IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

*International Journal of Advanced Research in Computer and Communication Engineering*
*Vol. 5, Issue 6, June 2016*

# Timeline Generation for Progressive Tweet Stream

**Dhanshri A. Nevase[1], Prof. Amrit Priydarshi[2]**

PG Student, DGOI, Pune, Dept of Computer Engineering, Savitribai Phule Pune University, Maharashtra, India[1]

Asst. Professor, DGOI, Pune, Dept of Computer Engineering, Savitribai Phule Pune, University, Maharashtra, India[2]

**Abstract:** Now a day's use of twitter, face book such type of micro blogging services are increased. People post his services on this sites every day .Going through this millions of tweets are difficult because that tweets contains redundancy and noise. This paper consider continuous arriving tweet stream to generate timeline. Timeline is a way of displaying a list of event in chronological order. To generate a timeline first require to cluster all tweet stream of related topic then summarize that tweet stream by continuous summarization and then generate timeline automatically from tweet stream. In traditional method dates of summaries are already pre-defined to produce timeline. In this paper timeline discovers changing dates and topic changes and generate timelines dynamically during the process of continuous summarization. This paper focus on efficiency and scalability factors.

**Keywords:** continuous summarization, tweet stream, timeline, summary.

## I. INTRODUCTION

Social media services twitter, facebook resulted in the lots of short text messages. Twitter receives over 400 millions tweets per day from different sources as news , blogs and many more. So problem of information overload is occur . To avoid this problem tweet summarization is used. Summarization is nothing but minimizing the size of document such that document must maintain important points of original document.. In tweet summarization, tweets occur in fast & continuously manner. So, tweet summarization consider temporal feature of arriving tweets. In temporal feature two methods drill down & roll up are used. In drill down method summaries are depends on short period of time (e.g. summary between 11am to 11pm on 29th August).Opposite of that in roll up method summaries contains larger period of time (e.g. Summary between 15th August to 30 August).In continuous tweet stream summarization three important concepts are considered as efficiency, flexibility & topic detection. But this is not an easy task because of arriving tweet stream in continuous & unwanted manner. In existing system they consider static & small scale data. Second they perform iterative summarization for every possible given time duration. And third the result of summaries is not depending on temporal feature. In current system we introduce a novel summarization framework called Sumblr (continuouS sUMmarization By stream cLusteRing). The framework consists of three main components, as clustering of tweet streams module, document Summarization module and the Timeline Generation module. In the clustering of tweet streams module, we design an efficient tweet stream clustering algorithm, an online algorithm allowing for effective clustering of tweets with only one pass over the data. The document summarization module supports two kinds of summaries as online and historical summaries Online summaries are used to detect topic evolution. Whenever large variation occur at a particular moment implies sub topic changes and create new node on the timeline. Topic evolution detection method is important that produces timelines by monitoring two kinds of variations. In this paper consider two different variations as summary based variations (SUM) and volume based variations (VOL) .

## II. LITERATURE SURVEY ON SUMMARIZATION AND TIMELINE GENERATION

A. Document Summarization

Document summarization is classified as extractive & abstractive summarization. Abstraction involves paraphrasing sections of the source document. Extractive performs the automatic system extracts objects from the entire collection, without modifying the objects themselves.

- Title : Multi-Document Summarization by Maximizing Informative Content-Words [2]

Author : W.-T. Yih, J. Goodman, L. Vanderwende, and H. Suzuki.

In this paper two components are used for multi document summarization. First component generate score for each word in the set of document using machine learning. Second component find set of sentences from document cluster for maximize the scores.

- Title : Document summarization based on data reconstruction [3] .

Author : Z. He, C. Chen, J. Bu, C. Wang, L. Zhang, D. Cai, and X. He.

This paper gives data reconstruction using document summarization based on data reconstruction (DSDR ) method. They use two approaches as linear construction & non negative linear construction. In linear construction take document by linear combination of selected

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
*International Journal of Advanced Research in Computer and Communication Engineering*
*Vol. 5, Issue 6, June 2016*

sentences. In non negative linear reconstruction only additive but not subtractive linear combination is used.

▪ Title : A participant-based approach for event summarization using twitter streams [4].
Author : C. Shen, F. Liu, F. Weng, and T. Li.
In this paper different types of events are considered. Participant take a part into event. So this paper give summarization based on participant & event. They generate a textual description of the scheduled events that are reported on Twitter.

▪ Title : On Summarization and Timeline Generation for Evolutionary Tweet Streams [1] .
Author: Zhenhua Wang, Lidan Shou, Ke Chen, Gang Chen, and Sharad Mehrotra. This paper is based on online summarization & historical summarization. To generate online summaries, take topic related tweet stream without any previous knowledge. Store all the tweets in each segment & select only one tweet as part of summary to reduce space as well as computation cost. To generate historical summaries maintain TCV snapshots.

B. Timeline detection:
Different visualization techniques are used to analyze massive contents in social media. Timeline is uesd for such type of visualization that makes analysis process easier and faster.

▪ Title : Characterizing debate performance via aggregated twitter sentiment [5].
Author : N. A. Diakopoulos and D. A. Shamma,
This paper generate timeline based on presidential debates by twitter sentiment in 2008.
Title : A visual backchannel for large-scale events [6].
Author : M. Dork, D. Gruen, C. Williamson, and S. Carpendale
This paper is based on back channel for conversations around events.
Title: Evolutionary timeline summarization: A balanced optimization framework via iterative substitution [7].
Author : R. Yan, X. Wan, J. Otterbacher, L. Kong, X. Li, and Y. Zhang, This paper propose evolutionary timeline summarization (ETS) method . It consist of series of predefined time stamped set to compute evolution timelines ETS does not focus on efficiency and scalability issues.

▪ Title : On Summarization and Timeline Generation for Evolutionary Tweet Streams [1] .
Author: Zhenhua Wang, Lidan Shou, Ke Chen, Gang Chen, and Sharad Mehrotra.
This paper generate summary based and volume based timeline dynamically . This methods discovers changing dates and generate timelines during the process of continuous summarization in online fashion.

## III. PROPOSED SYSTEM

In this paper design three algorithm for tweet text. In the first algorithm Summarizing all text tweets in Distributed Applications. The second algorithm Summarizing all text tweets in Time Intervals in Distributed Apps. And third algorithm is based on Summarizing all text tweets By Hidden Markov Model.

*A.* Pyramidal Time Frame:-
At a particular moment store tweet cluster vector is called snapshot. To store each and every moment snapshot is not a practical because it required large storage space.For storing snapshot use different levels of granularity depending on recency.The order of snapshot is changed from 0 to $\lfloor \log_\alpha T \rfloor$ where T is elapsed time from beginning of the stream.
The snapshots of different order are maintained as follows:
• Snapshots of the i-th order occur at time intervals of $\alpha i$, where $\alpha$ is an integer and $\alpha \geq 1$.
• At any given moment in time, only the last $\alpha l + 1$ snapshots of order i are stored.
For example, the clock time of 8 is divisible by $2^0$, $2^1$, $2^2$, and $2^3$ (where $\alpha = 2$). Therefore, the state of the microclusters at a clock time of 8 simultaneously corresponds to order 0, order 1, order 2 and order 3 snapshots. From an implementation point of view, a snapshot needs to be maintained only once.
The following observations are true:
• For a data stream, the maximum order of any snapshot stored at T time units since the beginning of the stream mining process is $\log_\alpha(T)$.
• For a data stream the maximum number of snapshots maintained at T time units since the beginning of the stream mining process is $(\alpha l + 1) \cdot \log_\alpha(T)$.
• For any user specified time window of h, at least one stored snapshot can be found within $(1+1/\alpha l-1)$ units of the current time.

| Order | Timestamps of snapshots in the same order |
|---|---|
| 4 | 81 |
| 3 | 54 27 |
| 2 | 72 63 45 36 18 9 |
| 1 | 84 78 75 69 66 60 57 51 48 42 |
| 0 | 86 85 83 82 80 79 77 76 74 73 |

Table 1 PTF with Let $\alpha$= 3 and l = 2

*B.* Tweet representation:-
The tweets are reperesented as tuples of tweet cluster Z , tweet word vocabulary V & desired number of tweets n . So TCV that is tweet Cluster Vector represent as (Z,V,n).
Any text document is denoted as TF –IDF score. The TF-IDF score is calculated by using tweet cluster and tweet word vocabulary. Similarity between two tweet stream is calculate as cosine similarity between two tweet word called as score. From all these score select n number of tweets which is having maximum score.
• Summarizing all text tweets in Distributed Applications

**Algorithm 1** SUMMALLTEXT

**INPUT**: Tweet corpus $Z$, tweet word vocabulary $V$, desired number of tweets $n$
**OUTPUT**: Set of key tweets $T$

**for** $i \in Z, w \in V$ **do**
  $\mathbf{z}_i(w) = \text{tfidf}(w, i, Z)$
**end for**
**for** $i \in Z$ **do**
  $\text{score}(i) = \sum_{j \in Z} \text{cosine}(\mathbf{z}_i, \mathbf{z}_j)$
**end for**
$T$ = top $n$ tweets with maximum score

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

*International Journal of Advanced Research in Computer and Communication Engineering*
*Vol. 5, Issue 6, June 2016*

• Summarizing all text tweets in Time Intervals in Distributed Application

The second algorithm is useful for generating all text tweets within give time segment TS . To generate historical summaries time interval is required . So, tweet is represented as minimum activity threshold and time segment TS.

**Algorithm 2** SUMMTIMEINT

**INPUT**: Tweet corpus $Z$, tweet word vocabulary $V$, desired number of tweets $n$, minimum activity threshold $\ell$, time segments $TS$
**OUTPUT**: Set of key tweets $T$

$TS' = \{s \in TS | \text{tweet volume in segment } s > \ell\% \text{ of all tweets}\}$
**for** each segment $s \in TS'$ **do**
  $Z[s] = Z$ restricted to time $s$
  $T_s = SummAllText(Z[s], V, n/|TS'|)$
**end for**
$T = \bigcup T_s$

• Summarizing all text tweets By Hidden Markov Model

**Algorithm 3** SUMMHMM

**INPUT**: Tweet corpus $Z$, tweet word vocabulary $V$, desired number of tweets $n$, minimum activity threshold $\ell$
**OUTPUT**: Set of key tweets $T$

Learn $\Theta$ by iterating the equations in Table 1 until convergence
Infer time segments $TS$ by the Viterbi algorithm [11]
$TS' = \{s \in TS | \text{tweet volume in segment } s > \ell\% \text{ of all tweets}\}$
**for** each segment $s \in TS'$ **do**
  $Z[s] = Z$ restricted to time $s$
  $T_s = SummAllText(Z[s], V, n/|TS'|)$
**end for**
$T = \bigcup T_s$

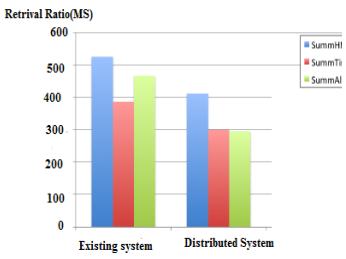
C. Timeline Detection:

The topic evolution method is main for generating timeline module. This method generate timeline by considering only real time case. When large variations occur when stream is continuously processing implies that sub topic changes and it create new node on the timeline . This method first collect all tweets based on the time units and then process stream continuously .Whenever large variation occur create new node on existing timeline . Finally it return timeline.

## IV. RESULT AND DISCUSSIONS

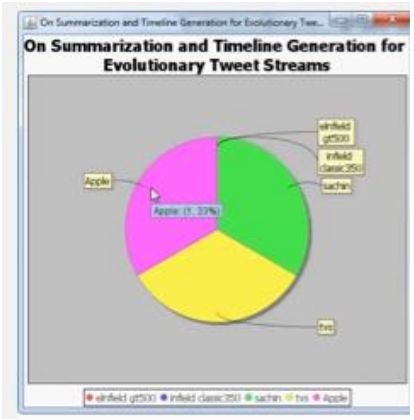
A. Document Summarization

This graph shows efficiency of proposed system is better than existing system. The result shows that retrieval rations in millisecond. Here compare three different algorithm from which Sum all text algorithm required minimum retrieval ratio.




B. Timeline Detection :

Timeline generate pie chart to show percentage of tweets according to rank .




## V. CONCLUSION

In this paper design summarization in distributed application using tweet cluster vector. The first algorithm Sum all text algorithm gives continuous summarization in distributed application . For generating this summary, uses structure called as tweet cluster vector. The second algorithm Sum Time Int generate historical summaries which is based on given time interval in distributed application. The third algorithm generate summaries using Hidden markov model. The result shows effectiveness of algorithm . Also it generate pie chart to show timelines as topic changes.

## REFERENCES

[1] Zhenhua Wang, Lidan Shou, Ke Chen, Gang Chen, and Sharad Mehrotra, " On Summarization and Timeline Generation for Evolutionary Tweet Streams," IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 27, NO. 5, MAY 2015 .

[2] D. Wang, T. Li, S. Zhu, and C. Ding, "Multi-document summarization via sentence-level semantic analysis and symmetric matrixfactorization," in Proc. 31st Annu. Int. ACM SIGIR Conf. Res.Develop. Inf. Retrieval, 2008, pp. 307–314.

[3] Z. He, C. Chen, J. Bu, C. Wang, L. Zhang, D. Cai, and X. He, "Document summarization based on data reconstruction," in Proc. 26th AAAI Conf. Artif. Intell., 2012, pp. 620–626.

[4] C. Shen, F. Liu, F. Weng, and T. Li, "A participant-based approach for event summarization using twitter streams," in Proc. Human Lang. Technol. Annu. Conf. North Amer. Chapter Assoc. Comput. Linguistics, 2013, pp. 1152–1162.

[5] N. A. Diakopoulos and D. A. Shamma, "Characterizing debat performance via aggregated twitter sentiment," in Proc. SIGCHI Conf. Human Factors Comput. Syst., 2010, pp. 1195–1198.

[6] M. Dork, D. Gruen, C. Williamson, and S. Carpendale, "A visual backchannel for large-scale events," IEEE Trans. Vis. Comput. Graph., vol. 16, no. 6, pp. 1129–1138, Nov. 2010.

[7] R. Yan, X. Wan, J. Otterbacher, L. Kong, X. Li, and Y. Zhang,"Evolutionary timeline summarization: A balanced optimization framework via iterative substitution," in Proc. 34th Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval, 2011, pp. 745–754