffered from high maintenance [21].

Meijer and Thaens studied the ‘push strategy’ used by Boston Police Department, the ‘push and pull strategy’ used by Metropolitan Police Department in DC and the ‘networking strategy’ used by Toronto Police Service and concluded that social media strategies of police departments were different because pre-existing differences in communication strategies were reinforced [22]. Mossberger et. al. [23] have studied the use of various interactive social network tools used in US and the work also suggests that “push” strategies are more dominant than networking and pull strategies in Facebook and Twitter.

Sara Hofmann and co-authors analyze the Facebook sites of German local governments using a multi-method approach and depict the success of communication between the government and users in terms of frequency and polarity of citizen’s reactions (sentiment analysis). Social Media offers the benefit of attaching multimedia features to a normal post to make it more attractive and gain popularity [24]. They conclude that posts that contain photos or videos are more liked or commented than those containing textual updates or a link. This shows the multimedia feature of Social media helps to capture the attention better.

METHODOLOGY

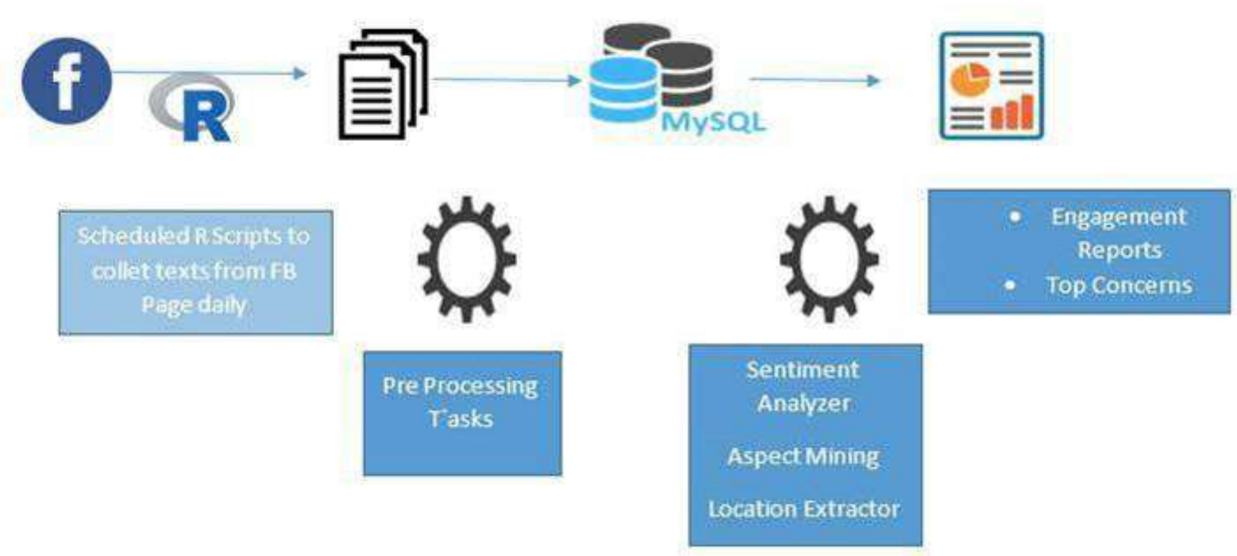

Figure 1. Work flow deployed in the project

Data Collection

The 'Rfacebook' package available in the CRAN repository was used to extract data from the official Facebook page of Kolkata Traffic Police Department (Kolkata Traffic Police). The package provided an interface to the Facebook API via R [29]. The data was a combination of unstructured (status, comments) as well as structured data (like count, comment count, and share count). However the reviews had to be extracted manually as there is no API to return reviews of a Facebook Page.

The extracted data include Page owner posts (posts by the owner of the page), visitor posts, comments on the page owner posts, and reviews of the page. Attributes of page owner post and visitor post that were used in developing the further algorithms and for keeping track of the records areid of the page or user making the post, name of the page or user making the post, message, created time, type (status, link, photo, video), id of the post, like count, comment count and share count. Attributes of the reviews of a page include id of the post, message, rating, and created time.

The data was collected on a recurring basis (daily) by a scheduled script execution using the package 'taskscheduleR' from the CRAN repository. The package allows scheduling R scripts and processes with the Windows Task Scheduler. The collected data was then dumped into a database, where new records were inserted into the database and existing records were updated in the database by means of a unique post_id provided by the Facebook API. If a post_id already existed in the database, then the like_count, comment_count and share_count was updated in the database. The reviews in the work are extracted manually.

Data is stored in the MySQL database using 'RMySQL' package from the CRAN repository. Databases designed for storage of data initially has been shown below with a few records inserted into it.

Page Owner Post:

sequence_key	id	message	type	created_time	link	likes_count	comments_count	shares_count	flag
2018-05-06	129115403803409_1000111740037100	NA	photo	2016-06-15T11:01:52+0000	https://www.facebook.com /KolkataTrafficPolice /phot...	9	0	2	1
2018-05-06	129115403803409_1000588596656081	NA	photo	2016-06-16T12:08:40+0000	https://www.facebook.com /KolkataTrafficPolice /phot...	8	1	1	1
2018-05-06	129115403803409_1001101066604834	NA	photo	2016-06-17T09:57:31+0000	https://www.facebook.com /KolkataTrafficPolice /phot...	12	0	0	1
2018-05-06	129115403803409_1001717696543171	NA	photo	2016-06-18T14:19:59+0000	https://www.facebook.com /KolkataTrafficPolice /phot...	23	4	10	1
2018-05-06	129115403803409_1002611436453797	Pl. find the uploaded action taken report on the b...	photo	2016-06-20T10:12:32+0000	https://www.facebook.com /KolkataTrafficPolice /phot...	10	0	0	1

Figure 2. Database Records for the table 'page_owner_post'

Visitor Post:

sequence_key	id	message	created_time	likes_count	comments_count	shares_count	flag
2018-05-13	129115403803409_10152046223607546	Came to know this page today and wanted to post a ...	2014-04-22T15:45:59+0000	1	1	0	1
2018-05-13	129115403803409_10152048217753517	Worth sharing pic !	2014-04-03T19:19:50+0000	1	1	1	1
2018-05-13	129115403803409_10152054817153371	Parking is being collected in Alipore Road, the la...	2014-02-25T18:16:32+0000	2	3	1	1

Figure 3. Database records for the table 'visitor_page'

Reviews:

sequence_key	id	message	rating	created_time
2018-05-12	51	Entire team are very hlpfull we get sefty..thnk u ...	5	2017-11-21T11:44:00+0000
2018-05-12	52	I proud to be Kolkata traffic police.It is faster ...	5	2017-12-03T17:54:00+0000
2018-05-12	53	Human In Khaki Always at Your Service. We must app...	5	2017-09-03T21:43:00+0000

Figure 4. Database records for the table 'reviews'

Data Preprocessing

Facebook seems to struggle with the issue of a post being posted multiple times due to slow internet connection and sometimes an update is posted multiple times by a user as a result of human error as well. This resulted in a lot of redundant data in the database which had no significance. These records had to be removed from the database using the ‘duplicated’ function present in the ‘base’ package of R programming. Furthermore, posts that were posted in vernacular languages like Bengali and Hindi had to be removed because currently there is no package which can process them. The posts in vernacular languages were removed using ‘detect_language’ function from Google’s Compact Language Detector 3, ‘cld3’ package available in CRAN repository.

The text mining, ‘tm’ package was used to clean the text to make it suitable for the purpose of analysis. Predefined transformation functions were used to remove whitespace, numbers, stop-words and punctuations. Custom defined functions deploying pattern matching algorithms on regular expressions were used to remove URL and email-ids. Finally stemming was performed to reduce all the words to their root form and get the common origin.

User Engagement Statistics

The police department certainly would want to reach out to the maximum number of citizens of their jurisdiction, so that most of them can benefit from the information posted by the police department. Analyzing user engagement statistics and citizen outreach would have been accurate if Facebook API provided the number of views each post got and the relative increase in the number of views. However since that option is not available, we have used like, comment and share count as the surrogate indicator.

Statistics such as like, comment and share count of status, link, photo and video updates were obtained for regular intervals to study the engagement of the police department with the citizens. Other measures that were used to analyze the engagement are average number of posts, posts on weekdays and weekends and increase in followers. Posts were classified based on type, likes generated, and content. Based on type, posts were classified into status, link, photo and video updates. Based on content, posts were categorized as traffic updates and other updates.

Posts that generated less than ten likes were classified as ‘no engagement post’ and post that generated more than five percent of the total likes garnered by the page during a time interval was classified as ‘viral post’. The ratio and content of ‘no-engagement posts’ and ‘viral posts’ also indicated a direct relation with the engagement statistics. User engagement statistics were simply derived by manipulating with various counts like likes, comments and shares count, along with the types of posts and number of posts generated to calculate the interaction factor of any period. User engagement statistics is beneficial in validating facts that are already known such as the multimedia features of a social networking site makes the connectivity easier and increases the interaction quotient. User engagement statistics are necessary for getting an overall summary of how the page has performed in a given duration, and offers an insight into how the admin/ owner of the page improve the interaction between the page and the public, and offer higher level of satisfaction. The statistics can be obtained for both monthly and weekly basis. The start date and end date can be entered by the user and the monthly/weekly intervals are calculated by an algorithm. The counts are stored in a .csv file initially that are processed to generate graphs for easy user understandability and deducing conclusions quickly by looking at the graphs.

Congested Locations

A location database was created in which names of places in Kolkata were stored. ‘geocode’ function from the ‘ggmap’ package available in the CRAN repository was then executed for each record of the database and the respective latitude and longitude were stored in the database.

Firslty, some data cleaning and preprocessing is done on the status updates. It involves:

- Removal of double spaces and “/r/n”
- Removal of punctuation marks
- Stop word elimination
- Converting the message to lower case

Traffic updates were filtered from the page owner posts by matching keywords such as “traffic”, “update” and “alert”. The method works yielding high accuracy since Kolkata Traffic Police have a certain template of sending out Traffic updates and this consistent pattern has made it easier for us to track traffic jams and congestions. All the locations available in the database are first matched to the traffic update to see if any or some of the locations are present in the textual update. If a match is found, then the location is stored in a dataframe along with the longitude and latitude. Otherwise the traffic update posts are checked if they contain words like “road”, “street”, “avenue”, “Sarani”, etc. and two, three and four word strings are generated and checked if the location exists or not. In the algorithm, unnecessary words are removed from the status update. If the location exists, then it is added to the database along with the location coordinates. This way, the location database keeps on improving by inserting places which were not previously available in the database.

Main Concerns

A domain dictionary was created in which important words along with their count were stored. The words initially stored in the dictionary were concerns that were common in many of the posts that were examined to get an overview. The 'dictionary_maker.R' file was then executed to find out important keywords that occurred frequently in the visitors post. Words obtained from the 'dictionary_maker.R' script, having count above a certain threshold was then added to the domain dictionary.

Concerns were obtained by finding out 'Adjective-Noun' pair tags and then classifying them as either primary concern or secondary concern. If the noun tag contained a word from the domain dictionary, then it was classified as a primary concern otherwise it was classified as a secondary concern. The parts of speech of a sentence structure was obtained using the POS tagging algorithm in the 'NLP' and 'openNLP' package available in the CRAN repository.

Sentiment Analysis

Sentiment Analysis is the means to determine whether a given text is positive, negative or neutral. Sentiment analysis was used to understand the feelings and views of the users towards Kolkata Traffic Police and its initiatives. The algorithm was applied to the reviews and the visitors post. Based on the opinion words present in the 'sentimentAnalysis' package, a score was assigned to each feature. The results of several sentiment analysis algorithms were combined to yield good results.

RESULTS AND DISCUSSION

User Engagement Statistics

The graphs for User Engagement Statistics showcased below have been deduced for the period of 1st March 2017 to 31st December 2017 on a monthly basis. Number of intervals obtained in the period is 10.

Number of Posts

Bar chart (Figure 5) shows the number of posts made by KTP during the period. Higher the number of post, more is the chance of engaging to users, since users will be constantly notified about traffic updates, and other related updates, for e.g. simply a post talking about Durga Puja, safety concerns for public, etc.

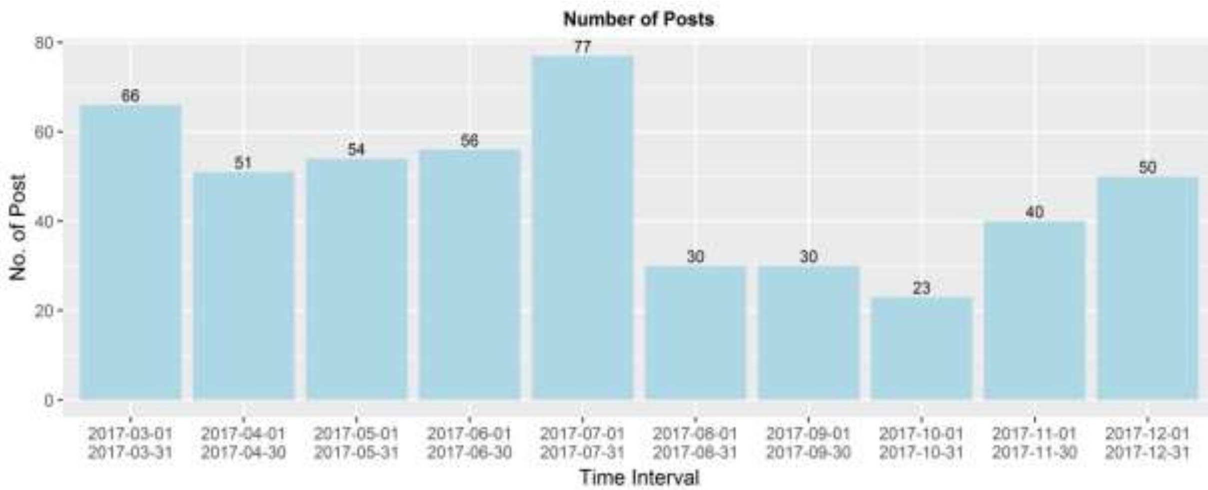

Figure 5: Number of posts in the interval (01-03-17 - 31-12-17)

Average No. of posts

Figure 6 shows the average number of posts updated by KTP on its official page daily. The figure depicts that KTP tries to post at least 1-2 posts per day. Month intervals that record less average posts will have less interaction than the other intervals, but it is not necessary that the relation is directly proportional since it depends on the type of posts that KTP has posted, and its content.

No. of posts on Weekdays and Weekends and Like counts

The number of posts updated by KTP during weekdays is much greater than the number of posts made during weekends, and consequently, the like count for weekend is much less than like count of weekdays. KTP does not post frequently on weekend even though people are more actively involved on these social network platform on weekends compared to weekdays. KTP can effectively exploit this behavior to post content rich posts, videos with message on weekend to gain popularity more easily. (Figure 7, 8)

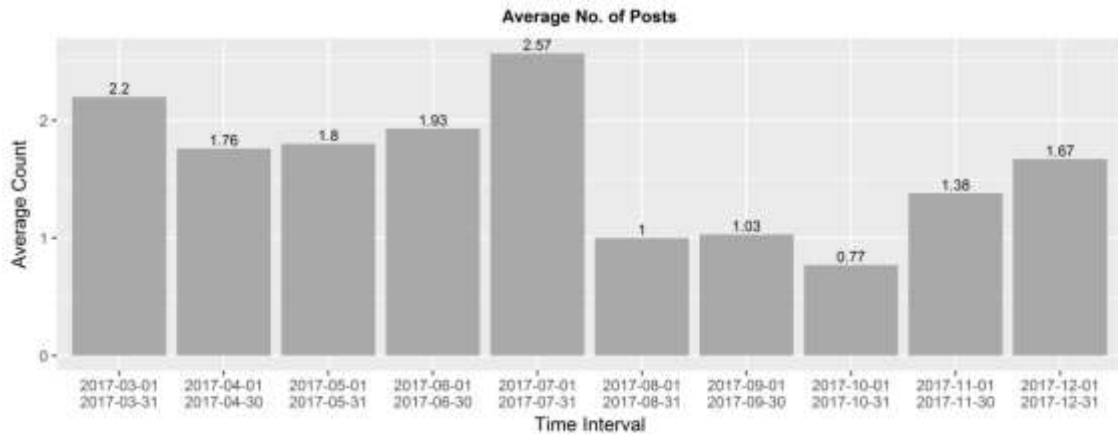

Figure 6 Average no. of posts in the interval (01-03-17 - 31-12-17)

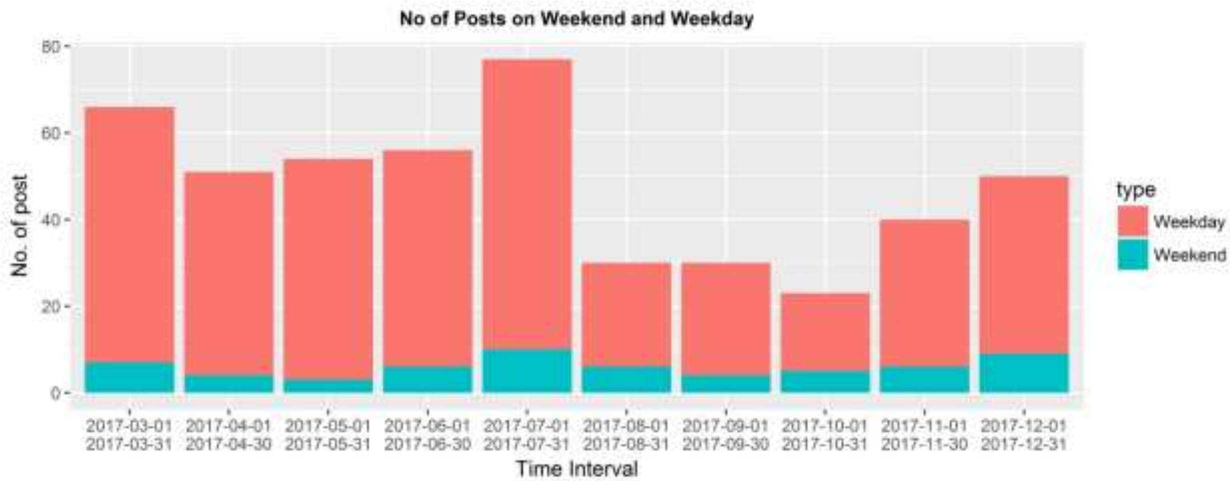

Figure 7 No. of posts on weekends and weekdays in the interval (01-03-17 - 31-12-17)

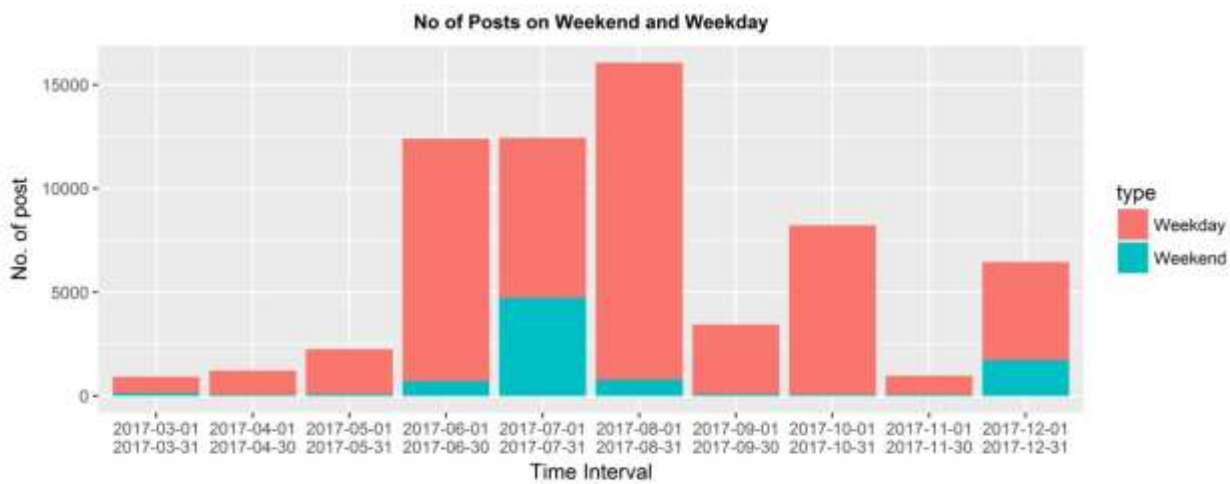

Figure 8 No. of posts on weekends and weekdays in the interval (01-03-17 - 31-12-17)

Distribution of Likes, Comments and Shares

Figure 9 and Figure 10 shows the like count, and share count received during the period in monthly intervals. Comparing the two bar graphs, it is evident that there has been a lot of activity during the month of August and September, which is reasonable by the average no. of posts made by it.

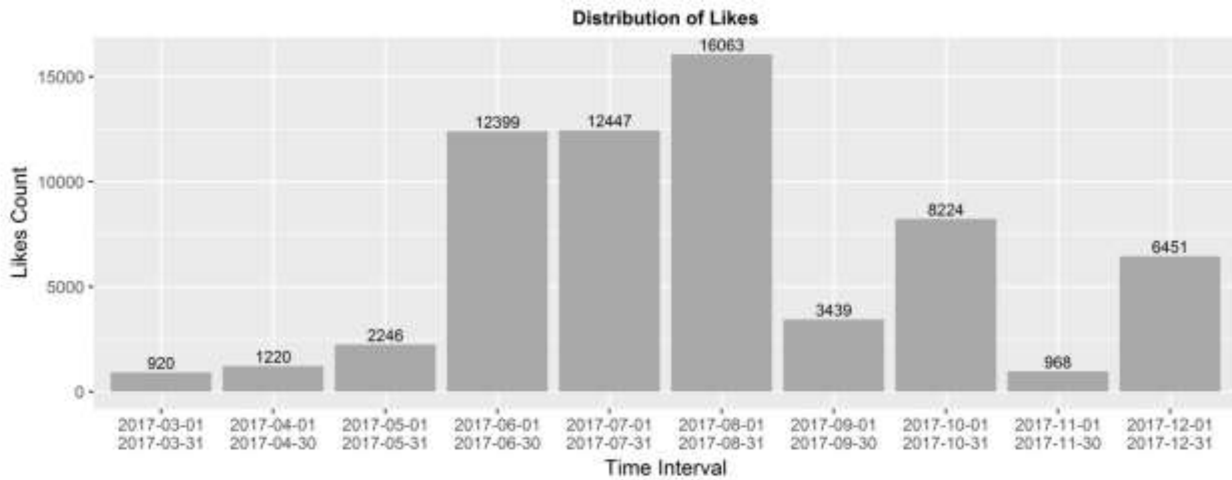

Figure 9 Distribution of likes in the interval (01-03-17 - 31-12-17)

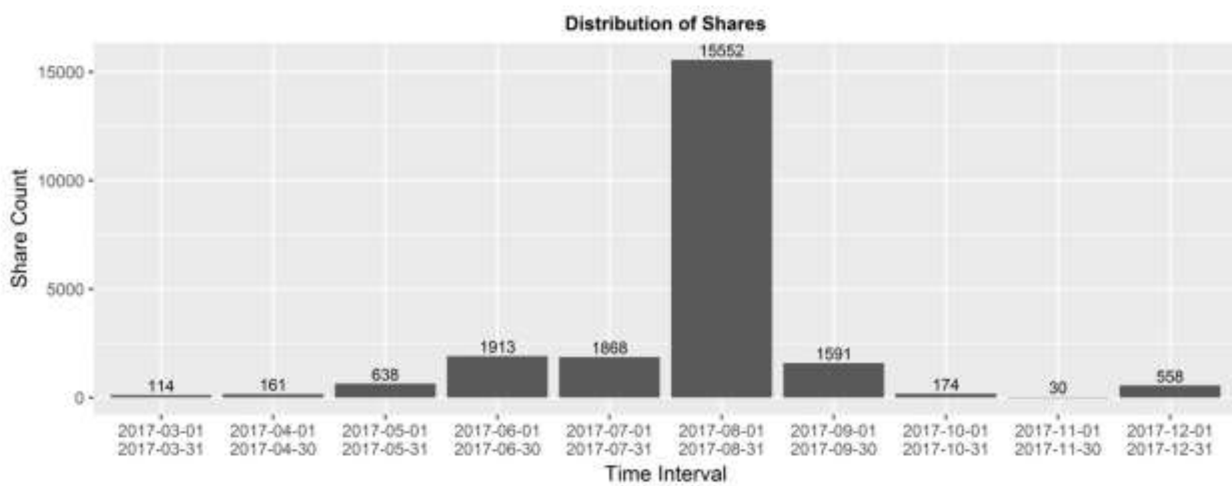

Figure 10 Distribution of shares in the interval (01-03-17 - 31-12-17)

Types of Post

Figure 11 shows the breakup of the no. of posts with respect to its type. August had a very high number of shares, and it can be justified by the fact that KTP had posted a video that went viral on Facebook. There have been video updates in the later intervals, however, they did not catch attention, and the stats validate the fact. For intervals, that maintained a high record of photo updates, like, comment and share count goes automatically higher than the intervals containing more of status updates. Thus, the multimedia features (photo and video updates) are capable of drawing users' attention. Figure 12,13 give a further understanding of the attention drawn by every type of post in an interval by using likes as a surrogated indicator. Intervals during which photos and videos are posted frequently, they draw the attention from the public. Textual (status) updates give negligible amount of interaction.

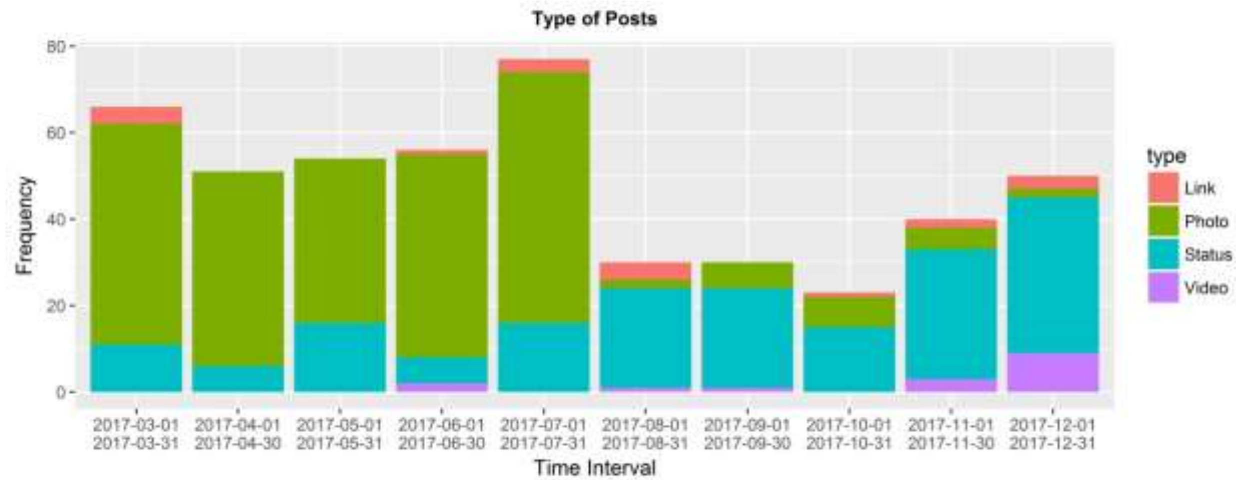

Figure 11 Distribution of types of post in the interval (01-03-17 - 31-12-17)

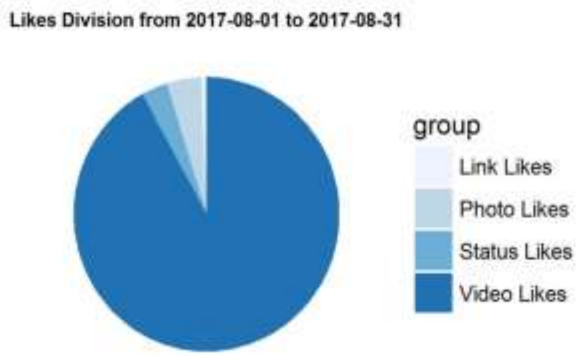

Figure 12 Likes obtained for period (01-08-17 - 31-08-17)

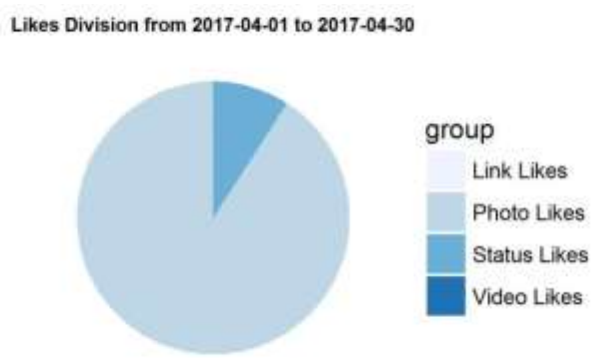

Figure 13 Likes obtained for period (01-04-17 - 30-04-17)

Interaction

It is important to note that it is better to conclude in relative terms, rather than considering only the counts (like, comment and share) count of a post, or the number of posts, since they both are dependent on each other. Thus, we define a parameter, interaction that is given by:

$$interacion = {}^{No.of\ likes\ in\ a\ given\ interval}/_{No.of\ posts\ in\ the\ given\ interval}$$

(w.r.t likes)

$$interaction = {}^{No.of\ comments\ in\ a\ given\ interval}/_{No.of\ posts\ in\ the\ given\ interval}$$

(w.r.t. comments)

$interaction$: ${}^{No.of\ shares\ in\ a\ given\ interval}/_{No.of\ posts\ in\ the\ given\ interval}$ (w.r.t shares)

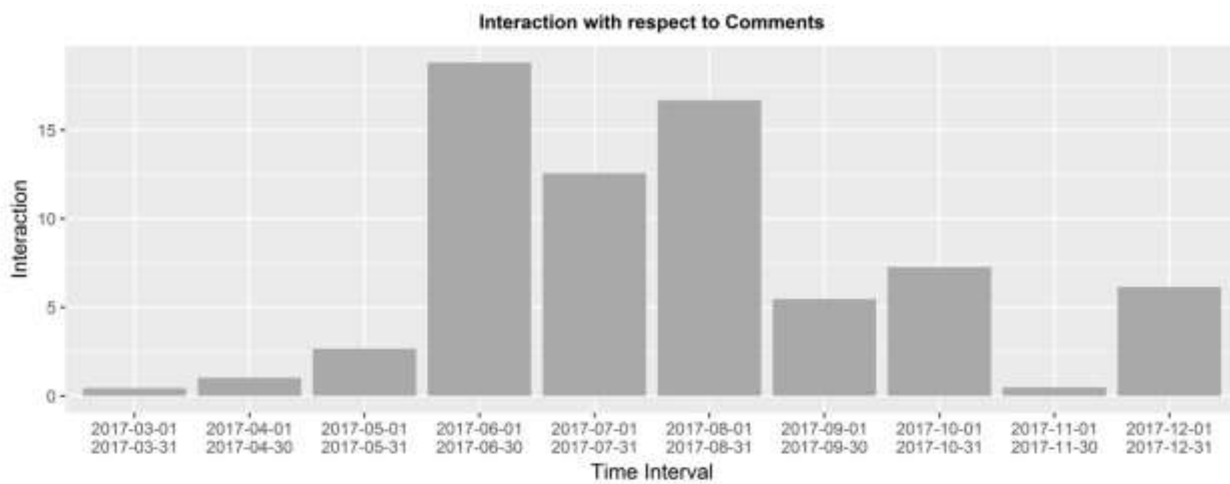

Figure 14 Interaction w.r.t. comments (01-03-17 - 31-12-17)

Figure 14 shows the interaction w.r.t comments and shares for the intervals and the result is as expected, August and June has a very high interaction quotient compared to other intervals.

Viral and No Engagement Posts

Viral and No Engagement posts can be viewed on entering the start and end date. The system returns the list of post ids for each month interval. Traffic police can certainly take advantage of the two in the following way:

- Viral Posts can be referred while posting updates since they have caught a significant amount of public attention
- No Engagement Post gives a brief idea on the content of posts that do not gain attention from the public, and KTP should think on ways on improving such kinds of posts.

Traffic Locations

The algorithm takes the start date and end date from user and returns a google map displaying the congested location for the period. Maps for two such instances are shown below. Figure 14 is the map obtained for congested places from 1st Novemeber2017 to 31st December, 2017. Congested locations' map is useful to both the parties (Kolkata Traffic Police and public) in the following way:

- The users can get a quick idea of congested places of a given date range quickly
- The police force can check frequently congested areas and their effectiveness in controlling traffic
- The users and the police force can determine the frequently congested area as the size of the point determines the frequency of congestion.

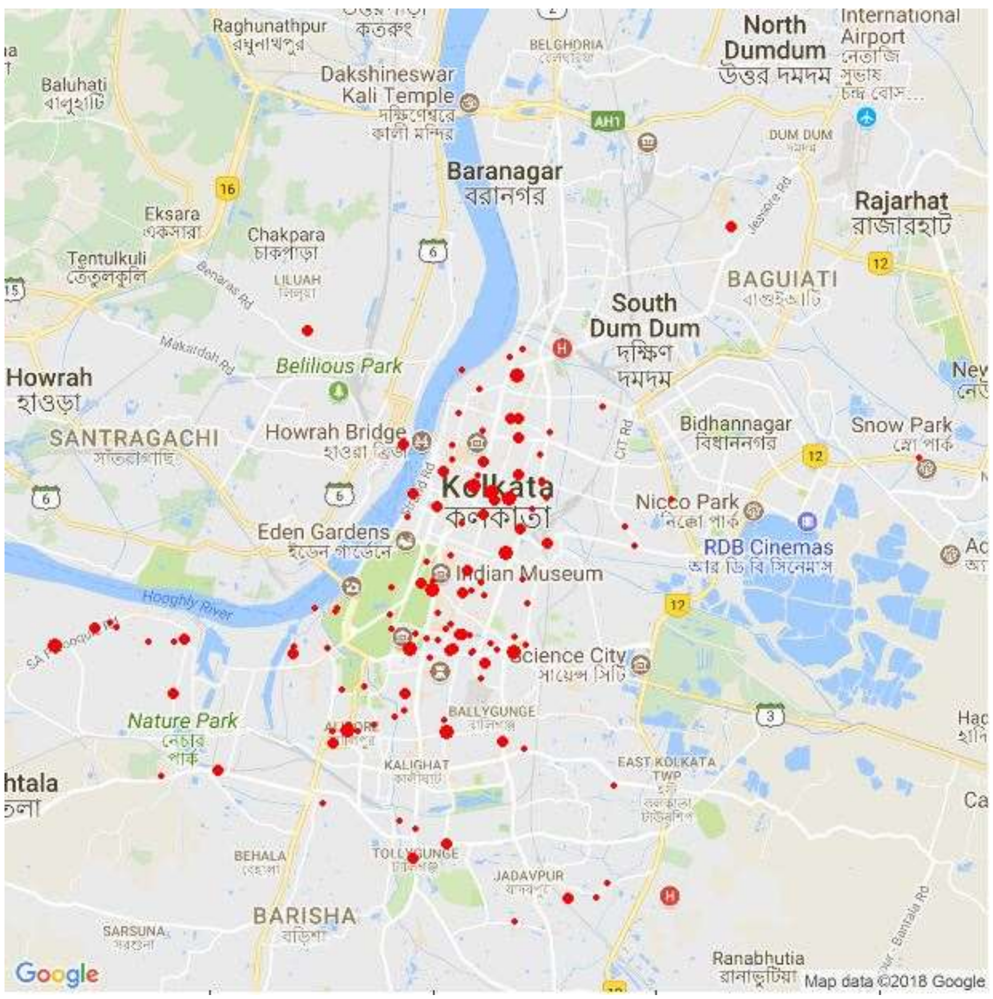

Figure 15 Traffic Location from 01-11-17 to 31-12-17

Concerns

id	message	created_time	sentiment	primary_concern	secondary_concern
1	Not so Happy Diwali ! Couple of days back near Ra...	2017-12-15T22:33:00+0000	negative	other job, rash driving, poor people, corruption r...	Happy Diwali, year old boy, negative stories, cert...
2	I am an advocate of the Hon'ble High Court, Calcut...	2017-09-01T20:02:00+0000	positive	dishonest activities, dishonest professionals	particular Traffic Surgent
3	Salute to for the excellent work & friendly nature...	2017-10-03T20:16:00+0000	positive	excellent work, friendly nature, high pressure sit...	

Figure 16. Database records containing the concerns for user reviews

Concern from reviews is a very important aspect since it gives an overview of what the users are talking about, and their complaints and feedback. Start and end date are given as input and the result of primary and secondary concern is obtained.

Sentiment Analysis

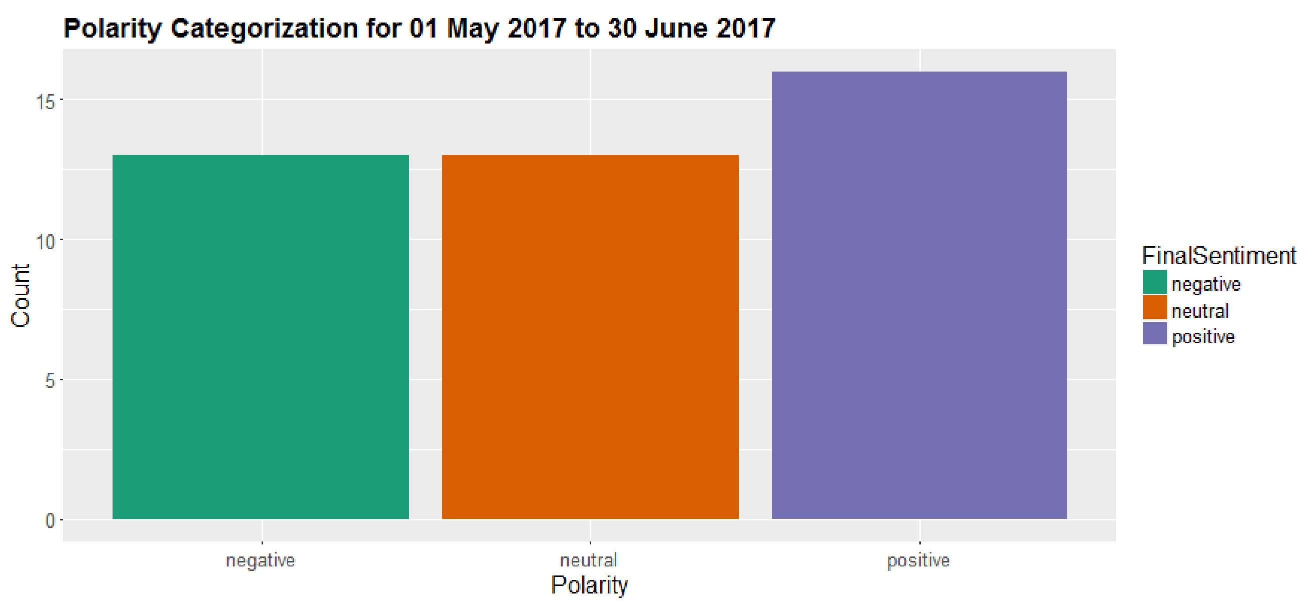

Figure 17. Polarity Graph for Reviews of May'17-Jun'17

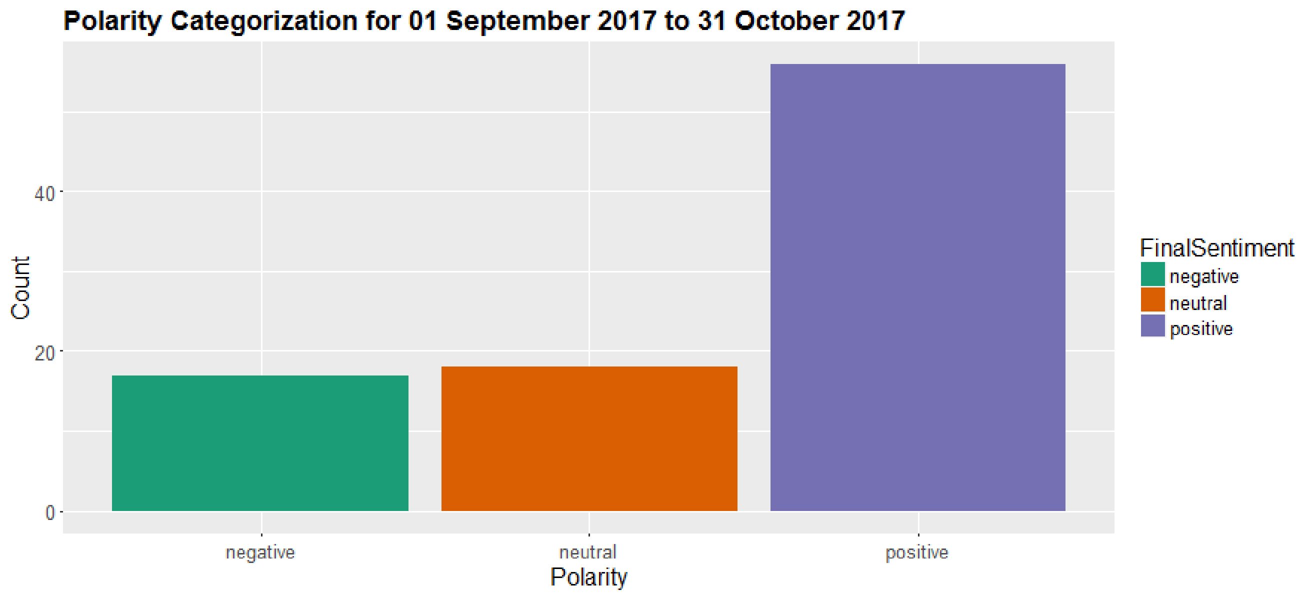

Figure 18. Polarity Graph for Reviews of Sep'17-Oct'17

Sentiment analysis algorithms were applied on the user reviews of the 'Kolkata Traffic Police' page. Polarity categorization and word cloud plots were generated. Polarity categorization was used to classify the emotions of a post as positive, negative or neutral. The visualization helps the department to roughly summarize the performance of the police force during a given period. Word cloud displays an image composed of words where the size of the word reflects the frequency. The word cloud is able to give a quick summary about the kind of 'mood' and 'buzz' among the users. Word cloud was able to tell about the important keywords that the citizens talked about, and this could be either something that concerned them, disturbed them or it could be something that pleased them (Figure 20)

Figure 19 Sentiment Polarity of Reviews from January to April 2018

Figure 20 Word Cloud for Reviews from January - October 2017

The algorithm was applied on bimesters for better results although the program provides the option to enter any date of choice. The positive reviews frequency shows a progressive trend with the frequency of positive reviews increasing, culminating during the period of September-October and then sharply falling back for the next bimester. Positive sentiment for the Police force is highest during the time of September-October due to a number of festivals like Durga Puja, Diwali as seen in Figure 18, 19. The positive reviews are mainly on accounts of excellent traffic and crowd management as the streets are very busy during this time. Many reviews also empathize with the police force for their dedication as these holidays are usually categorized as a family event. Positive sentiments for other months can be generalized to factors like effective traffic control, manual clearing up of water- logged areas, and empathy for police force during extreme weather conditions and individual experiences like car breakdown or lost items.

Figure 17 and Figure 18 delineate the polarity of sentiment expressed in the reviews. The months of May and June show a low sentiment score mainly due to harassment, bad traffic management, corruption and false cases.

Cases of harassment include unprofessional behavior such as public humiliation and the use of profane language. False cases include wrong traffic violation cases imposed on individuals. Complaints about bad traffic management deals with violations committed by buses, auto-rickshaws as well as space encroachment by hawkers and peddlers leading to congested roads. Corruption reviews talk about the bribery and extortions made by some police officers.

CONCLUSION

Local police department play an important role in channeling data from the citizens and broadcasting data to the citizens. Although the advent of social media has made this work simpler in a way, but the workload has also increased due to the staggering amount of information that is generated on social networking sites. The contributions of our work are manifold. The user engagement statistics helped the local police department to draw a relation between the number of posts, average number of post and the engagement of citizens. We were able to conclude that photo and video updates received more attention as compared to status and link updates as iterated in many papers. Also the parameter 'interaction' defined in our work could be correlated to the increase in followers and the percentage of no engagement posts. The graph showing percentage of complaints reflected improvement in Kolkata Traffic Police as the complaints decreased over a period of time. The citizens could use the project to understand the congested areas in Kolkata by using the geo-tagging plot which depicted the aggregation of congested areas. Furthermore the authority could use the geo-tagging plot to verify whether congestion in areas had been resolved over a period of time or not. Finally sentiment analysis was used to understand the emotions of the public, whether they were happy, angry, excited or disappointed of the work Kolkata Police was carrying out.

FUTURE SCOPE

Future scope of the work includes:

- Extraction of reviews in an automated way (web scrapping) rather than extracting it manually
- Analyzing the Visitor Posts for getting insights about complaints and problems
- Improving the algorithms for achieving a better accuracy of determining locations from page owner post and concerns from reviews
- Removal of posts in which the posts are written in English however, the words are a representation of words in vernacular languages, and it is not a valid word in English dictionary.

REFERENCES

1. Number of social media users worldwide from 2010 to 2021 (in billions). https://www.statista.com/statistics/278414/number-of-worldwide-social-network-users. Accessed 15 Nov 2017
2. Ross, C., Orr, E.S., Sisic, M., Arseneault, J.M., Simmering, M.G. and Orr, R.R., 2009. Personality and motivations associated with Facebook use. *Computers in human behavior*, *25*(2), pp.578-586.
3. Amichai-Hamburger, Y. and Vinitzky, G., 2010. Social network use and personality. *Computers in human behavior*, *26*(6), pp.1289-1295.
4. Pak, A. and Paroubek, P., 2010, May. Twitter as a corpus for sentiment analysis and opinion mining. In *LREc* (Vol. 10, No. 2010).
5. Digital population in India as of January 2018 (in millions). https://www.statista.com/statistics/309866/india-digital-population. Accessed 13 May 2018
6. Pang, B. and Lee, L., 2008. Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval*, *2*(1–2), pp.1-135.
7. Kang, J., Tang, L. and Fiore, A.M., 2014. Enhancing consumer–brand relationships on restaurant Facebook fan pages: Maximizing consumer benefits and increasing active participation. *International Journal of Hospitality Management*, *36*, pp.145-155.
8. Mohammed Allama Hossain, Rashika Daga and Saptarsi Goswami, 2017, April. Integration of Classical Features, Social Media Response and Power of Stars to Predict the Box Office Sales. In International Conference on Telecommunication, Power Analysis, and Computing Techniques
9. K. Kim, A. Oglesby-Neal, E. Mohr (2017) 2016 Law Enforcement Use of Social Media Survey. http://www.theiacp.org/Portals/0/documents/pdfs/2016-law-enforcement-use-of-social-media-survey.pdf. Accessed 24 Nov 2017
10. Bakshy, E., Messing, S. and Adamic, L.A., 2015. Exposure to ideologically diverse news and opinion on Facebook. Science, 348(6239), pp.1130-1132.
11. Muralidharan, S., Rasmussen, L., Patterson, D. and Shin, J.H., 2011. Hope for Haiti: An analysis of Facebook and Twitter usage during the earthquake relief efforts. Public Relations Review, 37(2), pp.175-177.
12. Vitak, J., Zube, P., Smock, A., Carr, C.T., Ellison, N. and Lampe, C., 2011. It's complicated: Facebook users' political participation in the 2008 election. CyberPsychology, behavior, and social networking, 14(3), pp.107-114.
13. Bird, D., Ling, M. and Haynes, K., 2012. Flooding Facebook-the use of social media during the Queensland and Victorian floods. *Australian Journal of Emergency Management, The*, *27*(1), p.27.
14. Bertot, J. C., Jaeger, P. T., & Hansen, D. (2012). The impact of polices on government social media usage: Issues, challenges, and recommendations. Government Information Quarterly, 29(1), 30–40, http://dx.doi.org/10.1016/j.giq.2011.04.004.
15. Kavanaugh, A. L., Fox, E. a., Sheetz, S. D., Yang, S., Li, L. T., Shoemaker, D. J., et al. (2012). Social media use by government: From the routine to the critical. Government Information Quarterly, 29(4), 480–491.
16. C. Fugate, *Five Years Later: An Assessment of the Post Katrina Emergency Management Reform Act*, tech. report, US Federal Emergency Management Agency, Oct. 2011.
17. Graham, M., Avery, E.J., & Park, S. (2015). The role of social media in local government crisis communications. Public Relations Review, 41(3), pp.386-394.
18. Mergel, I., 2013. A framework for interpreting social media interactions in the public sector. *Government Information Quarterly*, *30*(4), pp.327-334.
19. Chang, A.M. and Kannan, P.K., 2008. *Leveraging Web 2.0 in government*. Washington, DC: IBM Center for the Business of Government.
20. Picazo-Vela, S., Gutiérrez-Martínez, I. and Luna-Reyes, L.F., 2012. Understanding risks, benefits, and strategic alternatives of social media applications in the public sector. *Government information quarterly*, *29*(4), pp.504-511.

21. Denef, S., Bayerl, P.S., Kaptein, N., 2013. Social Media and the Police – Tweeting Practices of British Police Forces during the August 2011 Riots (pp. 3471-3480)
22. Meijer, A., and Thaens, M., 2013. Social media strategies: Understanding the differences between North American police departments. *Government Information Quarterly*, 30(4), pp.343-350.
23. Mossberger, K., Wu, Y. and Crawford, J., 2013. Connecting citizens and local governments? Social media and interactivity in major US cities. *Government Information Quarterly*, *30*(4), pp.351-358.
24. Hofmann, S., Beverungen, D., Räckers, M. and Becker, J., 2013. What makes local governments' online communications successful? Insights from a multi-method analysis of Facebook. *Government Information Quarterly*, *30*(4), pp.387-396.

APPENDICES

Code Snippet for extracting month intervals

```
getEngagementMonthly<-function(startdate,enddate)
{
  if(as.Date(startdate)>Sys.Date() | as.Date(enddate)>Sys.Date())
  {
    print("Enter a valid date")
    return("NA")
  }
  else
  {
  #mydb = dbConnect(MySQL(), user='root', password='', dbname='ktp', host='127.0.0.1')
  #query<- "select * from page_owner_post"
  #posts<-dbGetQuery(mydb,query)
  posts <- read.csv('C:/Users/DELL/Documents/KTP2/page_owner_posts.csv', stringsAsFactors = FALSE, header
= TRUE)
  df<-data.frame(matrix(ncol = 26, nrow = 0))
  x <- c("startdate",
"enddate","likes","comments","shares","status","photo","link","video","status_likes","photo_likes",
         "link_likes","video_likes","status_comments","photo_comments","link_comments",
         "video_comments","status_shares","photo_shares","link_shares","video_shares",
         "wend_posts","wday_posts","wend_likes","wday_likes","no")
  colnames(df)<-x
  dt=startdate
  flag=0
  nextdate=toString(seq(as.Date(dt), by = "month", length = 2)[2])
  if((as.Date(enddate)-as.Date(nextdate))<0)
  {
    flag=1
  }
  else
  {
    flag=0
  }
  while(flag==0)
  {
    likes=0;comments=0;shares=0;status=0;photo=0; link=0;video=0;status_likes=0;photo_likes=0;
    video_likes=0;link_likes=0;status_comments=0;photo_comments=0;video_comments=0
    link_comments=0;status_shares=0;photo_shares=0;video_shares=0; link_shares=0;wend_posts=0
    wend_likes=0; wday_likes=0;wday_posts=0;no=0
    nextdate=toString(seq(as.Date(dt), by = "month", length = 2)[2]);
```

```
    if((as.Date(enddate)-as.Date(nextdate))==0)
    {
      post<-subset(posts,as.Date(posts$created_time)>=dt & as.Date(posts$created_time)<nextdate)
      for(j in 1:nrow(post))
      {
        no=no+1;
        if(identical(toString(post$type[j]),"status")==TRUE)
        {
          status=status+1
          status_likes=status_likes+as.numeric(post$likes_count[j])
          status_comments=status_comments+as.numeric(post$comments_count[j])
          status_shares=status_comments+as.numeric(post$shares_count[j])
        }
        else if(identical(toString(post$type[j]),"photo")==TRUE)
        {
          photo=photo+1
          photo_likes=photo_likes+as.numeric(post$likes_count[j])
          photo_comments=photo_comments+as.numeric(post$comments_count[j])
          photo_shares=photo_shares+as.numeric(post$shares_count[j])
        }
        else if(identical(toString(post$type[j]),"video")==TRUE)
        {
          video=video+1
          video_likes=video_likes+as.numeric(post$likes_count[j])
          video_comments=video_comments+as.numeric(post$comments_count[j])
          video_shares=video_shares+as.numeric(post$shares_count[j])
        }
        else
        {
          link=link+1
          link_likes=link_likes+as.numeric(post$likes_count[j])
          link_comments=link_comments+as.numeric(post$comments_count[j])
          link_shares=link_shares+as.numeric(post$shares_count[j])
        }
        if(identical(weekdays(as.Date(post$created_time[j])),"Saturday")==TRUE |
identical(weekdays(as.Date(post$created_time[j])),"Saturday")==TRUE)
        {
          wend_posts=wend_posts+1
          wend_likes=wend_likes+as.numeric(post$likes_count[j])
        }
        else
        {
          wday_posts=wday_posts+1
          wday_likes=wday_likes+as.numeric(post$likes_count[j])
```

```
        }
      }
      likes=status_likes+photo_likes+video_likes+link_likes
      comments=status_comments+photo_comments+video_comments+link_comments
      shares=status_shares+photo_shares+video_shares+link_shares
      de<- data.frame(toString(dt),toString(as.Date(nextdate)-
1),likes,comments,shares,status,photo,link,video,status_likes,photo_likes,link_likes,
                  video_likes,status_comments,photo_comments,link_comments,video_comments,
                  status_shares,photo_shares,link_shares,video_shares,wend_posts,wday_posts,
                  wend_likes,wday_likes,no)
      names(de)<-c("startdate",
"enddate","likes","comments","shares","status","photo","link","video","status_likes","photo_likes",
                "link_likes","video_likes","status_comments","photo_comments","link_comments",
                "video_comments","status_shares","photo_shares","link_shares","video_shares",
                "wend_posts","wday_posts","wend_likes","wday_likes","no")
      df=rbind(df,de)
      #print(dt); print("-"); print(as.Date(nextdate)-1)
      flag=1
      break;
    }
    if((as.Date(enddate)-as.Date(nextdate))<0)
    {
      flag=1
      break
    }
    if((as.Date(enddate)-as.Date(nextdate))>0 & flag==0)
    {
      post<-subset(posts,as.Date(posts$created_time)>=dt & as.Date(posts$created_time)<nextdate)
      #Compute the counts as shown above
      de<- data.frame(toString(dt),toString(as.Date(nextdate)-
1),likes,comments,shares,status,photo,link,video,status_likes,photo_likes,link_likes,
                  video_likes,status_comments,photo_comments,link_comments,video_comments,
                  status_shares,photo_shares,link_shares,video_shares,wend_posts,wday_posts,
                  wend_likes,wday_likes,no)
      names(de)<-c("startdate",
"enddate","likes","comments","shares","status","photo","link","video","status_likes","photo_likes",
                "link_likes","video_likes","status_comments","photo_comments","link_comments",
                "video_comments","status_shares","photo_shares","link_shares","video_shares",
                "wend_posts","wday_posts","wend_likes","wday_likes","no")
      df=rbind(df,de)
      #print(dt); print("-"); print(as.Date(nextdate)-1)
    }

      dt=nextdate;
```

```
  }
  if(flag==1)
  {
    likes=0;comments=0;shares=0;status=0;photo=0; link=0;video=0;status_likes=0;photo_likes=0;
    video_likes=0;link_likes=0;status_comments=0;photo_comments=0;video_comments=0
    link_comments=0;status_shares=0;photo_shares=0;video_shares=0; link_shares=0;wend_posts=0
    wend_likes=0; wday_likes=0;wday_posts=0;no=0

    if((as.Date(enddate)-as.Date(nextdate))==0)
    {

      post<-subset(posts,as.Date(posts$created_time)==nextdate)
      #Compute the counts as shown above
      de<-
data.frame(toString(enddate),"NA",likes,comments,shares,status,photo,link,video,status_likes,photo_likes,link_li
kes,
                    video_likes,status_comments,photo_comments,link_comments,video_comments,
                    status_shares,photo_shares,link_shares,video_shares,wend_posts,wday_posts,
                    wend_likes,wday_likes,no)
      names(de)<-c("startdate",
"enddate","likes","comments","shares","status","photo","link","video","status_likes","photo_likes",
                  "link_likes","video_likes","status_comments","photo_comments","link_comments",
                  "video_comments","status_shares","photo_shares","link_shares","video_shares",
                  "wend_posts","wday_posts","wend_likes","wday_likes","no")
      df=rbind(df,de)
      #print(enddate); print("-"); print("NA")
    }
    else
    {
      post<-subset(posts,as.Date(posts$created_time)>=as.Date(dt) &
as.Date(posts$created_time)<=as.Date(enddate))
      #Compute the counts as shown above
      de<-
data.frame(toString(dt),toString(enddate),likes,comments,shares,status,photo,link,video,status_likes,photo_likes,li
nk_likes,
                    video_likes,status_comments,photo_comments,link_comments,video_comments,
                    status_shares,photo_shares,link_shares,video_shares,wend_posts,wday_posts,
                    wend_likes,wday_likes,no)
      names(de)<-c("startdate",
"enddate","likes","comments","shares","status","photo","link","video","status_likes","photo_likes",
                  "link_likes","video_likes","status_comments","photo_comments","link_comments",
                  "video_comments","status_shares","photo_shares","link_shares","video_shares",
                  "wend_posts","wday_posts","wend_likes","wday_likes","no")
      df=rbind(df,de)
```

```
      #print(dt); print("-"); print(enddate)
    }
  }
  #write.csv(df,'C:/Users/DELL/Desktop/csvs/month_engagement.csv',row.names=FALSE)
  #dbDisconnect(mydb)
  return(df)
  }
}
```

Code Snippet for plotting Graph

```
#Plots the graph showing the type of post in a given interval
library(ggplot2)
graphical_df <- data.frame(matrix(ncol = 3, nrow = 0))
colnames(graphical_df) <- c("date", "freq", "type")
k <- 1
for(i in 1:nrow(df))
{
  for(j in 1:4)
  {
    date_range <- paste(df[i, 1],df[i, 2],sep="\n")
    graphical_df[k, 1] <- date_range
    graphical_df[k, 2] <- df[i, 5+j]
    graphical_df[k, 3] <- type[j]
    k <- k + 1
  }
}

type = c("Status", "Photo", "Link", "Video")

ggplot(graphical_df, aes(x=date, y = freq, fill = type)) +
  geom_bar(stat="identity")+
 labs(x = "Time Interval", y = "Frequency") +
  ggtitle("Type of Posts") +
  theme(axis.title.x = element_text(), axis.title.y = element_text(),
        plot.title = element_text(size = 10, face = "bold")) +
  guides(color=guide_legend("Type:"))
ggsave("Typepost.jpeg", width = 9, height = 3.5, dpi = 720)
```

Code for extracting location from a textual update

```
source('C:/Users/DELL/Documents/FYP/location.R')
getLocation<- function(msg){
  locs<-read.csv('C:/Users/DELL/Desktop/csvs/locations.csv',header= TRUE, stringsAsFactors = FALSE)
  msg<- gsub('[[:punct:] ]+',' ',msg)
  s<-sapply(msg, tolower)
  msg<-removeWords(toString(s),stopwords('english'))
```

```
  df<-data.frame(matrix(ncol = 3, nrow = 0))
  x <- c("location","latitude","longitude")
  colnames(df)<-x
  msg<-gsub("\\s+"," ",msg)
  l=""
  s=""
  wrd2=""
  wrd3=""
  wrd4=""
  len=vapply(strsplit(msg, "\\W+"), length, integer(1))
  words_initial<-c("traffic", "update", "alert")
  word_loc<-c("road", "street", "crossing", "sarani", "ghat", "flyover", "block", "lane", "avenue", "bazar",
"bazaar", "stadium","rd","st")
  if(((sapply(words_initial, grepl, msg)[1]=="TRUE")) & ((sapply(words_initial, grepl, msg)[2]=="TRUE") |
(sapply(words_initial, grepl, msg)[3]=="TRUE")))
  {
    if(sapply("school", grepl, msg)[1]=="TRUE")
    {
      flag=0
      de<-data.frame("NA","NA","NA")
      names(de)<- c("location","latitude","longitude")
      df<-rbind(df,de)
    }
    else
    {
    for(i in 1:nrow(locs))
    {
      if(length(grep(toString(locs$location[i]),msg))==1)
      {
        de<-data.frame(toString(locs$location[i]),locs$latitude[i],locs$longitude[i])
        names(de)<- c("location","latitude","longitude")
        df<-rbind(df,de)
      }
    }

      i=1;
      while(i<=len)
      {
      str= word(msg,i)
      if((identical(str,toString(word_loc[[1]]))==TRUE) | (identical(str,toString(word_loc[[2]]))==TRUE) |
         (identical(str,toString(word_loc[[3]]))==TRUE) | (identical(str,toString(word_loc[[4]]))==TRUE) |
         (identical(str,toString(word_loc[[5]]))==TRUE) | (identical(str,toString(word_loc[[6]]))==TRUE) |
         (identical(str,toString(word_loc[[7]]))==TRUE) | (identical(str,toString(word_loc[[8]]))==TRUE) |
         (identical(str,toString(word_loc[[9]]))==TRUE) | (identical(str,toString(word_loc[[10]]))==TRUE) |
```

```
 (identical(str,toString(word_loc[[11]]))==TRUE) |(identical(str,toString(word_loc[[12]]))==TRUE) |
 (identical(str,toString(word_loc[[13]]))==TRUE) |(identical(str,toString(word_loc[[14]]))==TRUE))
{
 if((identical(str,toString(word_loc[[1]]))==TRUE) | (identical(str,toString(word_loc[[2]]))==TRUE) |
   (identical(str,toString(word_loc[[4]]))==TRUE) |
   (identical(str,toString(word_loc[[5]]))==TRUE) | (identical(str,toString(word_loc[[6]]))==TRUE) |
   (identical(str,toString(word_loc[[7]]))==TRUE) | (identical(str,toString(word_loc[[8]]))==TRUE) |
   (identical(str,toString(word_loc[[9]]))==TRUE) | (identical(str,toString(word_loc[[10]]))==TRUE) |
   (identical(str,toString(word_loc[[11]]))==TRUE) |(identical(str,toString(word_loc[[12]]))==TRUE) |
   (identical(str,toString(word_loc[[13]]))==TRUE) |(identical(str,toString(word_loc[[14]]))==TRUE))
 {
 str1=word(msg,i+1)
 if((identical(str1,toString(word_loc[[3]]))==TRUE))
 {
  if(i>=2)
   st2= word(msg,i-1)
  else
   st2="NA"
  if(i>=3)
   st3=word(msg,i-2,i-1)
  else
   st3="NA"
  if(i>=4)
   st4=word(msg,i-3,i-1)
  else
   st4="NA"
  str=paste(str,str1,sep=" ")
  i=i+1
 }
 else
 {
  if(i>=2)
   st2=word(msg,i-1)
  else
   st2="NA"
  if(i>=3)
   st3=word(msg,i-2,i-1)
  else
   st4="NA"
  if(i>=4)
   st4=word(msg,i-3,i-1)
  else
   st4="NA"
 }
```

```
}
else
{
  if(i>=2)
    st2=word(msg,i-1)
  else
    st2="NA"
  if(i>=3)
    st3=word(msg,i-2,i-1)
  else
    st3="NA"
  if(i>=4)
    st4=word(msg,i-3,i-1)
  else
    st4="NA"
}
if(identical(toString(st2),"NA")==FALSE)
{
st2<- removeWords(toString(st2),words_initial)
st2<-removeWords(toString(st2),word_loc)
st2<-gsub("\\s+"," ",st2)
if(identical(st2,"")==FALSE && identical(st2," ")==FALSE)
  wrd2=paste(st2,str,sep=" ")
else
  wrd2=""
}
else
{
  wrd2=""
}
if(identical(toString(st3),"NA")==FALSE)
{
  st3<- removeWords(toString(st3),words_initial)
  st3<-removeWords(toString(st3),word_loc)
  st3<-gsub("\\s+"," ",st3)
  if(identical(st3,"")==FALSE && identical(st3," ")==FALSE)
    wrd3=paste(st3,str,sep=" ")
  else
    wrd3=""
}
else
{
  wrd3=""
}
```

```
if(identical(toString(st4),"NA")==FALSE)
{
  st4<- removeWords(toString(st4),words_initial)
  st4<-removeWords(toString(st4),word_loc)
  st4<-gsub("\\s+"," ",st4)
  if(identical(st4,"")==FALSE && identical(st4," ")==FALSE)
    wrd4=paste(st4,str,sep=" ")
  else wrd4=""
}
else
{
  wrd4=""
}
flag=0
for(j in 1:nrow(locs))
{
  if(identical(toString(locs$location[j]),wrd2)==TRUE | identical(toString(locs$location[j]),wrd2)==TRUE |
identical(toString(locs$location[j]),wrd2)==TRUE)
  {
    flag=1
    de<-data.frame(toString(locs$location[j]),locs$latitude[j],locs$longitude[j])
    names(de)<- c("location","latitude","longitude")
    df<-rbind(df,de)
  }
}
  if(flag==0)
  {
    k=1;
    while(k<5 && flag==0 && (identical(wrd2,"")==FALSE) && (identical(wrd2,"NA")==FALSE))
    {
      s<-locationFunction(wrd2)
      if(identical(toString(s$lat),"NA")==TRUE)
      {
        k=k+1
        Sys.sleep(7)
      }
      else
      {
        l=wrd2;
        de<-data.frame(toString(wrd2),s$lat,s$lon)
        names(de)<- c("location","latitude","longitude")
        df<-rbind(df,de)
        locs<-rbind(locs,de)
        k=8;
```

```
        flag=1;
      }
    }
    k=1;
    while(k<5 && flag==0 && (identical(wrd3,"")==FALSE) && (identical(wrd3,"NA")==FALSE))
    {
      s<-locationFunction(wrd3)
      if(identical(toString(s$lat),"NA")==TRUE)
      {
        k=k+1
        Sys.sleep(7)
      }
      else
      {
        l=wrd3;
        de<-data.frame(toString(wrd3),s$lat,s$lon)
        names(de)<- c("location","latitude","longitude")
        df<-rbind(df,de)
        k=8;
        flag=1;
      }
    }
    k=1;
    while(k<5 && flag==0 && (identical(wrd4,"")==FALSE) && (identical(wrd4,"NA")==FALSE))
    {
      s<-locationFunction(wrd4)
      if(identical(toString(s$lat),"NA")==TRUE)
      {
        k=k+1
        Sys.sleep(7)
      }
      else
      {
        l=wrd4;
        de<-data.frame(wrd4,s$lat,s$lon)
        names(de)<- c("location","latitude","longitude")
        df<-rbind(df,de)
        k=8;
        flag=1;
      }
    }
  }
}
i=i+1;
```

```
     }
     if(flag==1)
     {
       df<-df[!duplicated(df$location),]
       locs<-rbind(locs,df)
       locs<-locs[!duplicated(locs$location),]
       write.csv(locs,'C:/Users/DELL/Desktop/csvs/locations.csv',row.names= FALSE)
     }
   }

  }
  else
  {
   de<-data.frame("NA","NA","NA")
   names(de)<- c("location","latitude","longitude")
   df=rbind(df,de)
  }
  return(df)
}
```

Code for getting primary and secondary concern from review message

```
get_concerns <- function(message)
{
  # Reading the .csv files
  dictionary <- read.csv("C:/Users/DELL/Desktop/csvs/generated_dictionary.csv",
                         header = TRUE, stringsAsFactors = FALSE)

  # Creating a new dataframe to store the final result
  review_result <- data.frame(concerns = character(), primary_concerns = character(),
                              secondary_concerns = character(), stringsAsFactors = FALSE)

  # Running a for loop to process every row/message of the file.
  for(i in 1:1)
  {
    strng <- message
    x <- NLP::as.String(strng)

    # Before POS tagging, we need to do Sentence annotation followed by word annotation
    wordAnnotation <- NLP::annotate(x, list(Maxent_Sent_Token_Annotator(),
Maxent_Word_Token_Annotator()))

    # POS tag the words & extract the "words" from the output
    POSAnnotation <- NLP::annotate(x, Maxent_POS_Tag_Annotator(), wordAnnotation)
    POSwords <- subset(POSAnnotation, type == "word")

    # Extract the tags from the words
    tags <- sapply(POSwords$features, '[[', "POS")
```

```
# Create a data frame with words and tags
tokenizedAndTagged <- data.frame(Tokens = as.character(x[POSwords]), Tags = tags, stringsAsFactors = 
FALSE)

# Define a flag(tags_mod) for pos tags - Flag set to 1 if it contains the POS tag we are interested in else 0
# In this case we only want Noun and Adjective tags (NN, JJ)
# Note that this will also capture variations such as NNP, NNPS etc
tokenizedAndTagged$Tags_mod = grepl("NN|JJ", tokenizedAndTagged$Tags)

# Initialize a vector to store chunk indexes
chunk = vector()

# Iterate through each word and assign each one to a group
# if the word doesn't belong to NN|JJ tags (i.e. tags_mod flag is 0) assign it to the default group (0)
# If the ith tag is in "NN|JJ" (i.e. tags_mod flag is 1) assign it to group i-1 if the (i-1)th tag_mod
# flag is also 1; else assign it to a new group
chunk[1] = as.numeric(tokenizedAndTagged$Tags_mod[1])
if(nrow(tokenizedAndTagged) > 1)
{
  for (j in 2:nrow(tokenizedAndTagged)) {

    if(!tokenizedAndTagged$Tags_mod[j]) {
      chunk[j] = 0
    } else if (tokenizedAndTagged$Tags_mod[j] == tokenizedAndTagged$Tags_mod[j-1]) {
      chunk[j] = chunk[j-1]
    } else {
      chunk[j] = max(chunk) + 1
    }

  }
}

# Split and chunk words
text_chunk <- split(as.character(tokenizedAndTagged$Tokens), chunk)
tag_pattern <- split(as.character(tokenizedAndTagged$Tags), chunk)
names(text_chunk) <- sapply(tag_pattern, function(x) paste(x, collapse = "-"))

# Extract chunks matching pattern
res = text_chunk[grepl("JJ-NN", names(text_chunk))]
res = sapply(res, function(x) paste(x, collapse =  " "))
res = sapply(res, function(x) paste(x))

# Initialize the variables. k to iterate the temporary primary result dataframe - rev_result.
# n to iterate the temporary secondary result dataframe - rev_result_secondary.
k <- 1
n <- 1
rev_result <- data.frame(concerns = character(), stringsAsFactors = FALSE)
rev_result_secondary <- data.frame(concerns = character(), stringsAsFactors = FALSE)

# If res contains values, then concerns have been obtained, otherwise there were no significant concerns
# available
```

```
if(length(res))
{
  review_result[i, 1] <- as.character("Obtained")

  # Iterates through every key phrase obtained and decides whether it is a primary concern or secondary concern
  # Also checks whether the phrase has not already been stored to avoid redundancy
  for(l in 1:length(res))
  {
    var1 <- word(res[[l]], start = -1, sep = fixed(" "))

    # Concerns are categorized into primary if the NN (noun) is present in the domain dictionary.
    if(match(var1, dictionary$topic, nomatch = 0) & match(res[[l]], rev_result$concerns, nomatch = 0) == 0)
    {
      rev_result[k, 1] <- res[[l]]
      k <- k + 1
    } else if(match(res[[l]], rev_result_secondary$concerns, nomatch = 0) == 0) {
      rev_result_secondary[n, 1] <- as.character(res[[l]])
      n <- n + 1
    }
  }
} else {
  review_result[i, 1] <- as.character("No significant concern")
}

# Stores the message as it is into the final result dataframe - review_result
#review_result[i, 1] <- message

# If rev_result or rev_result_secondary contains values then store them in review_result else leave blank
if(nrow(rev_result) | nrow(rev_result_secondary))
{

  # Store only the first concern from primary and secondary concern list into the review_result to avoid NA
  # while using paste function later
  if(nrow(rev_result))
    review_result[i, 2] <- rev_result[1, 1]
  else
    review_result[i, 2] <- as.character("")
  if(nrow(rev_result_secondary))
    review_result[i, 3] <- rev_result_secondary[1, 1]
  else
    review_result[i, 3] <- as.character("")

  # Initialize the variables. m to iterate the rev_result and o to iterate the rev_result_secondary
  m <- 2
  o <- 2
  flag <- 0

  # Keep copying the concerns from primary and secondary list to the review_result until the last.
  # Once both the list are complete, flag is set 1 and the loop terminates
  # paste function is used to place all the concerns side by side in review_result
  while((nrow(rev_result)>1 | nrow(rev_result_secondary)>1) & flag == 0)
```

```
      {
        if(m<=nrow(rev_result))
        {
          review_result[i, 2] <- paste(review_result[i, 2], rev_result[m, 1], sep = ", ")
          m <- m + 1
        } else if(o<=nrow(rev_result_secondary)) {
          review_result[i, 3] <- paste(review_result[i, 3], rev_result_secondary[o, 1], sep = ", ")
          o <- o + 1
        } else {
          flag <- 1
        }
      }
    } else {
      if(nrow(rev_result)==0)
        review_result[i, 2] <- as.character("")
      if(nrow(rev_result_secondary)==0)
        review_result[i, 3] <- as.character("")
    }

  }

  return(review_result)
}
```