Sentiment Analysis of Twitter Data using Statistical Methods

Jahiruddin
Department of Computer Science,
Jamia Millia Islamia (Central
University), New Delhi-110025,
India,
E-mail: jahir.jmi@gmail.com

ABSTRACT

Social media today has becomes a very popular tool in society. Millions of users share their opinions on different topics like politics, technologies, product and many more. Therefore social media is a rich source of data for opinion mining and sentiment analysis. In this paper we use the twitter data for sentiment analysis. First we use the Latent Dirichlet Allocation (LDA) to identify the key terms. These key terms are used to represent each tweet in n dimensional vector. Using this tweet vectors, we build a sentiment classifier, which is able to determine positive, negative, and neutral sentiment of each tweet. Experimental result show that our proposed method is efficient and out performs.

Keywords

Twitter, Latent Dirichlet allocation, Opinion mining, Sentiment analysis, Key term identification.

1. INTRODUCTION

Social media today has becomes a very popular tool in society. Millions of messages are flowing daily in popular social media websites like Twitter, Facebook. On these social media web sites, a huge number of users share their view on a number of topics and discuss current issues of the society. Due to easy accessibility of these social media and free format of message a large number of users shifted from traditional communication tools like email system to these social media sites. These social media sites becomes valuable sources of people’s opinions and sentiment as very large number of users post their opinions about products and services, express their views on political and religious issues, and share their thought about current problems of the society. Such data may be used as an important resource for marketing or social studies.

Twitter contains a very large number of short messages. The maximum length of a twitter message is 140 characters. The twitter message also called tweet. We use a dataset of tweets downloaded from twitter using its APIs. The contents of the tweet vary from personal opinion to public statements. Due to freely availability of a large number of tweets, twitter data can be used in opinion mining and sentiment analysis. Many manufacturing company may be interested to know public opinions about their product so that accordingly they can improve its quality. Political parties want to know, whether people like their manifesto or not. All these information may be obtained from twitter data, as a large number of tweets on different topics are posted by users daily.

In this paper, we study how sentiment analysis may be performs on twitter data. We will show how to use twitter data as a corpus for sentiment analysis. We use twitter data for the sentiment analysis due to following reasons:

- Twitter social media is used by a large number of people to express their opinion about different topics, thus it is a valuable source of people’s opinions.
- Twitter contains a huge number of tweets and it grows every day. The collected corpus may be very large.
- Twitter’s users vary from regular users to renown, company representatives to politicians, and even country ministers. Therefore, it is possible to collect tweets from different interests group of users.

We collected a corpus of 3100 tweets on three different events “Gaza under attack”, “Delhi Assembly Election 2015”, and “Union Budget 2015” distributed among three sets of tweets:

1. Tweets contains positive sentiment
2. Tweets that contains negative sentiment
3. Tweets that only state a fact do not express any sentiment

We use statistical methods to build a sentiment classifier that use the collected corpus as training and test data. These classifier models may be used in determining the sentiment of a new tweet as positive, negative, or neutral.

The remaining paper is organized as follows. Section 2 present a review of related work, our method is described in section 3. Evaluation of the method is given in section 4. Section 5 concludes the paper with future enhancement.

2. RELATED WORKS

Due to popularity of social media websites, opinion mining and sentiment analysis became a field of interest for many researchers. A very broad overview of the existing opinion mining and sentiment analysis work was presented in [1]. In this paper, the authors describe existing opinion mining and sentiment analysis techniques. However, not many researches in opinion mining and sentiment analysis considered blogs and even much less addressed social media. Yang et al. were used web-blogs data as a corpus for sentiment analysis [2]. They used emotion icons assigned to blog posts as indicators of users’ mood to determine the sentiment of the text. The authors applied Support Vector Machine (SVM) and Conditional random field (CRF) machine learning techniques to classify sentiments at the sentence level and then used several strategies to determine the overall sentiment of the document. They reported that the sentiment of the last sentence play an important role in determining the overall sentiment of the document. J. Read was used emoticons to form a training set for the sentiment classification to overcoming the domain, topic and time problems [3]. For this purpose, the author extracted paragraphs containing emoticons from Usenet newsgroups. The dataset was divided into positive and negative paragraphs based on existing happy and sad emoticons. He trained SVM and Naive Bayes classifier on emoticons dataset, and reported that the mean accuracy of the Naive Bayes classifier was 61.5%, while the SVM classifier was 70.1%. Go et al. build an algorithm that can classify tweets as positive or negative tweet [4]. They studied a number of clarifier based on n-gram and POS tags features. They reported that multi-nominal Naive Bayes based on unigram using mutual information outperform.

Another area of research on twitter data is event identifications. In [5], the authors proposed a general event identification framework from social media documents. They used similarity metric learning approaches to produce high quality clustering results. They reported that similarity metric learning method yield better performance than traditional approaches that use text-based similarity. Sakaki et al. proposed a method for identification of real time events from current twitter data [6]. In this paper they developed an algorithm to identify target events at real time. Their classifier was based on tweets features like keywords, number of words, and their context for detecting target event. They focused to identify earthquakes event only. In [7], the authors proposed a method to identify news event. The main feature of their method was that it deals noisy data. They described a number of noise handling strategies for overcoming noise. Becker et al. were developed a system that divided the twitter data into events and non-event tweets [8]. They consider temporal, social, topical, and Twitter-centric features for classification of tweets into event and non-event class. It classified the tweets into two classes only.

3. PROPOSED SYSTEM

We now present the complete architecture of our system, which is designed to classify the tweets based of their sentiments. This may be used by a company for analysis of their product. The proposed system is shown in figure 1. This is characterized by following key functionalities – Tweets Crawling, Tweets Pre-Processing and Tokenization, Key Terms Identification and Tweets' Feature Vector Generation, and Sentiment Analysis. A brief description about these functionalities is given in the following paragraphs:

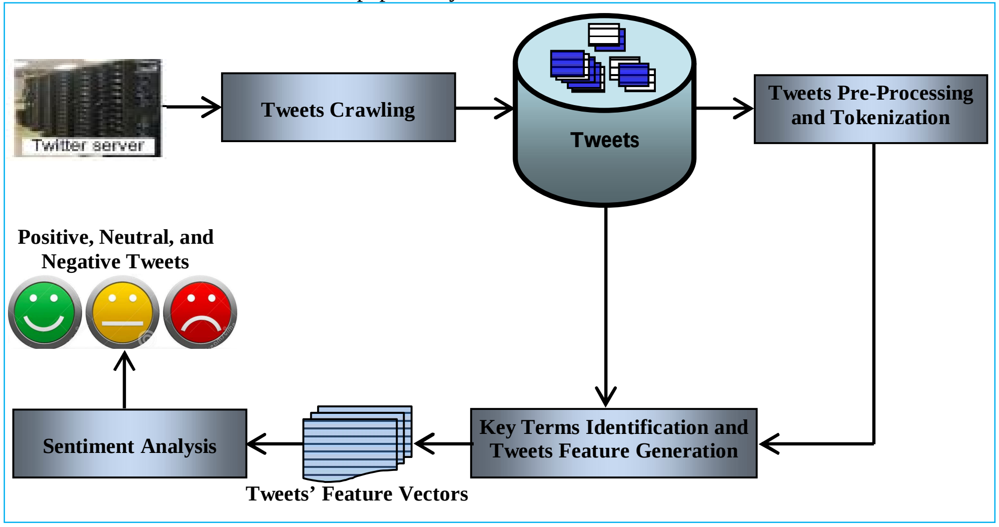

Twitter server
Tweets Crawling
Tweets
Tweets Pre-Processing and Tokenization
Positive, Neutral, and Negative Tweets
Sentiment Analysis
Tweets' Feature Vectors
Key Terms Identification and Tweets Feature Generation

Figure 1. System architecture

3.1 Tweets Crawling

A tweet crawling is focused to download the tweets from twitter database using search key. For this purpose the twitter's API "twitterAj-core-4.02.jar" may be used. It provides a number of classes and methods to fetch tweets' related data as well as users' related data. If we want to download locality and language based tweets it is also possible using this API. The downloaded tweets may be stored in database or text file according to our requirements.

3.2 Tweet Pre-Processing and Tokenization

A Tweet Pre-Processing and Tokenization is focused on filtering the unwanted tokens from the tweets. The words like URLs, mentions, RT, stopwords and the words containing special symbols are filtered out from the tweets and the remaining text of the tweets are splits into tokens based on blank space and punctuation mark, and form a bag of words.

3.3 Key Terms Identification and Tweets Feature Vector Generation

A Key terms identification and tweets feature generation is focused on modelling the each tweet in an n dimensional feature vectors. Every token of a tweet obtained from previous process is considered as a candidate term. Once the list of candidate terms from each tweet is identified, the set of tweets is converted into a term-tweet matrix A of order m x n. In this matrix, a row represents a candidate term and a column represents a tweet. The element $a_{i,j}$ of matrix A, determined as the weight of term t_i in j^{th} tweet using tf-idf method, which is calculated using equations 1 and 2.

$$a_{i,j} = tf(t_{i,j}) \times idf(t_i) \tag{1}$$

$$idf(t_i) = \log \frac{n}{\left|\{tw_j : t_i \in tw_;\}\right|} + 1 \tag{2}$$

Since the term-tweet matrix is generally sparse matrix and the dimension of the term as well as tweet vectors are large, we apply Singular Value Decomposition (SVD) to map the feature set into a low-dimensional space [9]. This increases the efficiency of the proposed system both in terms of memory and execution time requirements. For a given m x n matrix with m ≥ n, the SVD decomposes it into an m x n orthogonal matrix U, an n x n diagonal matrix S, and an n x n orthogonal matrix V such that A = USV' . In this decomposition, U represents the term matrix and V represents the tweet matrix. Each row of matrix V represents a tweet vector whose dimension is reduced from m to n in the new feature space. Based on matrix V, we grouped tweets into a number of clusters that is used to construct the input file for LDA.

In order to score the candidate term, we use Latent Dirichlet Allocation (LDA). LDA is a generative probabilistic model in which documents are represented as random mixtures over latent topics characterized by a distribution over words [10]. The input file for LDA is generated using the clusters of the tweets. In this file, the first line contains an integer value k representing the number of clusters (number of documents for LDA). Followed by this, there are k paragraphs; one for each cluster, containing the list of terms obtained from the corresponding tweets belongs to that cluster. We have used JGibbLDA to execute LDA on this dataset to get Θ and Φ matrices. We have sets the Dirichlet hyper parameters α and β as 0.1 and 0.5, respectively for of LDA execution. The elements of the Φ matrix represent the term-topic distributions and element of the Θ matrix represent the topic-cluster distributions. We use Θ and Φ matrices to assign a ranking score to each term using equations 3 and 4 in which is number of terms in i^{th} paragraph of the LDA input file, n is the number of topics (we have taken n = 100), and k is the number of clusters. After calculating the score of each term, we arrange them in decreasing order of their scores and identify top n-terms as key terms. For further detail see our previous work [11].

$$score(t_i) = \max{}_{j=1}^{n}\{\Phi_{j,i} \times \omega_j\} \qquad (3)$$

$$\omega_j = \sum_{i=1}^{k} n_i \times \Theta_{i,j} \qquad (4)$$

Thereafter, each tweet is modelled as an n-dimensional binary feature vector where each element will be either 1 or 0 depending on presence and absent of term in the tweet. These tweet feature vectors are used for training and testing of sentiment classifier.

3.4 Sentiment Analysis

The binary feature vectors of the tweets are used as input for sentiment analysis. We generate input file for sentiment classifier using these feature vectors. The Weka's Naive Bayes classifiers are used to classify the tweets as positive, negative, or neutral tweet depending upon their text.

Naive Bayes classifier is a probabilistic classifiers based on Bayes' theorem. If s is the sentiment of a given tweet T then probability that sentiment of the tweet is s is defined by using Bayes theorem as equation 5.

$$P(s \mid T) = \frac{P(s).P(T \mid s)}{P(T)} \qquad (5)$$

4. EXPERIMENTAL SETUP AND RESULTS

In this section, we present our experimental setup and results. For evaluation of the system, we downloaded 3100 tweets on three different events "Gaza under attack", "Delhi Assembly Election 2015", and "Union Budget 2015" using Twitter's API. The statistics of the downloaded tweets as shown in table 1. The sentiments of each tweet as positive, negative, or neutral are assigned by expert based on their message. The identification of key terms is very important task in this system. We applied LDA to assign a numeric score to each word of the tweets, and then arranged them in descending order of their scores. The top 80 terms with their LDA score are shown in table 2. Total 6600 key terms is identified after filtering the stop word and invalid tokens from these 3100 tweets.

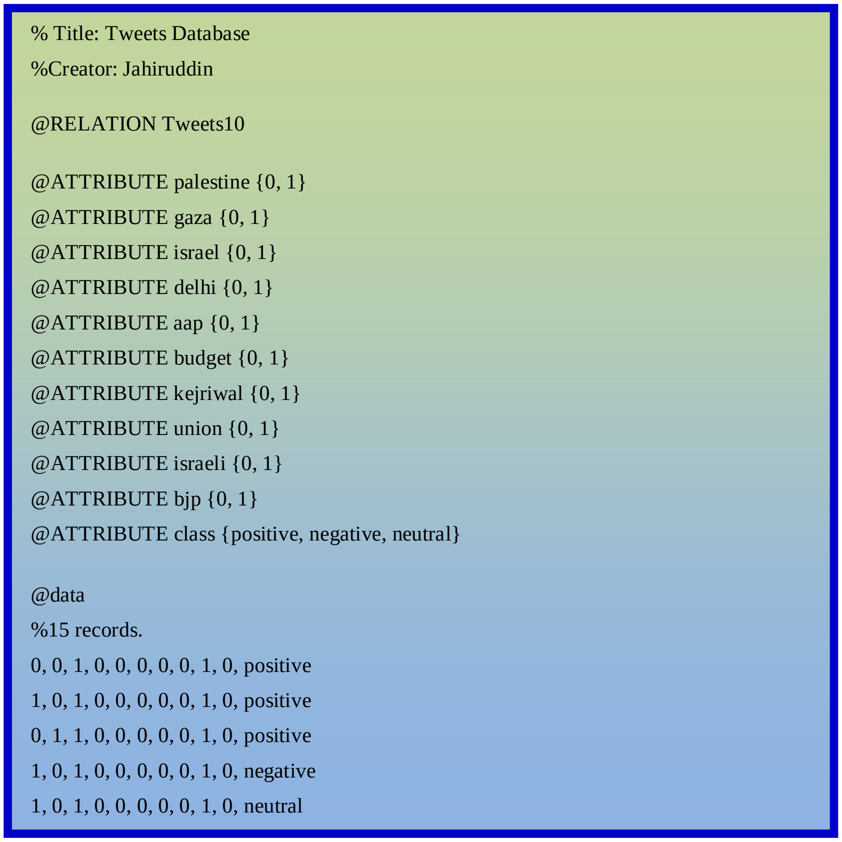
The feature vectors of the tweets are generated using these key terms. We have taken top n key terms for generating the input file for Weka to train and test the sentiment analysis classifier. We have generated input files for Weka by taking top 1000, 2000, 3000, 4000, 5000, 6000, and 6600 key terms for evaluation of our system. For generating the input file for Weka, we have written a program in Java that accept the number of key terms and number of tweets and it generate the input file for Weka based of tweet text and its sentiment. The number of attributes in the database is depends on the number of key terms and the number of records is equal to number of tweets. The database for 10 key terms and 15 tweets as input file for Weka is shown in figure 2.

Table 1. Tweets's data set statistics

Tweet Category	Tweets' Statistics				Users' Statistics		
	No. of tweets	Avg. no. of hash tags	Avg. no. of URLs	Avg. no. of mention	Avg. no. of followers	Avg. No. of friends	Avg. no. tweets
Gaza under attack	1500	1.3	0.37	0.95	2104.4	1093.84	18865.53
Delhi Assembly Election 2015	900	0.32	0.49	1.03	2352.48	600.97	29707.23
Union Budget 2015	700	0.98	0.71	0.83	1597.59	973.84	28244.1
Grand Total	**3100**	**0.94**	**0.48**	**0.95**	**2061.99**	**923.65**	**24130.86**

Table 2. Key terms and their LDA score

Key Terms	LDA Score	Key Terms	LDA Score	Key Terms	LDA Score	Key Terms	LDA Score
palestine	585.12	fatwa	63.99	arvindkejriwal	39.13	sarkar	26.69
gaza	481.05	modi	63.26	timesofindia	38.40	terrorists	26.61
israel	393.66	blame	63.26	jews	36.94	responsibility	26.61
delhi	331.67	children	63.16	peace	36.94	jaitley	26.33
aap	299.49	election	58.14	freepalestine	36.15	soldiers	25.82
budget	268.59	reasons	57.41	kill	35.35	support	25.82
kejriwal	198.56	hammas	56.80	rockets	33.76	minister	25.71
union	188.05	free	56.01	hospital	32.97	media	25.03
israeli	147.37	stop	55.22	world	32.18	military	25.03
bjp	138.59	people	53.63	kiski	31.08	mahmoud	24.23
hamas	132.28	introspect	53.02	president	30.59	highlights	23.85
bedi	96.91	conflict	52.83	results	30.35	killing	23.44
palestinian	95.73	civilians	47.27	illegal	29.79	abbas	23.44
gazaunderattack	90.17	occupation	47.27	day	28.20	voted	23.04
kiran	87.40	india	46.78	syria	28.20	elections	23.04
unionbudget2015	84.57	war	44.89	pray	27.41	aapsweep	23.04
arvind	72.77	human	44.09	corporate	27.41	prayforgaza	22.64
killed	67.13	don	40.59	uphold	27.41	attack	22.64
loss	65.46	palestinians	40.12	chief	26.69	injured	22.64
dilli	65.46	polls	39.86	victory	26.69	tax	22.62



```
% Title: Tweets Database
%Creator: Jahiruddin

@RELATION Tweets10

@ATTRIBUTE palestine {0, 1}
@ATTRIBUTE gaza {0, 1}
@ATTRIBUTE israel {0, 1}
@ATTRIBUTE delhi {0, 1}
@ATTRIBUTE aap {0, 1}
@ATTRIBUTE budget {0, 1}
@ATTRIBUTE kejriwal {0, 1}
@ATTRIBUTE union {0, 1}
@ATTRIBUTE israeli {0, 1}
@ATTRIBUTE bjp {0, 1}
@ATTRIBUTE class {positive, negative, neutral}

@data
%15 records.
0, 0, 1, 0, 0, 0, 0, 0, 1, 0, positive
1, 0, 1, 0, 0, 0, 0, 0, 1, 0, positive
0, 1, 1, 0, 0, 0, 0, 0, 1, 0, positive
1, 0, 1, 0, 0, 0, 0, 0, 1, 0, negative
1, 0, 1, 0, 0, 0, 0, 0, 1, 0, neutral
```

Figure 2. Tweets Database for Weka

The performance of the method is analyzed by taking into account the overall performance of the system. The aim of the sentiment analysis process is to classify the tweet into positive, negative, and neutral class based of their tweet text. A tweet is said to be correctly classified if its class is same as assigned by expert. The method was evaluated for its recall, precision, and F-measure values for different number of features (key terms). Recall, precision, and F-measure for this purpose are defined by equations 6 to 8 respectively.

Precision (π): the ratio of true positives among all retrieved instances

$$\pi = \frac{TP}{TP + FP} \tag{6}$$

Recall (ρ): the ratio of true positives among all positive instances

$$\rho = \frac{TP}{TP + FN} \tag{7}$$

F-measure (F): the harmonic mean of recall and precision

$$F = \frac{2\rho\pi}{\rho + \pi} \tag{8}$$

The Weka databases, generated with different number of key terms, are classified by Weka's Naive Bayes classifier with taking 10 fold. The average precision, recall, and F-measure of the classifier are shown in table 3 and corresponding graph is shown in figure 3. From table it is clear that it give the best performance if we take about 1/3 of the total key terms as feature attributes. In our case it is best for n=2000 key terms.

Table 3. Evaluation summary of the system

No. of Terms	TP Rate	FP Rate	Precision	Recall	F-Measure
1000	0.688	0.213	0.689	0.688	0.687
2000	0.707	0.207	0.706	0.707	0.705
3000	0.705	0.215	0.702	0.705	0.702
4000	0.702	0.215	0.698	0.702	0.699
5000	0.701	0.214	0.698	0.701	0.698
6000	0.708	0.218	0.703	0.708	0.703
6600	0.706	0.221	0.701	0.706	0.701

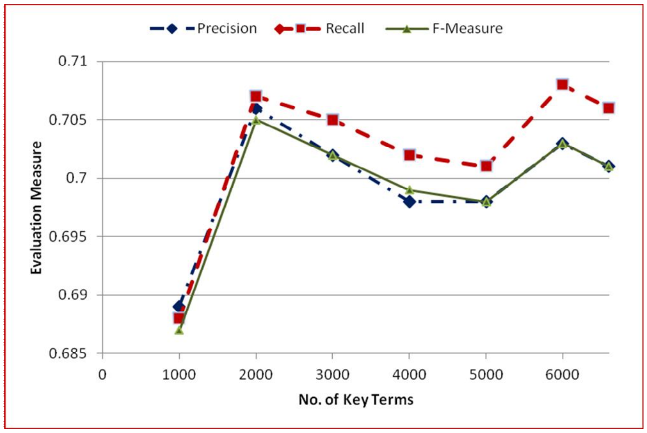


Figure 3. Precision, recall, and F-measure for different value of no. of terms

5. CONCLUSION AND FUTURE WORK

In this paper, we have presented sentiment analysis system to classify the tweets in positive, negative, and neutral sentimental tweets based on their tweet text. First step of the system is key terms identification. We have used LDA method to identify key terms and arranged them in descending order of their LDA score. Then we generated the feature vectors of each tweets by taking top n key terms as attributes. Each tweet is converted into binary feature vector. Then, we used Weka's Naïve Bayes classifier to train the system. The evaluation result show the our system give the best result as 70.6% precision, 70.7% recall, and 70.5% F-measure by taking 1/3 of total number of identified key terms as attributes. In future we are planning to improve the performance of system with some additional features and by taking large set of data.

REFERENCES

[1] B. Pang and L. Lee, "Opinion mining and sentiment analysis," Foundations and Trends in Information Retrieval, vol. 2, no. 1-2, pp. 1–135, 2008.

[2] C. Yang, K. H.-Y. Lin, and H.-H. Chen, "Emotion classification using web blog corpora," in Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence, 2007, pp. 275–278.

[3] J. Read, "Using emoticons to reduce dependency in machine learning techniques for sentiment classification," in Proceedings of the ACL Student Research Workshop, Ann Arbor, Michigan, 2005, pp. 43–48.

[4] A. Go, L. Huang, and R. Bhayani, "Twitter sentiment analysis," Stanford University, Stanford, California, USA, CS224N - Final Project Report, 2009.

[5] H. Becker, M. Naaman, and L. Gravano, "Learning similarity metrics for event identification in social media," in Proceedings of the third ACM international conference on Web search and data mining, 2010, pp. 291–300.

[6] T. Sakaki, M. Okazaki, and Y. Matsuo, "Earthquake shakes twitter users: Real-time event detection by social sensors," in Proceedings of the 19th international conference on World Wide Web, 2010, pp. 851–860.

[7] J. Sankaranarayanan, H. Samet, B. E. Teitler, M. D. Lieberman, and J. Sperling, "witterstand: news in tweets," in Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, 2009, pp. 42–51.

[8] H. Becker, M. Naaman, and L. Gravano, "Beyond trending topics: Real-world event identification on twitter," in Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media, 2011, pp. 438–441.

[9] G.E. Forsythe, M.A. Malcolm, and C.B. Moler, Computer Methods for Mathematical Computations, Prentice Hall Professional Technical Reference, ISBN:0131653326, 1977.

[10] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," Journal of Machine Learning Research, vol. 3, no. 4-5, pp. 993–1022, 2003.

[11] M. Abulaish, Jahiruddin, and L. Dey, "Deep text mining for automatic keyphrase extraction from text documents," Journal of Intelligent Systems, vol. 20, no. 4, pp. 327–351, 2011.