Clustering tweets using Wikipedia concepts

Guoyu Tang[†], Yunqing Xia[†], Weizhi Wang[‡], Raymond Lau[*], Thomas Fang Zheng [†]

[†] Tsinghua University
Haidian, Beijing 100084, China
{tgy09, yqxia, fzheng}@tsinghua.edu.cn

[‡]University of Southern California
Los Angeles, CA 90089, USA
weizhiwa@usc.edu

[*]City University of Hong Kong
Tat Chee Avenue, Kowloon, HK SAR
raylau@cityu.edu.hk

Abstract

Two challenging issues are notable in tweet clustering. Firstly, the sparse data problem is serious since no tweet can be longer than 140 characters. Secondly, synonymy and polysemy are rather common because users intend to present a unique meaning with a great number of manners in tweets. Enlightened by the recent research which indicates Wikipedia is promising in representing text, we exploit Wikipedia concepts in representing tweets with concept vectors. We address the polysemy issue with a Bayesian model, and the synonymy issue by exploiting the Wikipedia redirections. To further alleviate the sparse data problem, we further make use of three types of out-links in Wikipedia. Evaluation on a twitter dataset shows that the concept model outperforms the traditional VSM model in tweet clustering.

Keywords: Tweet clustering, tweet representation, Wikipedia concept

1. Motivation

Twitter brings people huge convenience by providing an instant, effective and convenient platform for both social communication and information acquisition. On the other hand, an increasing number of tweets are created every day in twitter. The problem of information overload has never been as serious as today. Manually maintaining the subscribed tweets is very laborious. In this work, we target at developing an effective tweet clustering system.

Two challenging issues are notable in tweet clustering. Firstly, the sparse data problem is serious in tweets. No tweet can be longer than 140 characters due to application restriction. Thus representing the tweets becomes very challenging. Secondly, synonymy and polysemy are more common in twitter than that in formal text. Due to diversified background, users intend to present a unique meaning with a great number of manners in tweets. This is linguistically referred to as synonymy. For example, *antenna* and *aerial* refer to same thing in most cases. On the other hand, words tend to hold different meanings in different tweets, which is defined as polysemy in linguistics. The typical example is *apple*, which sometimes refers to a kind of fruit while a US company in other cases.

The traditional vector space model (VSM) (Salton et al., 1974) views terms as words as features and converts each document into a word vector. As it is assumed that words are independent of each other, synonymy and polysemy cannot be dealt with. In the following research, the assumption was discarded and latent semantic analysis (LSA) (Deerwester et al., 1990) was developed to organize semantically similar words with a latent variable. Similarly, Latent Dirichlet allocation (LDA) (Blei et al., 2003) is later designed to manage words with topics. The common drawback of the above work is that semantic information is statistically discovered and quality of the information is completely determined by the training data. Since 2006, there arose a large number of work attempting to cluster tweets using the general methods that has been proved successful on news wire. Research in (Phan et al., 2008; Ritter et al., 2010; Ramage et al., 2010; Karandikar, 2010) indicates that highly related Twitter messages often have very little overlapping on the word level. To extend the tweets, Web is used in (Sahami and Heilman, 2006) as a source of additional knowledge for measuring similarity of short text snippets. It has also been confirmed that short text clustering can be improved by resolving synonyms with WordNet concepts (Hotho et al., 2003). Very recently, research indicates that Wikipedia is promising in representing text with concepts that are finely compiled by human (Gabrilovich and Markovitch, 2006; Banerjee et al., 2007; Hu et al., 2008; Spanakis et al., 2012).

Enlightened by the recent research efforts, we study Wikipedia concepts and attempt to represent tweets using Wikipedia concepts in this work. We address the polysemy issue with a Bayesian model, and the synonymy issue with algorithm based on Wikipedia *redirections*. To further alleviate the sparse data problem, we further make use of three types of *out-links* in Wikipedia. Evaluation on a twitter dataset shows that the concept model outperforms the

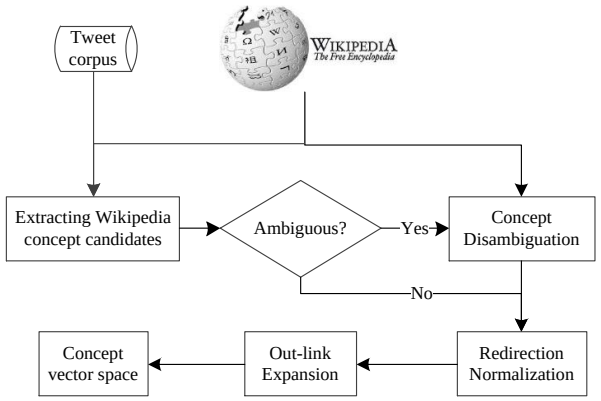

Figure 1: The proposed Wikipedia based tweet clustering method.

- *article*: a web page describing a single topic.
- *concept*: the title of Wikipedia articles
- *redirection* of a concept c: a concept redirected from the concept c
- *out-link* of a concept *c*: a concept (o) if there is a hyperlink from the corre-sponding articles of the concept c to the articles of concept *o*
- *definition* of a concept *c*: the first sentence of the corresponding article of concept c which is always an explanation to the concept
- *categories* of a concept *c*: the list of categories which corresponding article belong to.

Figure 2: Some definitions of Wikipedia.

traditional VSM model and LDA model in tweet clustering. The following findings are interesting: First, *redirections* in Wikipdia are very useful in resolving synonyms. Second, concept definition in Wikipedia is sufficient in polysemous word disambiguation. Third, related concepts in Wikipedia can help enrich tweet representation greatly.

2. Method

2.1. Architecture

The framework of the proposed method is presented in Figure 1. We first extract candidate Wikipedia phrases from tweets (see Section 2.3.). Then we disambiguate Wikipedia concepts with a Bayesian model so that each candidate phrase is mapped to one Wikipedia concept (see Section 2.4.). In what follows, we normalize the synonymous concepts to their *redirections* (see Section 2.5.). Finally, we explore three types of *out-links* to include related concepts with three kinds of granularity(see Section 2.6.).

In this work, we focus on tweet representation, which seeks to map each tweet to space of Wikipedia concepts. We follow the mechanism of vector space model. Differently, the features are Wikipedia concepts, and the weight of each concept is calculated in a manner that is similar to TF-IDF (term frequency and inverse document frequency). In our case, we count concept frequency and tweet frequency, referred to as CF-ITF (i.e., concept frequency and inverse tweet frequency). Tweet similarity is measured similarly. We apply the cosine similarity on the two tweet concept vectors.

Clustering algorithm is not our focus. Thus we employ the classic clustering algorithms in tweet clustering, i.e., HAC (Hierarchical Agglomerative Clustering) algorithm (Voorhees, 1986), Bisecting K-Means algorithm (Steinbach et al., 2000), and Graph based clustering algorithms (Steinbach et al., 2000).

2.2. Wikipedia concept

Wikipedia is a dynamic and fast growing resource. Articles about newsworthy events are often added within few days of their occurrence. Figure 2 presents some Wikipedia definitions used in this paper.

2.3. Mapping tweets to Wikipedia concept space

With Wikipedia API, we first collect all the phrases of Wikipedia concepts. Then we follow (Wong and Chan, 1996) to search candidate Wikipedia phrases within tweets. (Wong and Chan, 1996) presents a dictionary-based word segmentation approach: Forward Maximum matching algorithm. It is rather common that ambiguity occurs in the matching procedure. The disambiguation algorithm is given in the following section.

2.4. Concept disambiguation

If a candidate phrase is polysemous, namely, it has multiple meanings, it is necessary to perform word sense disambiguation to find its real meaning in the context. Unlike previous researches (Spanakis et al., 2012; Ferragina and Scaiella, 2010) which choose concept with the highest similarity between context and corresponding Wikipedia article, we propose a Bayesian approach and formalize the disambiguation problem in a generative model. For each ambiguous phrase, we first draw a distribution over concepts and then generate context words according to this distribution. It is thus assumed that different concepts will correspond to distinct lexical distributions. More formally, the context words around the ambiguous target word are first modeled as samples from a multinomial sense distribution, which means the distribution over context words within the context window v of a certain target word w can be specified as follows:

$$p(t|v) = \sum_{c_w} p(t|c_w)p(c_w|v) \quad (1)$$

where t is a context word, c_w, and $p(t|c_w)$ the probability distribution of word t_i under concept c_w . In practice, $p(t_i|c_w)$ reflects the probability that word t_i appears in the context given concept c_w. The goal of the work is to estimate $p(t|c_w)$ and an LDA model is adopted to induce word senses.

Note this model was first used in Word Sense Induction (Brody and Lapata, 2009). But in this paper, we need to disambiguate concept in Wikipedia. Thus we train concept-context words distribution in corresponding articles and

infer concept distribution in tweets. Gibbs sampling is used for parameter estimation and inference (Griffiths and Steyvers, 2004). The values of the hyper-parameters are the same as in (Brody and Lapata, 2009).
For instance, two tweets are given as follows:
T_1: *That man with one arm lost his other limb in an airplane crash*.
T_2: *The nation must arm its soldiers for battle*.
The candidate *arm* refers to several different concepts. Two common used concepts are: 1) *arm* (the upper part of the human upper limb) and 2) *weapon*. The concept-context words distribution in the two corresponding articles are as follows:
arm#1={ *limb*: 0.159, *forelimb*: 0.069, *sleeve*: 0.019 }
arm#2={ *weapon*: 0.116, *war*: 0.039, *battle*: 0.026 }
The probability of concept *arm*#1 in tweet T_1 is 0.998005. For tweet T_2, The probability of concept *arm*#2 is 0.944096.
In this work, we simply take the concept with the highest probability as concept of the candidate and use the concepts to represent document.

2.5. Redirection Normalization

Wikipedia guarantees that there is only one article for each concept by using *Redirect* hyperlink to group equivalent concepts to the preferred one. Synonymy in Wikipedia mainly comes from these redirect links. An example entry with a considerably higher number of redirect pages is United States. Its redirect pages correspond to synonyms (U.S.A., U.S., USA, US and Yankee land). To deal with these synonyms, we map all the synonymous concepts to their redirections. Thus all synonyms (U.S.A., U.S., USA, US and Yankee land) are replaced with United States. Synonymy problem is solved.

2.6. Expanding the tweet representation with relevant concepts

Due to limited length, tweets cannot provide sufficient information to traditional similarity calculation techniques. In this paper, we use relevant concepts from Wikipedia to expand the tweets representation. Each Wikipedia article contains a lot of hyperlinks, which express relatedness between them. In this paper, we use *out-links* of each concept to expand the concept vectors. Three methods are proposed.

- *Method* #1 (*concept expansion based on out-links in the whole articles*): In this method, all out-links in the corresponding article are used to expand the concept vectors. For example, the article of concept *Apple Inc.* contains out-links *iPhone smartphone*, *iPod*, *Apple Store*, *iTunes Strore* et al. In this method, all the concepts in this article are appended to the tweet vector.

- *Method* #2 (*concept expansion based on out-links in the same category*): Some of out-links are not semantic related. For example, in the article of travel, there is a sentence: travel is a movement of people. Concept people is an out-link but it is not semantic related to concept travel in tweet representation. Thus we need to extract related out-links. Category is an important element to reflect sematic of concept. In this method, we use category of concept to filter out unrelated concept. Out-links which do not have one common category with the target concept are filtered out.

- *Method* #3 (*concept expansion based on out-links in the definition*): In this method, with the assumption that out-links in the definitions are closely related to the concept, we only use those out-links to expand the concept vectors.

3. Evaluation

3.1. Setup

Test dataset: To evaluate our tweet clustering techniques, we selected a total of 6 popular hash-tags and extracted 2450 tweets with those tags. In our experiments, we only extract nouns and verbs as feature. We use TreeTagger (Schmid, 1994) to do lemmatization and POS tagging for English word.
Wikipedia: We use the Java Wikipedia Library (JWPL) to process the English Wikipedia dump and obtained 8145917 concepts.
Evaluation criteria: We adopt the evaluation criteria proposed by (Steinbach et al., 2000). The calculation starts from maximum F measure of each cluster. Let A_i represent the set of articles managed in a system-generated cluster c_i , A_j the set of articles managed in a human-generated cluster c_j . F measure of the system-generated cluster c_i is calculated as follows.

$$p_{i,j} = \frac{\|A_i \cap A_j\|}{\|A_i\|} \quad p_i = \max_j\{p_{i,j}\} \tag{2}$$

$$r_{i,j} = \frac{\|A_i \cap A_j\|}{\|A_j\|} \quad r_i = \max_j\{r_{i,j}\} \tag{3}$$

$$f_{i,j} = \frac{2 \cdot p_{i,j} \cdot r_{i,j}}{p_{i,j} + r_{i,j}} \quad f_i = \max_j\{f_{i,j}\} \tag{4}$$

where $p_{i,j}$, $r_{i,j}$ and $f_{i,j}$ represent precision, recall and f measure of cluster when compared with cluster c_j, respectively.

3.2. Experiment 1: Effect of concept disambiguation

In this experiment we aim to study how concept disambiguation influence the system performance. We implemented a system of concept disambiguation method we proposed.

- *Bayesian model with VSM (BM-VSM)*: A system uses the proposed Bayesian model in this work to disambiguate Wikipedia concepts. VSM is used to represent document. Cosine similarity measure is used to calculate document similarity and Bisecting K-means is used to cluster documents. The cluster number is set as 6 in test dataset.

We also implemented two baseline clustering systems:

- *Word with VSM (W-VSM)*: A baseline text clustering system that uses word as features and the classic VSM model is used to represent documents.

- *Wikipedia phrase with VSM (WP-VSM)*: A baseline text clustering system that uses Wikipedia phrase as features and VSM to represent documents. Note that in this system, we just use the candidate phrase without disambiguation.

Experiments results are presented in Table 1.

Method	BM-VSM	W-VSM	WP-VSM
F measure	0.746	0.712	0.734

Table 1: The F measure of tweet clustering with concept disambiguation methods.

Discussion on contribution of Wikipedia phrase: We can see from Table 1 that WP-VSM outperforms W-VSM. This indicates that using Wikipedia phrase as features is better than using word. For example, if a document contains the phrase *data mining*, it is more precise to use the sematic entity than two more ambiguous singletons data and mining.
Discussion on contribution of concept disambiguation: We can see from Table 1 that both BM-VSM performs better than WP-VSM. This indicates that better features can be got through concept disambiguation. The reason is worth noting: after concept disambiguation, a deterministic concept is assigned to every ambiguous Wikipedia candidate phrase in a tweet according to its context which makes document similarity calculation more accurately. For example, there are two tweets as mentioned in Section 2.3. As the word *arm* in two tweets is identified as different concept. The similarity between the two tweets is 0 while in WP-VSM it is higher because of the common word *arm*. In that case, similarity calculation in the concept space is obviously more accurate.

3.3. Experiment2: Effect of redirection normalization

In this experiment we aim to study how redirection normalization influences the system performance. We implement a system of redirection normalization methods based on BM-VSM.

- *Redirection normalization with BM-VSM (RN-BV)*: A system normalizes redirection after Bayesian concept disambiguation. Other setups are the same as BM-VSM.

Experiments results are presented in Table 2.

Method	BM-VSM	RN-BV
F measure	0.746	0.759

Table 2: The F measure of tweet clustering with redirection normalization.

Discussion: We can see from Table 2 that RN-BV performs better than BM-VSM. This indicates dealing with the synonymy problem through redirection normalization can improve the performance. For example, tweet containing *United States* holds a reasonable similarity with tweet containing *U.S.A*. even though they do not contain common word. This is more consistent with the real situation.

3.4. Experiment 3: Different out-link expansion methods

In this experiment we aim to study how different out-link expansion methods influence the system performance. We implement three system of different out-link expansion methods based on RN-BV.

- *Method#1 with RN-BV (M1-RB)*: A system expands tweet vectors with method1 described in Section 3.5 after redirection normalization and Bayesian concept disambiguation. Other setups are the same as RN-BV.
- *Method#2 with RN-BV (M2-RB)*: A system expands tweet vectors with method 2 described in Section 3.5. Other setups are the same as RN-BV.
- *Method#3 with RN-BV (M3-RB)*: A system expands tweet vectors with method 3 described in Section 3.5. Other setups are the same as RN-BV.

Experiments results are presented in Table 3.

Method	RN-BV	M1-RB	M2-RB	M3-RB
F measure	0.759	0.654	0.763	0.826

Table 3: The F measure of tweet clustering with different expansion method.

Discussion: We can see from Table 3 that M1-RB performs worst in three expansion systems and even worse than RN-BV. This is because using all out-links in the corresponding article to expand the tweet vector will include unrelated concept. Performances of M2-RB and M3-RB are better than RN-BV. This means with proper out-links the expansion methods can improve the performance. M3-RB performs better than M2-RB which indicates out-links of M3 is more precise than M2.

3.5. Experiment 4: Different clustering algorithms

In this experiment we aim to study how different clustering algorithms influence the system performance.
To further our model with start-of-art document representation models, We also implemented a LDA (Blei et al., 2003) based system which uses topic-document distribution to represent documents:

- *Word with LDA (W-LDA)*: A text clustering system that perform LDA first and use topic-document distribution as features to represent documents. The number of LDA topic is set to 80 and we set $\alpha = 0.2$ and $\beta = 0.1$ and number of iterations as 2000 according our empirical study .

Table 4 show results with different clustering algorithms
Discussion: Three observations can be found from Table 4. First, our model performs better than W-VSM with all clustering algorithms. This means after concept disambiguation, redirection normalization and out-link expansion, representing tweets in concept space can help tweet clustering in all clustering algorithms. Second, both W-LDA and our

Method	K-Means	HAC	Graph
M3-RB	0.826	0.701	0.727
W-VSM	0.712	0.547	0.608
W-LDA	0.768	0.623	0.704

Table 4: The F measure of tweet clustering with different expansion method.

model perform better than W-VSM. W-LDA captures latent semantic information from data while our model use concepts from Wikipedia. This means both latent and explicit sematic information are helpful for tweets clustering. Third,our model performs better than W-LDA. This means representing tweets in concept space from our model is better than in latent semantic space from traditional LDA for tweet clustering. The reason is as follows: Tweets are too sparse for LDA which will affect the accuracy of topics while our model expands tweets by means of Wikipedia to alleviate the sparse data problem.

4. Conclusion

In this work, we use concept vector to represent tweets. Different from the previous researches which use similarity between documents and Wikipedia article, we use a Bayesian model to disambiguate words in tweets. We further address synonymous concepts with their redirections and propose out-link expansion methods to expand the concept vectors. Three conclusions are drawn from the experiment results. Firstly, the Bayesian concept disambiguation model is helpful for tweets clustering . Secondly, out-links in the definitions helps to effectively expand the tweet representation. At last, our model outperforms the traditional VSM model and LDA model in tweet clustering.

5. Acknowledgments

This work is supported by NSFC China (No. 61272233). We thank the reviewers for the valuable comments.

6. References

Somnath Banerjee, Krishnan Ramanathan, and Ajay Gupta. 2007. Clustering short texts using wikipedia. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '07, pages 787–788, New York, NY, USA. ACM.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, March.

Samuel Brody and Mirella Lapata. 2009. Bayesian word sense induction. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, EACL '09, pages 103–111, Stroudsburg, PA, USA. Association for Computational Linguistics.

Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. 1990. Indexing by latent semantic analysis. *JOURNAL OF THE AMERICAN SOCIETY FOR INFORMATION SCIENCE*, 41(6):391–407.

Paolo Ferragina and Ugo Scaiella. 2010. Tagme: on-the-fly annotation of short text fragments (by wikipedia entities). In *Proceedings of the 19th ACM international conference on Information and knowledge management*, CIKM '10, pages 1625–1628, New York, NY, USA. ACM.

Evgeniy Gabrilovich and Shaul Markovitch. 2006. Overcoming the brittleness bottleneck using wikipedia: enhancing text categorization with encyclopedic knowledge. In *proceedings of the 21st national conference on Artificial intelligence - Volume 2*, AAAI'06, pages 1301–1306. AAAI Press.

T. L. Griffiths and M. Steyvers. 2004. Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101(Suppl. 1):5228–5235, April.

Andreas Hotho, Steffen Staab, and Gerd Stumme. 2003. Wordnet improves text document clustering. In *In Proc. of the SIGIR 2003 Semantic Web Workshop*, pages 541–544.

Jian Hu, Lujun Fang, Yang Cao, Hua-Jun Zeng, Hua Li, Qiang Yang, and Zheng Chen. 2008. Enhancing text clustering by leveraging wikipedia semantics. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '08, pages 179–186, New York, NY, USA. ACM.

Anand Karandikar. 2010. Clustering short status messages: A topic model based approach. Master's thesis, July.

Xuan-Hieu Phan, Le-Minh Nguyen, and Susumu Horiguchi. 2008. Learning to classify short and sparse text & web with hidden topics from large-scale data collections. In *Proceedings of the 17th international conference on World Wide Web*, WWW '08, pages 91–100, New York, NY, USA. ACM.

Daniel Ramage, Susan T. Dumais, and Daniel J. Liebling. 2010. Characterizing microblogs with topic models. In William W. Cohen and Samuel Gosling, editors, *ICWSM*. The AAAI Press.

Alan Ritter, Colin Cherry, and Bill Dolan. 2010. Unsupervised modeling of twitter conversations. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, pages 172–180, Stroudsburg, PA, USA. Association for Computational Linguistics.

Mehran Sahami and Timothy D. Heilman. 2006. A web-based kernel function for measuring the similarity of short text snippets. In *Proceedings of the 15th international conference on World Wide Web*, WWW '06, pages 377–386, New York, NY, USA. ACM.

Gerard Salton, A. Wong, and C. S. Yang. 1974. A vector space model for automatic indexing. Technical report, Ithaca, NY, USA.

Helmut Schmid. 1994. Probabilistic part-of-speech tagging using decision trees. 12:44–49.

Gerasimos Spanakis, Georgios Siolas, and Andreas Stafylopatis. 2012. Exploiting wikipedia knowledge for conceptual hierarchical clustering of documents. *Comput.*

J., 55(3):299–312, March.

M. Steinbach, G. Karypis, and V. Kumar. 2000. A comparison of document clustering techniques. *KDD Workshop on Text Mining*.

E. M. Voorhees. 1986. Implementing Agglomerative Hierarchical Clustering Algorithms for use in Document Retrieval. In *Information Processing & Management*, volume 22, pages 465–476. Pergamon Press.

Pak-kwong Wong and Chorkin Chan. 1996. Chinese word segmentation based on maximum matching and word binding force. In *Proceedings of the 16th conference on Computational linguistics - Volume 1*, COLING '96, pages 200–203, Stroudsburg, PA, USA. Association for Computational Linguistics.