RESEARCH ARTICLE OPEN ACCESS

Social Sensor for Real Time Event Detection

Prajakta Patil[1], Pooja Patil[2], Amruta Salvi[3], Prof. Mrunmayee Hatiskar[4]
Department Of Computer Engineering[1, 2, 3, 4] RMCET (Mumbai University) Ambav, Devrukh, India[1, 2, 3, 4]

ABSTRACT
Social sites play an important role in our day to day life. Peoples are spending at most of the time on social networking sites and for this reason social networking sites have received a more attention in last few years. Social networking websites have received greater attention in the past few years .Twitter is one of the most popular networking site which received attention from peoples by providing its real time nature. The important characteristic of our system is its real-time nature. We create a system to collects the tweets from various groups of users and detect real time events like earthquake, traffic etc. To detect real time events, we build a classifier of comments based on features such as the keywords in a comments, the number of words. With the help of comments we will approximately find out the location of the target event. We assume each user as a sensor and apply priority base algorithm which is used for tweet analysis.
To resolve the traffic congestion problem, we have to consider the volume of the traffic, traffic speed, road occupancy etc. Classify tweets into a positive and negative class. Produce a probabilistic model for event detection. We create group of users on the basis of their interest such as peoples from traffic police department are post the tweets related to traffic event, also peoples form whether department are post their tweets related to flood etc. Each tweet is associated with user group, tweet, time and location. By processing time and location information, we can detect the real time events.
***Keywords*:** Classifier, microblogging, feature

I. INTRODUCTION

Twitter is a popular social networking site providing microblogging service where users post tweets. These tweets may tells an opinion about topics, feelings etc. We propose a method to automatically extract features from a tweet. .In order to train a classifier, supervised learning usually requires hand-labeled training data. With the large range of topics discussed on Twitter, it would be very difficult to manually collect enough data to train a sentiment classifier for tweets. Our solution is to use priority algorithm, in which our training data consists of tweets with emoticons.

The emoticons serve as noisy labels in a tweet. With the help of the our system API, it is easy to extract large amounts of tweets with emoticons in them

We present the results of our experiments and our thoughts on how to further improve results.

1.1. Defining Sentiment.

For the purposes of our project, we define sentiment to be a personal positive or negative feeling.

1.2. Characteristics of Tweets.

Twitter messages have many unique attributes, which differentiates our research from previous research:
Length: The maximum length of the message is 70 characters. From our training set, we calculate that the average length of a tweet is 8 words or 38 characters. With the help of our platform, it is very easy to collect millions of tweets for training

II. APPROACH

Our approach is to use one machine learning classifier and feature extractor, which is Priority based message stack queuing. We build a framework that treats classifiers and feature extractors as two distinct components.

2.1. Emotions.

Since the training process makes use of emoticons as noisy labels, it is crucial to discuss the role they play in classification. We will discuss in detail our training and test set in the Evaluation section. We strip the emotions out from our training data. Stripping out the emoticons causes the classifier to learn from the other features (e.g. unigrams and bigrams) present in the tweet. The classifier uses these non-emoticon features to determine the sentiment.

III. ALGORITHMS

We use Priority based message stack queuing algorithm for our project. We take advantage of the following properties to reduce the feature space.

Usernames Users often include usernames in their tweets in order to direct their messages..

3.1. Priority based message stack queuing:

Message Queuing is useful when the client application is often disconnected from the network. In our case the tweets can be send according to priority stack checking the date of telecast and the nature of message to be unique.

A message can have a priority that defines the order in which the message will be read from a queue.

In C#, we can also create Message Queues programmatically using the Create() method of MessageQueue class. With Create() method, the path of the new queue must be passed.

Example:

```
using System;
using System.Messaging; namespace FirstQueue
{
class Program
  {
static void Main(string[] args)
{
try
  {
if (!MessageQueue.
Exists(@".\Private$\FirstQueue"))
  {
MessageQueue.Create (@".\Private$\FirstQueue");
  }
MessageQueue queue = new
MessageQueue (@".\Private$FirstQueue");
queue.Send ("First Message ", " Label ");
   }
Catch (MessageQueueException ex)
  {
Console.WriteLine (ex.Message);
   }
  }
 }
```

IV. EVALUATION

4.1 Experimental Set-up.

We create a real time system blogging that includes following modules:

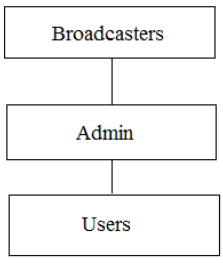

Figure 1. Modules

4.1.1 The Broadcast group.

The broadcast group is actually the registered people who operate as groups for casting the news from any part of the world and this news queue is streamed to the admin.

4.1.2 Admin Module.

Admin filters the news casts, check credibility and implement the priority queuing algorithm and telecasts to the users.

The tweets will be real time. So new updates are display timely.

Old news is given a time frame of display and is pulled off after it becomes outdated or irrelevant.

4.1.3 Users.

The users are the social media network users who can view and comment on the site. Users can send the real time trendy tweets.

And the users can benefit with current alerts and the location in case if it is concern.

V. FUTURE-WORK

In tweets machine learning techniques works well for classifying sentiment. We believe that the accuracy could still be improved. Below is a list of ideas we think could help in this direction.

Internationalization: We focus only on English sentences, but Twitter has many international users. It should be possible to use our approach to classify sentiment in other languages [1].

Utilizing emoticon data in the test set: From our training data emoticons are stripped. This means that if our test data contains an emoticon feature, this does not influence the classifier towards a class. This should be addressed because the emoticon features are very valuable [1].

VI. RELATED WORK

We consider each registered user as a sensor and apply particle filtering which are namely used for estimation of location. The working of particle filter is better than other comparable methods for locations estimation of target events. Each tweet has its own post Time so in a temporal model we show when target event occurs. How many tweets which are similar are posted on our platform depending on that target event behaviour can be analyzed. In spatial model, each tweet also has it's a location (from where tweets was posted) and by using "particle filter", can locate the targeted events from using tweets. Our work is on detection of real time events like earthquakes using tweets. Some researchers examined the twitter and they found out that it is one of the microblogging service. Real time nature an important characteristic of micro blogging service. So for our research twitter has been considered, to detect

the real time event depending on the tweets posted. For example using twitter people make comment on various topics and real time events. In the real time the tweet analysis also helps to know the important topics on which people are focusing.

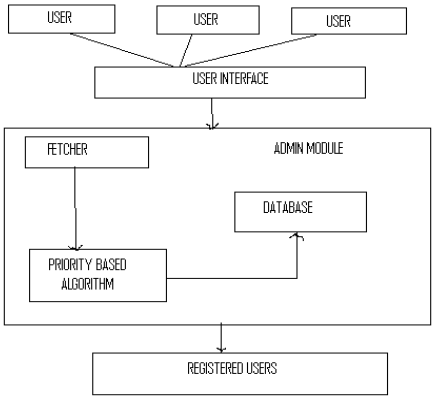

Figure 2. Proposed system

VII. CONCLUSION

Peoples need to be get notifications of disaster events in short time and Twitter is one of the most popular networking site which received attention from peoples by providing its real time nature. So We are creating a system to collects the tweets from various group of users and detect real time events like earthquake, traffic etc. by using Priority based message stack queuing algorithm and Classify tweets into a positive and negative class. So it will produce a probabilistic model for event detection.

REFERENCES

[1] "*Twitter sentiment classification using distant supervision*" by Alec Go, Richa Bhayani, Lei Huang.

[2] D. O. Computer, C. Wei Hsu, C. Chung Chang, and C. jen Lin. A practical guide to support vector classification chih-wei hsu, chih-chung chang, and chih-jen lin. Technical report, 2003.

[3] N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, March 2000.

[4] B. J. Jansen, M. Zhang, K. Sobel, and A. Chowder. Micro-blogging as online word of mouth branding. In *CHI EA '09: Proceedings of the 27th international conference extended abstracts on Human factors in computing systems*, pages 3859{3864, New York, NY, USA, 2009. ACM.

[5] T. Joachims. Making large-scale support vector machine learning practical. In B. SchAolkopf, C. J. C. Burges, and A. J. Smola, editors, *Advances in kernel methods: support vector learning*, pages 169{184. MIT Press, Cambridge, MA, USA, 1999.

[6] C. D. Manning and H. Schutze. *Foundations of statistical natural language processing*. MIT Press,1999.

[7] G. Mishne. Experiments with mood classification in blog posts. In *1st Workshop on Stylistic Analysis of Text for Information Access*, 2005.

[8] K. Nigam, J. Laberty, and A. McCollum. Using Maximum entropy for text classification. In *IJCAI-99 Workshop on Machine Learning for Information Filtering*, pages 61{67, 1999.

[9] B. Pang and L. Lee. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1-2):1{135, 2008.

[10] B. Pang, L. Lee, and S. Vaithyanathan. Thumbs up? Sentiment classification using machine learning techniques. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 79{86, 2002.

[11] J. Read. Using emoticons to reduce dependency in machine learning techniques for sentiment classification. In *Proceedings of ACL-05, 43nd Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2005.