iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 7 **Issue:** XI **Month of publication:** November 2019

DOI: http://doi.org/10.22214/ijraset.2019.11010

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.177
Volume 7 Issue XI, Nov 2019- Available at www.ijraset.com

Summaration Generating Timeline for Evolutionary Tweet Streams and Categorizing Tweets

Pediredla Pratyusha[1], Kunjam Nageswara Rao[2], Sitaratnam Gokuruboyina[3]
[1]*M.Tech,Department of Computer Science and System Engineering, Andhra University College of Engineering ,Andhra university(A), Visakhapatnam, India*
[2]*Professor, Department of Computer Science and System Engineering, Andhra University College of Engineering, Andhra university(A) & Adjunct Professor, IBCB Visakhapatnam, India*
[3] *Associate professor, Department of Computer Science and Engineering, Lendi Institute of Engineering and Technology, Vizianagaram, India*

Abstract: ***Now a day's social networking sites are the fastest medium which delivers news to the user as compared to the newspaper and television. There so many social networking sites are present and one of them is Twitter. People run-out of time and even then, they find a small time to know what's happening all over the world. So, in such hefty schedule people can't go through all the tweets which people post all day long and the model proposed is summarizing the tweets and generating timeline for evolutionary tweet streams. It consists three components, first tweet stream clustering for clustering tweets using Bisect k-means cluster algorithm, so that the tweets which are taken from the twitter Application Programming Interface (API) are pre-processed and then cluster. Second component tweet summarization cluster vector technique for generating rank summarization using LexRank algorithm, third component is to categorize tweets into positive, negative and neutral tweets using sentiment analysis where tweets are collected by using hash tags (screen names) of twitter individual account so that it will display the positive, negative ,neutral tweets of that individual account. Doing categorization tweets for the particular user then it will display the positive tweets percentage:22.2%, negative tweets percentage 11.11%, neutral tweets of 66.6% and the results are plotted on bar and pie chat.***

Keywords: Tweet stream, tweet clustering, summary, timeline, categorizing.

I. INTRODUCTION

This social platform is very convenient to use that's why the celebrities, corporations, and organizations also create their own social pages to interact with their fans and the public. To express their opinions on each message users are giving a like and leaving a comment on it or forwarding it. Due to this the numbers of comment are increasing rapidly and generation rate is remarkably high. Consequently, users unnecessarily must undergo the whole remark listing of each message and it is nearly impossible on every occasion. But still users understand to know what other peoples are talking about and what the opinions out of these discussions. A summary is normally generated with major categories of strategies, called
extraction and abstraction. Extractive involves finding relevant sentences that belong to the summary. Abstractive summarization involves identifying or paraphrasing sections of the content material to be summarized.

A. About Lex Rank

Lex rank is an unsupervised approach to text summarization based on graph-based centrality scoring of sentences. The main idea is that sentences "recommend" other similar sentences to the reader. Thus, if one sentence is very similar to many others, it will likely be a sentence of great importance. The importance of this sentence also stems from the importance of the sentences "recommending" it. Thus, to get ranked highly and placed in a summary, a sentence must be similar to many sentences that are in turn also similar to many other sentences. This makes intuitive sense and allows the algorithms to be applied to any arbitrary new text.

B. Evolutionary Tweet Streams

People tweet about a particular tweet posted by another person, again another person posts tweets related to the second person and again third person tweets related to the second person, this process goes on. The tweets generated in this way are known as evolutionary tweet streams.

C. Summarization and Timeline Generation

Summarization is done using sumblr framework which uses LexRank as a part of it. LexRank is a modified page rank algorithm which uses graph-based summarization technique where sentences recommend sentences and more similar sentences are considered important which will be the summary.The core of the timeline generation module is a topic evolution detection algorithm which produces real- time and range timelines in a similar way. We shall only describe the real-time case here. The algorithm discovers sub-topic changes by monitoring quantified variations during the course of stream processing. A large variation at a particular moment implies a sub-topic change, which is a new node on the timeline.

II. RELATED WORK

Tweet summarization consists of two steps. First step requires tweet data clustering and then second actually summarization is performed. Introduce a summarization framework called Sumblr. Sumblr is the continuous summarization by stream clustering. This is the first which studied continuous tweet stream summarization. This framework consists of three main components, namely the Tweet Stream Clustering module, the High-level Summarization module and the Timeline Generation module. Sumblr is useful to work on dynamic, fast arriving, and large-scale tweet streams [1]. creators expect to make condensations of tweets from live drifting likewise continuous themes. The fundamental objective is to gather the tweets by criticalness or convenience so that an end client can be given a sensible think of the most imperative substance from the Twitter stream. Summarization is refined using a non-parametric Bayesian model associated with Hidden Markov Models and a novel perception display expected to allow positioning base [2]. Authors presented a new application, namely sequential summarization for Twitter trending topics. The two proposed systems recognize the subtopics and additionally extricate huge tweets to make sub-rundowns. The assessments to the extent the three estimations, including extension, interest and relationship and also the human assessment all demonstrate that the stream/semantic combination ST+SE-PA philosophy is the best decision among all the proposed approaches [3].creators address the troubles of outlining calculation to gathering bearing stream upon the sliding window show, including variable reviewing rate, data insecurity, obliged resources, propelling property, and the effect of the outdated tuples. In perspective of such issues, they have proposed a system for trajectory stream clustering, including three sections, the information pre- processing part, the online part that separating summary statistics of trajectory stream segment over sliding window, and the offline part that re-clustering micro-clusters based on such statistical information. In particular, cluster features can be kept up viably when new trajectory line segments consistently come in, though the impact of the lapsed records can be expelled securely to keep away from performance degradation with negligible damage to result quality [4]. creators have given arrangement on a sensible issue of stream mining with activity recognition. The strategy unites dynamic and also incremental learning procedure for perceiving quantities of exercises. They additionally join directed, unsupervised and dynamic figuring out how to gather a healthy and compelling acknowledgment structure. Past methodologies for stream classification did not address this crucial issue. Authors tried given procedure on genuine datasets and talked about the framework performance contrasted with other classification systems. The cultural identification and Colour continue plays a significant role in society [5]. Author aimed on consolidating known facts related to cultural responses to colours by data-mining social media. To separate the utilization of 11 fundamental colour terms in Japanese and German Twitter sustains, word clusters and co-occurrences are analyzed [6]. CLU Stream is one of the most typical stream clustering methods. It is having online micro- clustering component and also offline macro clustering component. Online micro clustering component require efficient process to store summaries. Offline components use only summary statistics. The pyramidal time frame was also proposed by authors to recall historical micro clusters for different time durations [5]. We have studied the paper “A framework for clustering evolving data streams” (C. C. Aggarwal, Johan, J. Wang, and P. S. Yu) in which TCVs are considered as potential sub-topic; for stream clustering, Cluster stream method is used. It includes online and offline micro clustering component. For recalling historical micro cluster, pyramidal time frame also proposed for random time duration [1]. For using function Lex rank in TCV rank algorithm we have studied: “Lex Rank: Graph based lexical centrality as salience in text summarization**”** (G. Erkan and D. Ramdev) in this paper Lex ranking is calculated. Depending on the similar data graph is created; Lex rank is used for finding top ranked tweets among large data set.

Also we referred, “Text stream clustering based on adaptive feature selection” (L. Gong, J. Zeng, and Shang) worked on a various service on the Web such as news filtering, text crawling, etc. It mainly focuses on topic detection and tracking (TDT). Clustering is used for analysing text stream[2].

Again we have studied paper “Evolutionary timelines Summarization A balanced optimization framework via iterative substitution” (R. Yan, X. Wan, J. Otterbacher, L.Kong, X. Li, and Y.Zhang) evolutionary timeline summarization which consist of time stamped summaries which is used to generate timeline dynamically during the process of continuous summarization (Sumblr) [3].

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.177
Volume 7 Issue XI, Nov 2019- Available at www.ijraset.com

III. EXISTING SYSTEM

The existing system is based on SUMBLR frame work (continuous summarization by stream cluster ring). tweets are summarized using TCV rank summarization algorithm and tweets are formed into clusters and that tweets are categorized based on evolution.

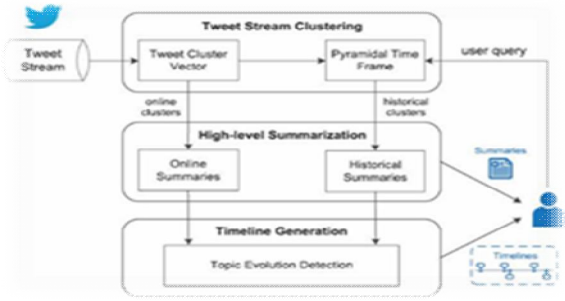


Figure1: sumblr frame work

A. Tweet Stream Clustering

There are two types of data online and offline. This module maintains the online statistical data. It efficiently clusters data based on topic and maintain compact cluster information.

B. High-Level Summarization

There are two types of summaries in this module: online and historical summaries. What is currently discussed among the public is described in online summary and the historical summary helps people to understand the main happenings during a specific period.

C. Timeline Generation

The core of the timeline generation module is a topic evolution detection algorithm which produces real-time and range timelines in a similar way. The algorithm discovers sub-topic changes by monitoring quantified variations during the course of stream processing. A large variation at a particular moment implies a sub- topic change Which is a new node on the timeline. The main process is described in Algorithm 3. First the tweets are binned by time (e.g., by day) as the stream proceeds. This sequenced binning is used as input of the algorithm.

IV. PROPOSED SYSTEM

The tweets streams are collected directly from the twitter API and summarized using Lex rank algorithm which has best efficiency and flexibility compared to other summarization algorithm and similar tweets are formed into clusters. Clustering the tweets based on the similarity and showing the graph. The cosine similarity is calculated and based on the scores generated by cosine similarity; then the cluster graph is generated.Also, categorize tweets into positive, negative and neutral tweets using sentiment analysis where tweets are collected by using hash tags (screen names) of twitter individual account.

V. METHODOLOGY

Using twitter API (APPLICATION PROGRAMING INTERFACE) to collect tweets based on search strings And then collect tweets according to hashtags of individual twitter account. Generatings timeline for particular tweets.After collecting tweets, that tweets are saved to datasets(.csv) files. Pre-processing the tweets i.e., removing stop words the tweets which are collected from streaming API. Clustering the tweets based on the similarity and showing the graph. Summarize the pre-processing data using Lex Rank. Doing categorization tweets by using sentimental analysis (positive, negative, neutral) for the particular user. Showing bar and pie graphs of positive, negative, neutral tweets of the particular user.

1) *Pre-processing the Tweet Data:* Pre-processing the raw tweets is important before we cluster the tweet data. The first step of pre-processing is to remove stop words. Consider a tweet t, at first, we delimited the tweet by special characters, to get the words array way. Stop words collection c is available in wake jar. So, for each word win way, if w is present in c then remove w from way. By this way we can remove stop words from t. Although stemming is an essential step in pre-processing as per natural language processing. In our scenario, we should not stem our tweet data because tweets contain words which are not found in regular English language but their correctness is important for our cluster formation and summarization steps. For example, tweets contain the word “iPhone” when they are filtered on the topic, say, “Apple”. If this tweet is stemmed after the removal of stop words, the word “iPhone” results in “iPhone”, which might be a proper result after stemming but the same has

to be used for clustering and summary generation. This may lead to some meaningless but popular words appearing in our summary.

2) *Tweet Stream Clustering:* In the tweet stream clustering, an efficient tweet stream clustering algorithm called k-means algorithm is designed which is an online algorithm for effective clustering of tweets. The algorithm employs data structures to keep important tweet information in clusters. The Cosine similarity is calculated and based on the scores generated by the cosine similarity; the clusters are generated.

A. Algorithms

1) Clustering Algorithm

a) Input: Cluster set.

b) Output: Assigning cluster for new tweets.

i) Step I- Collection of new tweet stream.

ii) Step II- Depending upon two attribute it creates new cluster

a. MaxSim (maximum similarity)

b. MBS (minimum boundary similarity)

iii) Step III- If is less than then it creates new cluster.

iv) Step IV-otherwise update new cluster.

2) Summarization Algorithm

a) Input: Cluster Set

b) Output: Summarization according to rank

i) Step I- Building similarity graph for all tweet.

ii) Step II-Computing Lex Rank to know which tweets are top ranked

iii) Step III-Adding tweets into summary according to equation

$$t = \underset{t_i}{argmax} \left[\lambda \frac{n_{t_i}}{n_{max}} LR(t_i) - (1-\lambda) \underset{t_i \epsilon S}{avg\ Sim}(t_i, t_j)\right]$$

iv) Step IV- Checking summary length till it reached to max size.

$$(t_i \epsilon\ T - s)$$

v) Selecting tweet globally based on above equation.

3) Topic Evolution Detection Algorithm

a) Input: A tweet stream binned by time units.

b) Output: A timeline node set.

i) Step I: Binning tweets by time.

ii) Step II: Appending new timeline nodes whenever large variation detected.

By using

```
While!stream.end( )do
Bin ci=stream.next( )
```

VI. RESULTS AND DISCUSSION

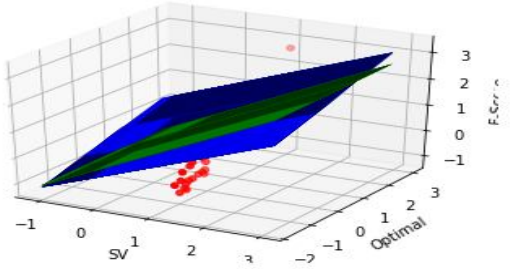
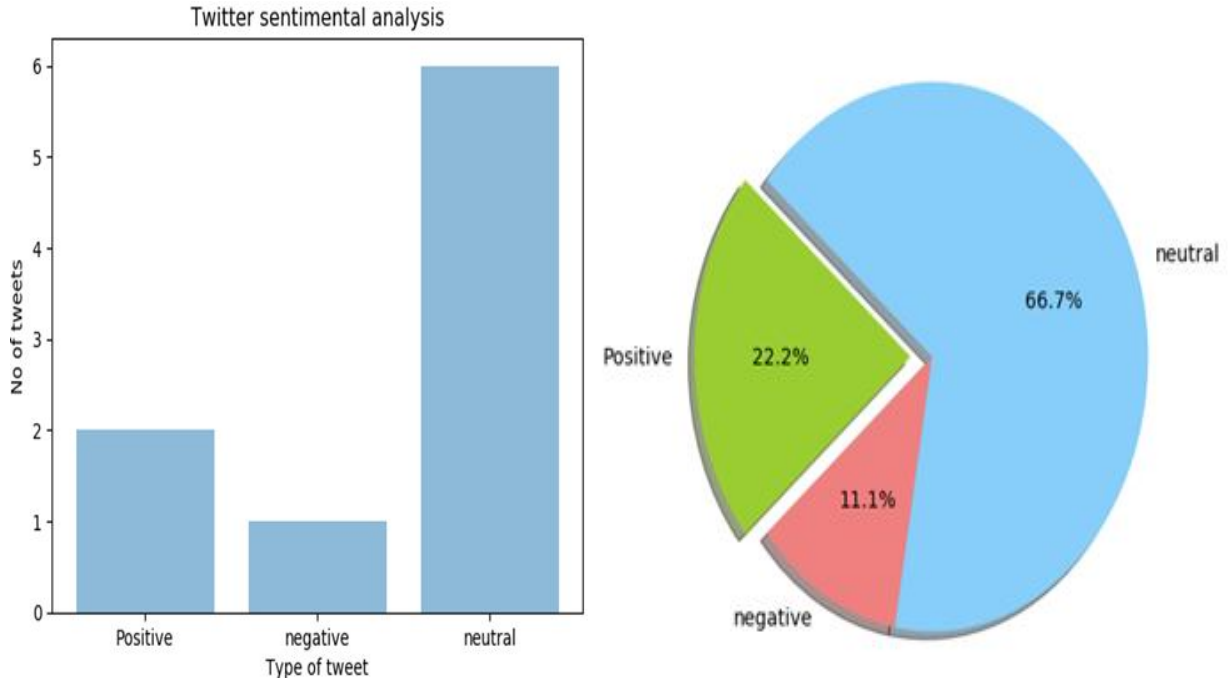


Figure 6.1: Tweets stream clustering

Pre-processing the tweets i.e., removing stop words the tweets which are collected from streaming Application Programming Interface (API) and then clustering the tweets based on the similarity and showing the graph. Clustering the tweets based on the similarity and showing the graph. The cosine similarity is calculated and based on the scores generated by cosine similarity; then the cluster graph is generated. If a user wants to keep track of an event / news-story, it is difficult for him to have to read all the tweets containing identical or redundant information and techniques to summarize large number of tweets using graph-based approach for summarizing tweets, where a graph is first constructed considering the similarity among tweets of two users, and community detection techniques are then used on the graph to cluster similar tweets.

Finally, a representative tweet is chosen from each cluster to be included into the summary. Which help to capture the semantic similarity among tweets.



(a) bar graph of categorization (b) pie chart of categorization

Figure 6.2: categorizing tweets using sentimental analysis

Doing categorization tweets by using sentimental analysis (positive, negative, neutral) for the particular user. After that it will display the positive tweets percentage:22.22%, negative tweets percentage 11.11%, neutral tweets of 66.6% and the results are plotted on bar and pie chat. In the pie graph tweets are categorized into 66.7% of neutral tweets ,22.2% of positive tweets and 11.1% negative tweets of a particular user which we taken. The bar graph represents the number of tweets of each category.

VII. CONCLUSIONS

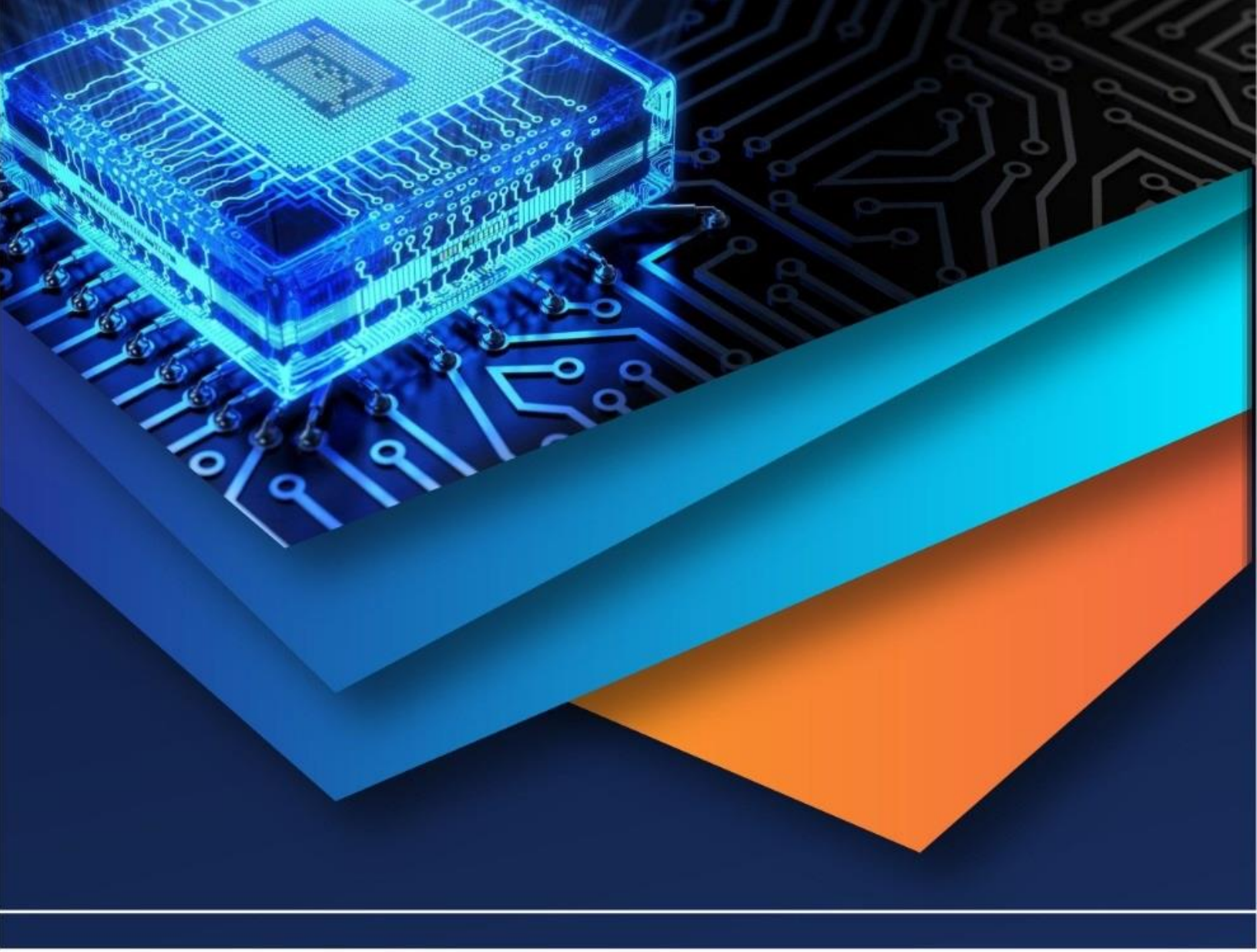
The previous work is based on SUMBLR frame work (continuouS sUmmarization by stream cLusteRing). tweets are summarized using TCV rank summarization algorithm and tweets are formed into clusters and that tweets are categorized based on evolution. To overcome this problem, we have proposed a method to improve the functionality of the Lex Rank method for summarization of the tweets.

The tweets streams are collected directly from the twitter API and summarized using Lex Rank algorithm which has best efficiency and flexibility compared to other text summarization algorithm and similar tweets are formed into clusters. Also categorize tweets into positive, negative and neutral tweets using sentiment analysis where tweets are collected by using Hashtags (screen names) of twitter individual account.

In future with new extension packages of python create better GUI (graphical user interface) and provide dynamic way to screen names for summarization and categorization.

REFERENCES

[1] C. C. Aggarwal, J. Han, J. Wang, and P. S. Yu, "A framework for clustering evolving data streams," in Proc. 29th Int. Conf. Very Large Data Bases, 2003, pp. 81–92.
[2] T. Zhang, R. Ramakrishnan, and M. Livny, "BIRCH: An efficient data clustering method for very large databases," in Proc. ACM SIGMOD Int. Conf. Manage. Data, 1996, pp. 103– 114.
[3] P. S. Bradley, U. M. Fayyad, and C. Reina, "Scaling clustering algorithms to large databases," in Proc. Knowl. Discovery Data Mining, 1998, pp. 9–15.
[4] L. Gong, J. Zeng, and S. Zhang, "Text stream clustering algorithm based on adaptive feature selection," Expert Syst. Appl., vol. 38, no. 3, pp. 1393–1399, 2011.
[5] Q. He, K. Chang, E.-P. Lim, and J. Zhang, "Bursty feature representation for clustering text streams," in Proc. SIAM Int. Conf. Data Mining, 2007, pp. 491–496.
[6] J. Zhang, Z. Ghahramani, and Y. Yang, "A probabilistic model for online document clustering with application to novelty detection," in Proc. Adv. Neural Inf. Process. Syst., 2004, pp. 1617–1624.
[7] S. Zhong, "Efficient streaming text clustering," Neural Netw., vol. 18, nos. 5/6, pp. 790– 798, 2005.
[8] C. C. Aggarwal and P. S. Yu, "On clustering massive text and categorical data streams," Knowl. Inf. Syst., vol. 24, no. 2, pp. 171–196, 2010.
[9] R. Barzilay and M. Elhadad, "Using lexical chains for text summarization," in Proc. ACL Workshop Intell. Scalable Text Summarization, 1997, pp. 10–17.
[10] W.-T. Yih, J. Goodman, L. Vanderwende, and H. Suzuki, "Multidocument summarization by maximizing informative contentwords," in Proc. 20th Int. Joint Conf. Artif. Intell., 2007, pp. 1776–1782.
[11] J. Xu, D. V. Kalashnikov, and S. Mehrotra, "Efficient summarization framework for multi- attribute uncertain data," in Proc. ACM SIGMOD Int. Conf. Manage., 2014, pp. 421–432.
[12] B. Sharifi, M.-A. Hutton, and J. Kalita, "Summarizing microblogs automatically," in Proc. Human Lang. Technol. Annu. Conf. North Amer. Chapter Assoc. Comput. Linguistics, 2010, pp. 685–688.
[13] D. Inouye and J. K. Kalita, "Comparing twitter summarization algorithms for multiple post summaries," in Proc. IEEE 3rd Int. Conf. Social Comput., 2011, pp. 298–306.
[14] S. M. Harabagiu and A. Hickl, "Relevance modeling for microblog summarization," in Proc. 5th Int. Conf. Weblogs Social Media,



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY