ISSN: 2278 – 1323
International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)
Volume 5, Issue 5, May 2016

Classification of Microblogging Messages into General Topics Based on the Content of the Messages with Clustering

Santrupti S. Desai, Manohar Madgi

Abstract— **Since microblogging messages such as tweets are of short length and noisy, clustering is a major challenge. And due to sparseness, the traditional clustering techniques are inaccurate as well as inefficient. So this paper presents a new text representation framework where microblogging messages which are short and noisy are classified into general topics based on the content of messages with clustering. We divide classification into two different tasks : (1) Stored data classification of user annotated data(tweets), and (2) Live tweets classification .**

Index Terms— **Microblogging, Clustering, Classification.**

I. INTRODUCTION

Microblogging is a activity or a form of communication which lets users to publish about their activities, opinions, status etc. as short and frequent messages online. The microblogging services trace what people are thinking in real time, providing valuable information to understand human behaviour. Several microblogging services which also referred as social networking sites include Twitter, Facebook, orkut etc. The most popular microblogging services are Twitter and Facebook. Commonly the microblogging messages from Twitter are called as tweets. Tweets may contain text, links, emotions or combination of these which are often related to specific event or topic of user's interest like politics, sports, music or some personal opinions and thoughts. In real time these messages are broadcasted to the followers of users.

In this work we will be more focusing on twitter wherein monthly active users are about 310 million by April 2016. In twitter, users may subscribe to other users' updates which is called following and the users who will follow a user are called followers. Twitter allows the use of two meta characters '@' and '#' where '@' is for indicating username and '#' indicates hashtag, which is a sequence of non whitespace predated by the hash character[2]. With the increasing popularity of the Twitter, a huge amount of data is produced for every second. So this overloaded information on the user interface tend to produce difficulties for the users to fulfill their wishes to get the needed information. Many times, many Twitter users do not have patience to read each and every message or tweet. They may have patience only to read the latest tweets and sometimes redundant tweets. In this case, in the huge amount of streaming data, many user interested tweets may be buried. With the large number of tweets, efficiently guessing the main topics from the tweets is a very difficult task for users. Therefore the abundant volume of short text massages which are noisy as well, will make the user difficult to conveniently access information to find and locate the specific topics which are user interested. This notably dispirit the user involvement in microblogging services and eventually makes microblogging services awfully accessible and not as much interesting.

Therefore it is enchanting to contribute users a well organized way to discover the topics/subtopics which are present in the microblogging messages of the users or followees (being followed by followers). For example, when rumors about third generation Apple iPad Air was spread, many microblogging users published or posted messages about this event. At the same time, some others users posted messages on other events such as speech of Prime Minister, cricket match etc. When the user wants to read messages related to Prime Minister speech, he/she has to spend much time for searching the messages related to this event since the user interface is overwhelmed by all these microblogging messages which are all shuffled or arranged in chronological order. Many times users are interested in specific topics and do not have patience glance all the messages. In such cases user interested topics are buried in disorganized messages which in turn makes users hard to locate and read their interested messages. To make microblogging messages easily and efficiently accessible to users and to make messages of user interest to be quickly identified by them , clustering is needed. So we propose clustering of microblogging messages to explore the information.

The various characteristics of microblogging or social media data imposes several challenges for directly applying existing text analytical methods for processing this data are as follows.

A. Short Length of Text

The major challenge is the limitation on text length. If we consider Twitter, it allows only 140 characters to post the texts. As the length of the messages is too short, they may contain sparse data and can't provide enough statistical information for measurement of similarity which is the important task in the text processing methods. And they can't

Manuscript received May, 2016.

Ms. Santrupti S. Desai, *Department of Computer Science and Engineering , K.L.E Institute of Technology affiliated to Visvesvaraya Technological University, Hubballi, India, 8884650571.*

Prof. Manohar Madgi*, Department of Computer Science and Engineering , K.L.E Institute of Technology affiliated to Visvesvaraya Technological University, Hubballi , India, 9448036699.*

provide much contextual clues to apply machine learning techniques.

B. Informal Language

Another challenge is the informal language used on microblogging services. In microblogging many slang words such as Hiiiiii, Good 9t, Gm, Coooool etc. which are convenient to the user are widely used. But this unstructured form of textual data is the source of bringing difficulties for text processing methods.

C. Constant changes in vocabulary

With new words and expressions or phrases, the microblog's vocabulary will be changing continuously. For example, in Twitter, new event names, product names, movie names etc. can show up suddenly in the language and turn out to be exceptionally well known. In this way, classification system of text can't be static and it should respond for the changes occurred in the language.

D. Use of various languages

Most of the microblogs especially Twitter is used by several users from everywhere throughout the world. Therefore numerous languages are utilized as a part of posts.

Due to sparseness, applying traditional techniques for clustering and classification is both inaccurate and inefficient. So this paper presents a new textual framework for classifying and clustering of live and stored tweets. For classification we used Naive Bayesian algorithm and for clustering we used k-means algorithm due to its simplicity and efficiency.

II. RELATED STUDY

Several works on microblogging messages include text categorization [9], influence study, sentiment analysis[7], event or topic modeling, text summarization etc.

Most of the recent related work concentrates on the elimination of the problem of data sparseness. One instinctive solution to solve this is to increase the short text with additional information to make it like a large document of text. Then traditional algorithms for clustering or classification can be applied to it. As discussed in [2,3,11], their primary focus is on combining short text messages with web search engines such as Bing, Google to extract more information about the short text. In [12] Poschko said that if two hashtags co-occur in a tweet, then they are similar. With co-occurrence frequency as a distance measure, he created a clustered graph.

The other works on clustering text-related entities typically focuses on bag-of-words(BOW). BOW model takes all the words of entity followed by reduction of dimensionality. It makes clustering computationally feasible as in [15,16].

As said in [4,5,13], more recent works focused on web searches and instead used data repositories. One of the biggest data source is Wikipedia. Short text messages can be strengthened with additional semantic knowledge by combining knowledge accessible within the Wikipedia. In [5,6], user-defined categories and concepts are extracted from Wikipedia and significant improvement in accuracy is shown by the experimental results. The selected Wikipedia articles' titles are used to increase the quality of text. However this approach will not catch the up-to-date information and when the input data is very volatile in its theme like news feeds, this approach is not suitable. And because of the time constraints, for real-time applications, online querying of Wikipedia and concepts are unsuitable.

In [5] Banerjee also said that usage of extra Wikipedia concepts other than titles did not render any notable improvements in various clustering algorithms' performance.

In [4], to gain more knowledge, Phan not only used the defined categories in Wikipedia but also brought out hidden topics of articles of Wikipedia. As mentioned in [5], even though it removes the data sparseness problem, it is more time consuming and there is need to know which all concepts of Wikipedia are useful.

For the classification of documents, given the document training data, the most well-known approaches start by assessing the words' co-occurrence matrix versus documents. It is surely understood, in any case, such count matrices have tendency to be exceedingly noisy and sparse, particularly when training data is moderately small. Subsequently, usually the documents are depicted in a feature space that is high dimensional, which is long way from optimal for algorithms of classification. A standard methodology to reduce dimensionality of features is feature selection. In this methodology, one chooses a subset of words, utilizing some pre-defined paradigm, and as features it uses only the chosen words for classification. In the task of information retrieval, other similar strategies for dimensionality reduction are Latent Semantic Indexing (LSI) and Probabilistic LSI.

An alternative methodology is to minimize the feature dimensionality by gathering 'similar' words into much littler number of word-groups, and utilize these groups as features. The crucial step in such strategies is the determination of 'words' similarity'. Formal optimal solution which inclines absolutely on information theoretical considerations for this is the usage of Information Bottleneck (IB) method .

All these works are different from our work in the following ways : Our work first retrieves both live and stored twitter data and then does clustering of tweets followed by the classification of those tweets into general topics.

III. PROPOSED FRAMEWORK

The proposed novel framework clusters the short and noisy microblogging messages such as tweets. This framework performs two tasks : (1) Offline classification of tweets i.e. classification of stored tweets and (2) Online classification of tweets i.e. classification of live tweets. The Fig. 1 represents the architecture of the new framework.

As shown in Fig. 1, the framework contains five main stages : (1) Retrieval of online/offline tweets, (2) Data pre-processing, (3) Feature extraction, (4) Clustering and (5) Classification of tweets.

In our framework, for live tweets classification, the user should have Twitter account and Twitter application which has OATH i.e. Obtain Authorization access. OATH access generates Access token, Access secret key, Consumer key, and Consumer secret key which are required for the retrieval of live tweets.

At first user sends authentication request and REST API by entering the valid keys as mentioned previously to the server to retrieve online or timeline tweets through GUI i.e. Graphical User Interface.

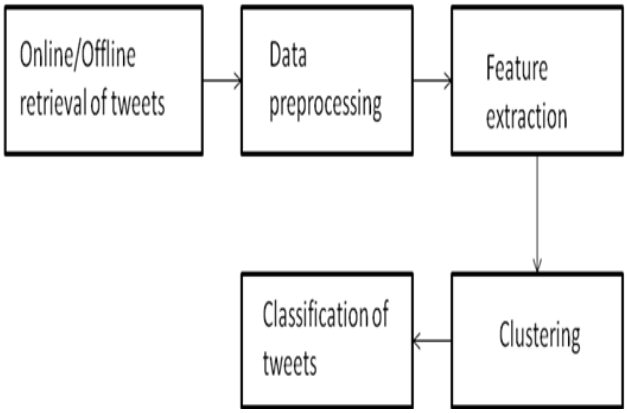

Fig. 1 : Architecture of new framework

Once live tweets are retrieved, they are clustered and classified into six categories i.e. into politics, sports, business, technology, entertainment and others as neutral. User can also select stored tweets classification i.e. offline classification. In that 3000 tweets from database are retrieved and then classified into six categories as mentioned above. This framework also lets the user to add new features to the training datasets.

A. *Retrieval of Online/Offline Tweets*

Twitter tweets can be retrieved using the Twitter API. User's timeline tweets can be retrieved using both REST API and Streaming API. Here we have used REST API .

1) Retrieval of online tweets : In order to get timeline tweets we have used REST API. REST API stands for Representational State Transfer. It requires the four keys such as Access token, Access secret key, Consumer key, Consumer secret key. Twitter user should have OATH access to his/her account. By obtaining authorization the user can get the access tokens. Recently Twitter developers have set limitation on retrieving most recent user timeline tweets to 20 at a time. And we have used this API to retrieve recent tweets of a particular link or profile (e.g. @TechCrunch). And it is limited to only 300 tweets per day by Twitter developers.

2) Retrieval of offline tweets: To perform this task database containing tweets should be needed. So we have used a MySQL database containing 3000 tweets for offline or document classification. Whenever the tweets of particular link or profile are retrieved, they all are imported to the database in our framework.

B. *Data Preprocessing*

It is a very important stage in our framework where it involves following steps.

1) Tokenization: It is a process of splitting up of stream of text into tokens(words, phrases or other meaningful elements). Here each tweet is broken up into words i.e. tokens and passed as input to next stage.

2) Stop word removal: In text mining, most of the frequently used words in English are unwanted words for mining. So such words are called stop words. And the division of the natural language are the stop words. Stop words include a, an, the, and, are, as, at, be, for, from, has, he, in, is, it, its, of, on, that, the, to, was, were, will, with etc. In this step the stop words should be eliminated to make the tweets look less heavier and more important for analytics. Dimensionality of the term space will be reduced by removing stop words from the tweets. Here we have used classic method for removing stop words.

3) Stemming: In this step root or stem of the words are identified. For example, the words define, defined, defining and definition all can be stemmed to the word "define" [3]. Here we have used stemming algorithm to reduce number of words, eliminate various suffixes, to have accurately matching stems, and to save time and memory space.

C. *Extraction of Features*

Extraction of features is also an important step in text classification and clustering. It is process of selecting the subset of the words form the training dataset. This subset acts as features in the process of text classification. In this work method used to represent text is Bag-Of-Words model i.e. BOW model. BOW model is the most widely recognized strategy to depict text. The text is divided into words in this technique and every word depicts a feature. For example, for sports dataset, sachin tendulkar, cricket, football, India etc. act as features. This step is useful for the classification and clustering in two ways : (1) by reducing the size of the vocabulary which can be trained used for applying the classier algorithm efficiently, and (2) by eliminating the noise features which in turn makes the clustering and feature extraction to increase the accuracy of classification.

D. *Clustering of Tweets*

Clustering is an unsupervised machine learning method which organizes a large amount of unordered text documents into small number of groups of similar objects. By using the similarity function, similarity between the objects can be measured. The clustering algorithm used in our work is k-means because of its simplicity and high efficiency.

1) K-means: K-means clustering algorithm is the simplest unsupervised learning algorithm, which aims to classify the test dataset through certain number of clusters i.e. fixed priori. The main idea of this algorithm is to divide n data points into k clusters where each data point in test dataset belongs to the cluster with nearest mean. Here data point refers to single tweet. As in [16] at first, k-means algorithm selects k random points from the test dataset and these k random points are assigned as centroids. Then, assigns each data point to the closest centroid which in turn results in the creation of k clusters. In the next step, to reduce the distance between the centroids and all the points in their cluster, the centroids are reassigned and reassigns each point to the nearest centroid. This process continues until the confluence is arrived.

Using several different metrics, the distance between data points and centroids can be measured. Among these, most widely used metrics are cosine and Euclidean distance. Since cosine distance is faster, and is better qualified in dealing with sparse matrices, and calculates the distance independent of the tweets' length, we used cosine distance metrics, which is the distance between two data points. So long tweet with

many words might still be considered more similar to a shorter tweet with less words. For example, if two data points i.e. two tweets containing five words in common, then cosine would result in a distance that is not dependent from how many times the five words appear in each of the tweets.

The cosine of the angle between two vectors is measured by the cosine distance i.e.

$$\cos\theta = \frac{x.y}{||x||||y||}$$

where x and y are TF-IDF vectors i.e. term frequency-inverse document frequency vectors corresponding to the documents x and y. The range of resulting distance is from -1 to 1 K-means algorithm is extremely dependent on centroid initialization. Clustering of tweets will help in improving the classification accuracy.

E. Classification of Tweets

In twitter, tweets will appear to the user in chronological order which makes user difficult to read or retrieve the user interested tweets. So it is necessary to classify the tweets into general topics such as politics, sports, business, technology, entertainment and neutral for better information retrieval. Text classification is the supervised machine learning technique, where given a set of training datasets that are classified into one or more predefined categories or classes to automatically classify new text documents. Several classification algorithms include nearest-neighbours, regression, decision trees, neural networks, support vector machines etc. In our work for classifying tweets into general topics, the classification algorithm used is Naive Bayes because of its high efficiency to implement and easy to follow mathematically.

The basic idea of this algorithm is to calculate the probability of a document D which belongs to a class C. Here document refers to the test dataset of tweets which we classify it as class having maximum posterior probability P(C|D). According to the Bayes theorem P(C|D) is calculated by the following mathematical formula.

$$P(C|D) = \frac{P(D|C)\,P(C)}{P(D)} \propto P(D|C)P(C)$$

where C={C_1,C_2,C_3,……C_k} i.e. k number of classes. In our work k=6 i.e politics, sports, business, technology, entertainment and neutral. So by using this Bayes classifier we classified both stored tweets as well as live tweets into six categories as mentioned before.

Fig. 2 shows the graph for 3000 stored tweets classification. It represents how many tweets belong to particular category among 3000 tweets. Fig. 3 shows the graph for the classification of 20 timeline tweets. It also depicts how many tweets belong to each class among 20 timeline tweets.

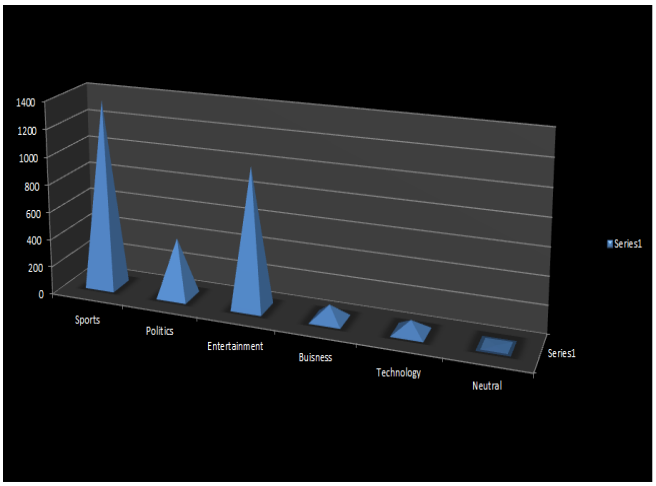

Fig. 2 : Graph for 3000 stored tweets classification

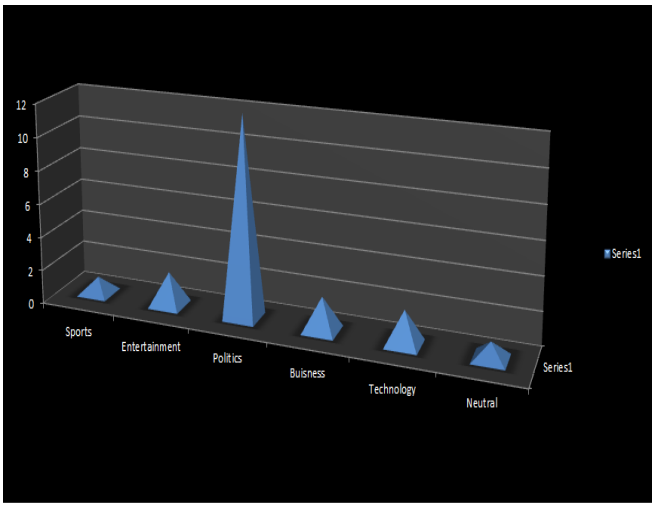

Fig. 3 : Graph for timeline tweets classification

IV. CONCLUSION

In this work, we presented a new framework for the classification of live twitter data as well as stored tweets with clustering. We have classified tweets into politics, sports, business, technology, entertainment and neutral. For non-real time tweets we have considered 3000 tweets which are stored in database.

The possible improvements of our work include classification of more than millions of non-real time tweets for which hadoop framework can be used. And we can apply this for real time tweets classification also if twitter developers allow to retrieve more number of tweets at a time.

REFERENCES

[1] OrenTsur, AdiLittman and AriRappoport, "Efficient Clustering of Short Messages into General Domains", Proc. of 7th Int. AAAI Conf., 2013.
[2] D. Bollegala, Y. Matsuo, and M. Ishizuka, "Measuring semantic similarity between words using Web search engines", Proc. of Int. WWW conf., 2007 .
[3] M. Sahami and T. Heilman , "A Web based kernel function for measuring the similarity of short text snippets", Proc. of Int. WWW conf., 2006.
[4] Phan, X.-H., Nguyen, L.-M., and Horiguchi, S, "Learning to classify short and sparse text & web with hidden topics from large-scale data collections", In Proc. of Int. WWW Conf. Apr. 2008, pp. 91-100.

[5] S.Banerjee, Ramanthan K, and Gupta , Clustering short text using Wikipedia, In Proc.of Int. ACM SIGIR conf., July 2007, pp. 787-788.
[6] P. Schonhofen , Identifying document topics using the Wikipedia category network, Proc. IEEE/WIC/ACM Int. Conf. on Web Intelligence, 2006.
[7] E.Bakshy,J.M.Hofman,W.A.Mason,andD.J.Watts,"Identifying influencers on Twitter," in Proc. 4th ACM Int. Conf. on Web Seach Data Mining,2011, pp.65–74.
[8] X. Hu, J. Tang, H. Gao, and H. Liu, "Unsupervised sentiment analysis with emotional signals," in Proc. 22nd Int. Conf. World Wide Web, 2013, pp. 607–618.
[9] Antonia Kyriakopoulou and Theodore Kalamboukis, "Text Classification Using Clustering" Journal of Machine Learning Research 3, 2003.
[10] C.Ramasubramanian, R.Ramya "Effective Pre-Processing Activities in Text Mining using Improved Porter's Stemming Algorithm", IJAR, Vol. 2, Issue 12, December 2013.
[11] W. Yih and C. Meek,"Improving similarity measures for short segments of text", Proc. AAAI, 2007.
[12] Jan Pöschko, "Exploring Twitter Hashtags", arXiv:1111.6553v1 [cs.CL], 28 Nov 2011.
[13] Hu, X., Sun, N., Zhang, C., and Chua, T.-S, Exploiting internal and external semantics for the clustering of short texts using world knowledge, CIKM, Nov. 2009, pp.919-928.
[14] Daniel Godfrey , Caley Johns , Carol Sadek ,Carl Meyer, Shaina Race" A Case Study in Text Mining: Interpreting Twitter Data From World Cup Tweets"arXiv:1408.5427v1 [stat.ML] 21 Aug 2014.
[15] Allie Mazzia, James Juett, "Suggesting Hashtags on Twitter", EECS 545, Computer Science & Engineering, University of Michigan.
[16] Anand Karandikar, "Clustering short status messages: A topic model based approach" Univ. of Maryland, MS,2010.

Miss Santrupti S. Desai, Mtech in Computer Engineering, from K.L.E.I.T Hubballi affiliated to Vishweshwarayya Technology University , Belagavi, Karnataka, India.

Prof. Manohar Madagi, Associate Professor, Department of Computer Science and Engineering, K.L.E.I.T Hubballi affiliated to Vishweshwarayya Technology University , Belagavi, Karnataka, India.