International Journal of Recent Trends in Engineering & Research
ISSN(ONLINE) : 2455-1457
IMPACT FACTOR : 3.344

Incremental Short Text Summarization On Comments In Real Time From Social Network Services

Dhivyabharathi.s[1], Suriya.k[2],Shalini.R[3] and Vinitha.R[4]
[1]*Computer Science and Engineering,Kathir College Of Engineering*
[2,3,3]*Information Technology, Kathir College Of Engineering*

Abstract- In this paper, We mainly focuses on comments which has been posted in social sites like facebook, twitter etc. This is mainly used to improve the quality of comments by grouping comments with similar content together and generate a concise opinion summary for this message. In this we are using a IncreSTS algorithm that update clustering results with latest incoming comments in real time.Therefore,We design an at-a-glance visualization interface that help the users to identify the comments easily and rapidly get an overview of the summary. From this experimental results,We possesses the advantages of high efficiency, high scalability, and better handling outliers, which justifies the practicability of IncreSTS on the target problem.

Keywords:real-time short text summarization, incremental clustering, comment streams, social network services

I. INTRODUCTION

Data mining is the process of discovering interesting patterns (or knowledge) from large amounts of data. The data sources can include databases, data warehouses, the Web, other information repositories, or data that are streamed into the system dynamically.Data mining, the extraction of hidden predictive information from large databases, is a powerful new technology with great potential to help companies focus on the most important information in their data warehouses. In this paper, Social network services (SNS) are prevalent and have become important communication platforms in our daily life. According to the 2012 statistics by the largest social networking site Facebook, there are over 500 million daily active users and an average of 3:2 billion interactions (including Likes and Comments) is generated each day. Besides, Micro-blogging giant Twitter has over 400 million user base and there are close to 200 million messages posted in a day. Due to the popularity and convenience of these platforms, celebrities, corporations, and organizations also set up social pages to interact with their fans and the public. As can be observed, not only the quantity of comments is large, but also the generation rate is remarkably high. Users unnecessarily and almost impossibly go over the whole comment list of each message.

In this paper, we do not focus on traditional comment streams that usually express more complete information, such as the discussion on products or movies. We target at comment streams in SNS that are in short text style with casual language usage. For each social message, our main objective is to cluster comments with similar content together and generate a concise opinion summary. We want to discover how many different group opinions exist and provide an overview of each group to make users easily and rapidly understand. On the other hand, the techniques of document clustering based on topic modeling concepts, such as Latent Dirichlet Allocation (LDA) and Latent Semantic Analysis(LSA), are another possibility to cope with this problem. Besides, the process of parameter estimation is time-consuming, and thus they are not applicable to real-time tasks.

1.1 Clustering

Clustering analyzes data objects without consulting class labels. Clustering can be used to generate class labels for a group of data which did not exist at the beginning. The objects are clustered or grouped based on the principle of maximizing the intra-class similarity and minimizing the inter-class similarity.

In this paper, we model the short text summarization as a clustering problem.To meet the practical requirement on SNS and enable the real-time processing, we define a new incremental clustering problem. Detailed definitions are presented in this section. Consider two comments represented in the term vector model, va = (t1;a; t2;a; :::; tN;a) and vb =(t1;b; t2;b; :::; tN;b). Each dimension corresponds to a separate term, and N is the number of dimensions. Since we define that the weights of terms are equal, if the term ti occurs in the comment va , ti;a will be set to 1. Otherwise, ti;a will be set to 0. Note that the vectors are not normalized to unit length. The reason for this design is that the length of each comment is usually very short compared to other text documents. In this situation, normalization is not so helpful for determining the similarity between vectors. Moreover, it has been widely observed that text data have directional properties. where va _ vb is the inner product of two vectors, and D is a positive integer constant. The denominator of original cosine similarity is the product of the lengths of two vectors. However, we regard that the similarity of two comments should not be affected by their vector lengths due to the characteristic of short length. Most comments are composed of only several words (not even a sentence). y will also have corresponding mutual sub-terms. Therefore, the value of inner product will be higher.

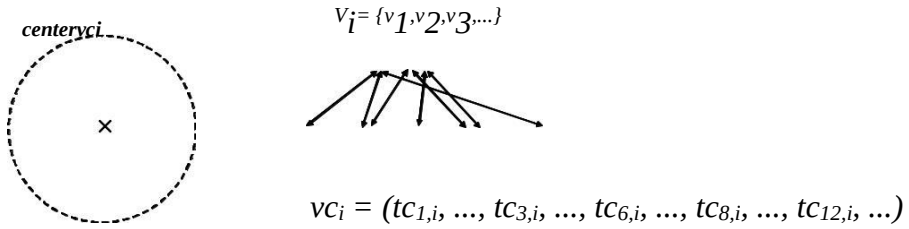

Fig 1.1 An illustration of the designed data structure for a cluster.

II. MODULE DESCRIPTION

2.1 Problem Description

We focus on the comment stream added for one message on SNS and aim to generate the immediate summary of comments. The problem we tackle is described as follows.
Givena set of comments S, and the desired number of groups k, find top-k groups $fC_1,C_2,\ldots,C_j,\ldots,C_k g$ which have top-k most comments, and the number of comments in C_j is larger than or equal to that of comments in C_{j+1} (i.e., $jC_j j C_{j+1} j$). Not all comments in S should be included in top-k groups. Moreover, the comments in C_j express similar opinions and are a subset of S. Our main objective is to discover top-k groups where the comments in the same group express similar opinions while the comments belonging to different groups express diverse points of view.

2.2 Modules

- New User Registration
- Upload and view posts
- Rating and filtering
- Summarization
- Sentiment analysis

2.2.1 New User Registration

In this module user create account from the social network website after login the web pages user password will be check the data base and authentication user only allow the main pages of the

project. Once a message is posted on SNS, users can leave comments immediately and the number of comments may rise quickly and continuously.

2.2.2 Upload and view posts

In this module user can search the friend list and request upload the image file after the friends will open and give the comments to the particular user the command based the characters will be denote and predicting the algorithm of clustering in the conversation view.

2.2.3 Rating and filtering

Social filtering systems that use ratings require a large number of ratings to remain viable. The effort involved for a user to rate a document may outweigh any benefit received, leading to a shortage of ratings filtering applications.

2.2.4 Summarization

Summarization is the process of reducing a text document with a computer program in order to create a summary that retains the most important points of the original document.

2.2.5 Sentiment analysis

Sentiment Analysis is the process of determining whether a piece of writing is positive, negative or neutral. It's also known as opinion mining, deriving the opinion or attitude of a speaker

III. SYSTEM DESCRIPTION

3.1 System Work

As can be observed, not only the quantity of comments is large, but also the generation rate is remarkably high. Users unnecessarily and almost impossibly go over the whole comment list of each message. However, we may still desire to know what are they talking about and what are the opinions of these discussion participants. With these motivations, we are inspired to develop an advanced summarization technique targeting at comment streams in SNS. Numerous studies and systems have proposed techniques and mechanisms to generate various types of summaries on comment streams. We explore the problem of incremental short text summarization on comment streams from social network services. We model this problem as an incremental clustering task and propose the IncreSTS (standing for Incremental Short Text Summarization) algorithm to discover the top-k clusters including different groups of opinions towards one social message. For each comment cluster, important and common terms will be extracted to construct a key-term cloud. This key-term cloud provides an at-a-glance presentation that users can easily and rapidly understand the main points of similar comments in a cluster.

IncreSTS, which can incrementally update clustering results with latest incoming comments in real time. With the output of IncreSTS, we design a visualization interface consisting of basic information, key-term clouds, and representative comments. This at-a-glance presentation enables users to easily and rapidly get an overview understanding of a comment stream. From extensive experimental results and a real case demonstration, we verify that IncreSTS possesses the advantages of high efficiency, high scalability, and better handling outliers, which justifies the practicability of IncreSTS on the target problem

IV. RELATED WORK

Owing to the large quantity of user-generated data on SNS, the research topics on alleviating the information overload problem and discovering useful knowledgehave attracted much attention recently. In addition to the comment stream data discussed in this paper, previous works also target at different types of social data and explore various research topics related to solving the information

overload problem on SNS. In this section,we can broadly classify these works into five categories:1) Human-Assisted Mechanisms, 2) Summarization, 3)Rating and Filtering, 4) Topic and Event Detection, and5) Sentiment Analysis. Note social network services are not restricted to well-known social websites, such as Facebook, Twitter, etc. The Web services providing interaction functionality for users can be generally included.

4.1 IncreSTS Algorithm: Incremental Version

Due to the high popularity of SNS, the number of comments for a specific message may increase very quickly, and users will request to view the summary of comments at any time. Moreover, since new messages appear continuously, users generally only view the summary of a specific message once and will not go back to browse the updated summary in the future. In such a context, to immediately produce the latest top-k clusters, we propose the IncreSTS algorithm that has the capability of incremental update. The primary concept of IncreSTS is to maintain the clustering result of the previous phase, and to incrementally update the clustering result with newly-incoming comment. Algorithm 2 outlines the algorithmic form of IncreSTS. Three main steps are involved in IncreSTS. Initially, we have to find the cluster which the newly-incoming comment new should be added into. In line 1 of Algorithm 2, the distances between new and all existing clusters are calculated. Among the clusters (in the setCb) whose distances are smaller than _r , we choose the cluster Cadded that has most comments. Thus, the comments in Ci are impossible to be absorbed into Cadded.

Procedure Key-Term Extraction

Input: *vc: the center of a cluster*
θ%: the threshold of overlapping percentage ***Output:*** $S_{key\text{-}terms}$*: the set of representative key-terms*

1. *Initialize* $S_{key\text{-}terms} = vc$;
2. ***for*** *each set of n-gram terms in* $S_{key\text{-}terms}$
3. *Eliminate the terms whose counts do not rank top k in this set;*
4. ***for*** *each* $termt_i in S_{key\text{-}terms}$
5. ***if*** *there exists any* $termt_j$ *where* $(t_j.ngram == t_i.ngram \&\& t_j.count >= t_i.count)$
6. ***if*** *there are over θ% of words* int_i *also contained* int_j
7. *Eliminate* t_i *from* $S_{key\text{-}terms}$;
8. ***for*** *each* $termt_i in S_{key\text{-}terms}$
9. ***if*** *there exists any* $termt_j$ *where* $(t_j.ngram > t_i.ngram)$ ***if*** *there are over θ% of words* int_i *also contained* int_j
10. *Eliminate* t_i *from* $S_{key\text{-}terms}$;
11. *Output the set* $S_{key\text{-}terms}$ *of representative key-terms;*

End

Algorithm: Algorithmic form of procedure Key-Term Extraction

4.1.1 Experimental Design

We collect real comment streams from Facebook through Facebook Graph API. Among the top25 Facebook pages (with most number of likes) in September 2012, we choose 10 of them, and for each page, 10 social messages having from 1000 to 3500 comments are retrieved. Table 1 summarizes the detailed information of this dataset comprising 100 comment streams totally. Experimental data are obtained from the average results of these streams. In addition, on average each message has 2072 comments, and each comment has 31:2 characters.

After transforming each comment into the term vector model representation, if the longest n-gram (denoted as N GRAM) is set to 3, each comment is composed of 10:3 terms on average. If

NGRAM is set to 5, each comment consists of 14:5 terms. It can thus be perceived that the lengths of most comments are short. On the other hand, the average numbers of shares and likes reach over 7000 and 80000, indicating each social message attracts much attention. This emphasizes the necessity of providing an at-a-glance summarization for interested users.

	#streams	#comments (per stream)	#characters (per comment)	#terms (per comment) (3-gram)	#terms (per comment) (5-gram)	#shares (per stream)	#likes (per stream)
Ladygaga	10	2350	33.2	11.2	16.3	6286	102778
JustinBieber	10	2638	29.0	10.0	14.3	3193	60883
Eminem	10	1677	33.8	11.4	16.5	2137	33573
Rihanna	10	2463	28.3	9.4	13.2	4320	92694
Shakira	10	2494	30.3	10.9	15.4	6650	69098
Facebook	10	1675	37.9	12.3	18.0	38	13513
TheSimpsons	10	1255	32.7	10.1	14.2	7491	69564
Southpark	10	1740	29.4	9.0	12.2	3223	49693
Harrypottermovie	10	2859	29.5	9.5	13.1	8009	156239
Disney	10	1565	28.4	9.0	12.3	35401	199024
Average	10	2072	31.2	10.3	14.5	7675	84706

TABLE 1: Detailed information of comment streams collected from Facebook.

V.CONCLUSION AND FUTURE WORK

In this model, we use a novel incremental clustering problem and propose the IncerSTS algorithm which can update the clustering results with latest incoming comments in real time and the content of any vulgar can be properly hidden which may increase the quality of comments.With the output of IncreSTS, a visualization interface consisting of basic information, Key terms clouds and representative comments. From these experimental results, IncreSTS algorithm possesses the advantage of high efficiency, scalability and better handing Outliers which justifies the practicability of IncreSTS on target problem.

REFERENCES

1. J. Allan, J. Carbonell, G. Doddington, J. Yamron, and Y. Yang. ”On-line new event detection and tracking”. Proc. of the 21th annual international ACM SIGIR conference on Research and development in information retrieval(SIGR ’98), pages 37–45, 1998.
2. M. Ankerst, M. M. Breunig, H.-P. Kriegel, and J. Sander. “Optics: Ordering points to identify the clustering structure”. Proc. of the 1999 ACM SIGMOD International Conference on Management Of Data (SIGMOD ’99), 28(2):49–60, 1999.
3. S. Baccianella, A. Esuli, and F. Sebastiani.”Multi-facet Rating of Product Reviews”. Proc. of the 31st European Conference on IR Research on Advances in Information Retrieval (ECIR’09), pages 461– 472, 2009.
4. H. Becker, M. Naaman, and L. Gravano. “Learning Similarity Metrics for Event Identification in Social Media”. Proc. of the 3rd ACM International Conference on Web Search and Data Mining (WSDM’10), pages 291–300, 2010.
5. H. Becker, M. Naaman, and L. Gravano.” Selecting Quality Twitter Content for Events”. Proc. of the 5th International AAAI Conference on Weblogs and Social Media (ICWSM’11), pages 442–445, 2011.
6. M. S. Bernstein, B. Suh, L. Hong, J. Chen, S. Kairam, and E. H. Chi. “Eddi: Interactive Topic-based Browsing of Social Status Streams”.Proc. of the 23nd Annual ACM Symposium on User Interface Software and Technology (UIST’10), pages 303–312, 2010.
7. J. Bollen, H. Mao, and A. Pepe. “Modeling Public Mood and Emo-tion: Twitter Sentiment and Socio-Economic Phenomena”. Proc. of the 5th International AAAI Conference on Weblogs and Social Media (ICWSM’11), pages 450–453, 2011.
8. D. Chakrabarti and K.Puner. “Event Summarization Using Tweets”.Proc. of the 5thInternational AAAI Conference on Weblogs and Social Media (ICWSM’11), pages 66–73, 2011.
9. J. Chen, R. Nairn, L. Nelson, M. Bernstein, and E. H. Chi. “Short and Tweet: Experiments on Recommending Content from Information Streams”. Proc. of the ACM SIGCHI Conference on Human Factors in Computing Systems (CHI’10), pages 1185–1194, 2010.
10. D. Comaniciu and P. Meer. “Mean shift: A robust approach toward feature space analysis”. IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(5):603–619, 2002.`