

 Open access • Proceedings Article • DOI:10.1145/2567688.2567694

## Theme Based Clustering of Tweets — [Source link](#)

[Rudra M. Tripathy](#), [Shashank Sharma](#), [Sachindra Joshi](#), [Sameep Mehta](#) ...+1 more authors

**Institutions:** [Silicon Institute of Technology](#), [Indian Institute of Technology Delhi](#), [IBM](#)

**Published on:** 21 Mar 2014

**Topics:** [Cluster analysis](#) and [Document clustering](#)

Related papers:

- [INTWEEMS: a framework for incremental clustering of tweet streams](#)
- [Topic Based Temporal Generative Short Text Clustering](#)
- [A graph based clustering technique for tweet summarization](#)
- [Twitter Topic Summarization by Ranking Tweets using Social Influence and Content Quality](#)
- [Clustering and topic modeling over tweets: A comparison over a health dataset](#)

Share this paper:    

View more about this paper here: <https://typeset.io/papers/theme-based-clustering-of-tweets-1155frj9lv>

# Theme Based Clustering of Tweets

Rudra M. Tripathy  
Silicon Institute of Technology  
Bhubaneswar, India

Shashank Sharma  
IIT Delhi  
New Delhi, India

Sachindra Joshi  
IBM Research-India  
New Delhi, India

Sameep Mehta  
IBM Research-India  
New Delhi, India

Amitabha Bagchi  
IIT Delhi  
New Delhi, India

## ABSTRACT

In this paper, we present overview of our approach for clustering tweets. Due to short text of tweets, traditional text clustering mechanisms alone may not produce optimal results. We believe that there is an underlying theme/topic present in majority of tweets which is evident in growing usage of hashtag feature in the Twitter network. Clustering tweets based on these themes seems a more natural way for grouping. We propose to use Wikipedia topic taxonomy to discover the themes from the tweets and use the themes along with traditional word based similarity metric for clustering. We show some of our initial results to demonstrate the effectiveness of our approach.

## Keywords

Clustering, Social Networks, Twitter, Wikipedia

## 1. INTRODUCTION

Microblogging services like Twitter have become a very important medium for the dissemination of ideas, news and opinions as well as a platform for marketing, sales and public relations. These services have also emerged as an important source of real-time news updates for crisis situations, such as the Mumbai terror attacks or Iran protests.

Clustering tweets is an extremely challenging task primarily due to very short text and non conformance to grammatical rules. We propose a novel clustering algorithm which simultaneously takes into account the words which form the tweets as well as the underlying theme which is present in the tweets. First, each tweet is mapped to a set of wikipedia topics. The distance between two tweets is computed by graph distance on wikipedia topic graph. This metric helps us to capture the closeness of topics or underlying theme between the tweets. Next, the distance between the words in tweets is computed. This helps us to take into account the short forms, spelling mistakes (for same word) into account. A weighted combination of both metrics is used as final distance metric for the clustering algorithm.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to publish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

Copyright 20XX ACM X-XXXXX-XX-X/XX/XX ...\$10.00.

## 2. RELATED WORK

One way to identify the topics in the tweets, is by extracting the topic from its content which requires to use document clustering techniques. To cluster tweets, most of the literatures use clustering techniques based on bag-of-words concept. In the paper [6], Ramage et al. characterize the tweets using labeled LDA methods. Cataldi et al. [1] try to identify emerging topics in the Twitter network based on term frequency and users authority. The set of emerging topics are found by creating topic graphs which links the emerging terms with their relative co-occurrent terms. In a similar study Mathioudakis et al. [4] identify the trending topics using bursty keywords and their co-occurrences. In [8] Weng et al. find the influential users in Twitter by taking both topical similarity and link structure between the users. For topic identification they use Latent Dirichlet Allocation (LDA) which uses bag of words concept. Chen et al. [2] study the problem of recommending tweets using different approach, one of them is based on topic. For topic identification they use TFIDF technique which also used bag of words. The measure issue in these studies which make clustering based on bag of words is the Twitter data is sparse, because the tweets are limited to only 140 characters and are not structured. So it is better to use the concept of the tweets rather than words.

One of the solution to this problem is to map each word to a concept by leveraging the Wikipedia as a knowledge based. Michelson et al. [5] leverage the Wikipedia as a knowledge base to identify users' topical interest. The authors map each word to a category of Wikipedia and called as entity. Comparing to our work, we use clustering technique based on the graph distance of the whole tweets, not on the tweets of a particular user. In a similar work, Genc et al. [3] use Wikipedia based classification technique for tweets categorization. The metric they use for classification is semantic distance, i.e., the distance between their closest Wikipedia pages. The primary difference between this work with our work are, Genc et al. use each word to find its Wikipedia pages, whereas we have an adaptive method which starts from bigrams and gradually move to unigrams, if needed. Moreover, we consider the word frequency also to aid in clustering. Finally, we present results on large Tweets dataset as compared to small set used by Genc et al.

## 3. METHODS

Due to short length of tweets (maximum 140 characters), tweets don't follow any structure. Therefore, standard document clustering techniques based on distribution of words

fail to make proper clusters. In this work we propose to use wikipedia as external knowledge base to cluster the tweets and compute the cluster centers. These cluster centers act as representation for the tweets in the cluster. The key steps of our methods are:

### 3.1 Data Representation

We use Wikipedia taxonomy graph to map each tweet into a set of Wikipedia nodes, where each node in Wikipedia taxonomy graph represents a Wikipedia page and an edge between two nodes  $i$  to  $j$  represents relationship between two pages.

From each tweet, first we remove all the stop words using the list given in [7]. After removing the stop words, we construct bi-grams (say.  $w_1$  and  $w_2$ ) and search if there is any Wikipage assigned for the bigram. If such a page is found then the two words are appended by the neighbors of the corresponding node in the Wikipedia graph. Please note this appending operation provides the context and hierarchy to the tweets. For example, Roger Federer will be appended by Tennis etc. If the bi-gram cannot be mapped to the wikipage then we consider individual unigrams and follow the same procedure. A sliding window protocol used to compute bigrams. Therefore, a tweet with  $m$  words requires maximum  $2 * (m - 1)$  number of comparison. In a smaller dataset, we evaluated the performance of trigrams and found that the tri-grams provide marginal improvement in the quality while increasing the computational cost. We explain this step using Figure 1.

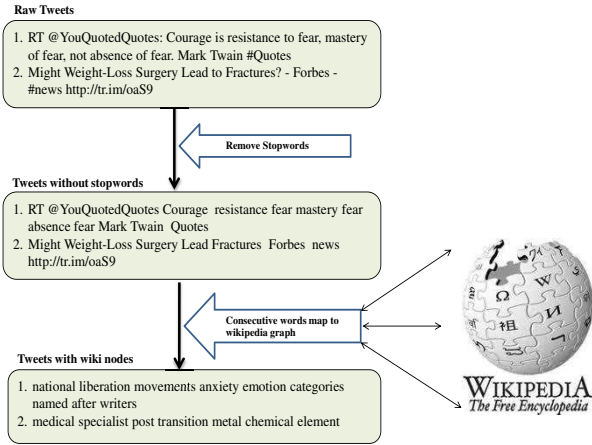


Figure 1: Tweets to Wikipedia nodes

Twitter allows maximum 140 characters in tweets. Therefore, people generally use short words instead of actual words to express their thoughts. For example the short words like Twaffic for *Twitter traffic*, clk for *click*, chk for *check* etc are used. If we search these words in the Wikipedia page we may not be found any page, thereby we loose the information of these words. But these words contribute significantly towards the meaning of the tweets. Therefore, we also consider occurrence of each words in the tweets. We represent each tweet as a vector of their words frequencies.

**Distance measure:** Since we represent each tweet in two ways: Wikipedia representation and words frequency representation, we use two distance measures for clustering.

One is Wikipedia graph distance measure and cosine similarity measure. The graph distance measure is used for the Wikipedia representation of tweets and the cosine similarity measure is used for the word frequency representation of tweets.

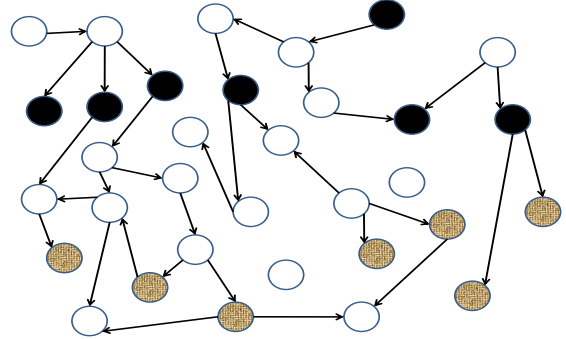


Figure 2: Graph Distance

The graph distance between two tweets is defined as the minimum distance between any two nodes between the tweets. Figure 2 shows an example, how the graph distance is computed. Let the black nodes represent the representative nodes for tweet-1 and checked nodes represent the representative nodes for tweet-2. The graph distance between tweets-1 and tweet-2 is the minimum distance between the nodes of tweet-1 and nodes of tweet-2. The graph distance measure is 1 in this example.

**Cluster Efficiency:** Clusters produced by a algorithm are efficient, if the tweets present inside a cluster are related. Since there are no tools available which can say whether two tweets are related or not, we do an user study to compare tweets among the clusters.

### 3.2 Mathematical Formulation

Let  $T = \{t_1, t_2, t_3, \dots, t_N\}$  be the set of  $N$  tweets. We represent each  $t_i$  using Wikipedia representation as well as words frequency representation. Let  $WI$  is the set of Wikipedia pages where,  $|WI| = n_1$  and  $WO$  is the set of words where,  $|WO| = n_2$ .

Suppose  $M_{WIKI}$  be the matrix which is used for Wikipedia representation of tweets. The order of the matrix is  $N \times n_1$ .  $M_{WIKI}$  is a Boolean sparse matrix and is defined as follows:

$$M_{WIKI_{i,j}} = \begin{cases} 1 & \text{if } (WI_j \text{ is neighbor of any} \\ & \text{Wikipage assigned to tweets} \\ & t_i \text{ using bigram approach)} \\ 0 & \text{otherwise} \end{cases}$$

In a similar way let  $M_{WORD}$  be the corresponding matrix for words frequency representation of tweets. The order of  $M_{WORD}$  is  $N \times n_2$  and is defined as:

$$M_{WORD_{i,j}} = \begin{cases} Freq(WO_j) & \text{if } (WO_j \in t_i) \\ 0 & \text{otherwise} \end{cases}$$

The distance measure between two tweets  $t_i$  and  $t_j$  is defined as:

$$dist(t_i, t_j) = \alpha * Gdist(t_i, t_j) + \beta * Wdist(t_i, t_j)$$

where  $\alpha$  and  $\beta$  are two adjustable parameter and  $\alpha + \beta = 1$ . The value of the parameters make biased towards one measure. In all our experimentation we have considered  $\alpha$



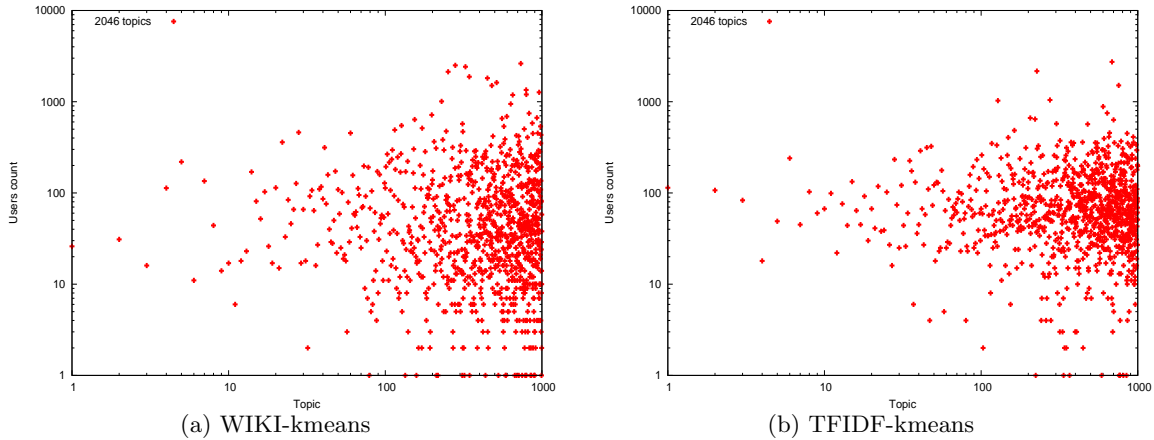


Figure 3: Distribution of Clusters points

Wikipages. This can be done by looking at the cluster centers of the Wikipedia representation of tweets. The cluster center for each cluster is constructed as follows:

1. for each  $WI_i \in WI$ , computer the frequency of  $WI_i$  in that cluster, i.e., how many tweets use  $WI_i$  in its representation.
2. Choose  $WI_i$  as a node in the cluster center if it is used by more than 10% of tweets.

That is the most commonly used Wikipages in the tweets for a cluster, are considered for constructing the center. The cluster center for the cluster, whose tweets cloud (Figure 4) is shown above, is given below:

Former Olympic sports, Outdoor recreation, Ball games, Team sports, Olympic sports, Precision sports, Individual sports.

The cluster centers of WIKI-kmeans are very closely related to the semantic of the tweets assigned to that cluster. This we think is one of the important benefit of our algorithm. We can use only the cluster center to give a broad topics to the tweets assigned to that cluster. Each node in the cluster center can be used to give a first level topic to the tweets which use that node in their representation. To refine their topic one could think to apply multi-level clustering. For example tweets under *Team sport* can be passed through a clustering technique to assign topics like *Football, Cricket, Hockey, etc..*

### 4.3 Validation

We compare our algorithm which is based on two ideas: frequencies of words and Wikipedia mapping with a well known algorithm TFIDF (Term Frequency Inverse Document Frequency) for document clustering. TFIDF is based on word frequency. Validation we mean, cluster validation, i.e., whether the tweets in a cluster are related or not. Since there is no training dataset available for the tweets, it is very difficult to validate the clustering algorithm. Therefore, we have decided to conduct an user-study to validate the clusters.

We conducted a user study on the results we got from both the clustering algorithms to get an unbiased result on the quality of the cluster. For conducting a user study (survey), we set up two web pages: one for the TFIDF-kmeans

clustering algorithm and other one for the WIKI-kmeans clustering algorithm, in such a way that in each web page, there are six pairs of tweets. In each pair, there are two tweets and the pairs are made in such a way that tweets from the first three pairs belong from the same cluster and tweets from the next three pairs belong from the different cluster. In that survey, we ask from the user that whether the two tweets from each pair are related or not. If user feels that the tweets in the pair are related, then, he can answer “Yes”, if not, then, he can answer “No” or else, he can also answer “Maybe” if he is not sure whether tweets in the pair are related or not. We recorded the answers filled by the users in the database to calculate the F-score of the clustering algorithm. To calculate F-score, we use the following formula:

$$F\text{-score} = 2 * \frac{Precision * Recall}{Precision + Recall}$$

So, to calculate F-score, we need to calculate Precision and Recall, and to calculate Precision and Recall we need to know the following 4 parameters:

- True Positive Results (tp): Tweets in a pair are related and from same cluster
- True Negative Results (tn): Tweets in a pair are unrelated and from different cluster
- False Positive Results (fp): Tweets in a pair are unrelated and from same cluster
- False Negative results (fn): Tweets in a pair are unrelated and from same cluster

By looking into the database of the survey we have conducted, we can easily find tp, tn, fp and fn, and using them, we can calculate Precision and Recall using the following formulas:

$$Precision = \frac{tp}{tp + fp}$$

$$Recall = \frac{tp}{tp + fn}$$

We find that F-score of the TFIDF-kmeans clustering algorithm is 0.438, whereas F-score of WIKI-kmeans clustering algorithm is 0.523, which proved that clustering performed by our clustering algorithm, that is, WIKI-kmeans clustering algorithm is better than that of TFIDF-kmeans clustering algorithm.

## 5. CONCLUSIONS

To cluster the unstructured and sparse documents, tweets, we proposed a clustering technique which is based on words frequencies and Wikipedia mapping of the tweets. We have found that our proposed algorithm outperform the algorithm which is only based on words frequencies.

We found that the cluster center for each cluster gives a semantic meaning to that cluster. The events/topics involved in all the tweets for a particular cluster intuitively represent the same events that of center. Therefore, these cluster centers can be used to give broad category to the tweets in that cluster.

Currently, we are working multi-label clustering in stead of single label. The first label cluster centers can be used as the first level topics and so on, which helps to refine the topics quality.

## 6. REFERENCES

- [1] M. Cataldi, L. Di Caro, and C. Schifanella. Emerging topic detection on twitter based on temporal and social terms evaluation. In *Proceedings of the Tenth International Workshop on Multimedia Data Mining, MDMKDD '10*, pages 4:1–4:10, New York, NY, USA, 2010. ACM.
- [2] J. Chen, R. Nairn, L. Nelson, M. Bernstein, and E. Chi. Short and tweet: experiments on recommending content from information streams. In *Proceedings of the 28th international conference on Human factors in computing systems, CHI '10*, pages 1185–1194, New York, NY, USA, 2010. ACM.
- [3] Y. Genc, Y. Sakamoto, and J. V. Nickerson. Discovering context: Classifying tweets through a semantic transform based on wikipedia. *Lecture Notes in Computer Science: Human Computer Interaction International*, 2011.
- [4] M. Mathioudakis and N. Koudas. Twittermonitor: trend detection over the twitter stream. *Proceedings of the 2010 international conference on Management of data*, pages 1155–1157, 2010.
- [5] M. Michelson and S. A. Macskassy. Discovering users' topics of interest on twitter: a first look. In *Proceedings of the fourth workshop on Analytics for noisy unstructured text data, AND '10*, pages 73–80, New York, NY, USA, 2010. ACM.
- [6] D. Ramage, S. Dumais, and D. Liebling. Characterizing Microblogs with Topic Models. In *ICWSM*, 2010.
- [7] G. Salton and C. Buckley. Onix text retrieval toolkit. <http://www.lextek.com/manuals/onix/stopwords2.html>.
- [8] J. Weng, E. Lim, J. Jiang, and Q. He. Twitterrank: finding topic-sensitive influential twitterers. In *Proceedings of the third ACM international conference on Web search and data mining, WSDM '10*, pages 261–270, New York, NY, USA, 2010. ACM.
- [9] J. Yang and J. Leskovec. Patterns of temporal variation in online media. In *Proceedings of the fourth ACM international conference on Web search and data mining, WSDM '11*, pages 177–186, New York, NY, USA, 2011. ACM.