Probabilistic Semantic Similarity Measurements for Noisy Short Texts Using Wikipedia Entities

Masumi Shirakawa† Kotaro Nakayama‡ Takahiro Hara† Shojiro Nishio†

†Graduate School of Information Science and Technology, Osaka University, Osaka, Japan
{shirakawa.masumi, hara, nishio}@ist.osaka-u.ac.jp

‡Center for Knowledge Structuring, University of Tokyo, Tokyo, Japan
nakayama@cks.u-tokyo.ac.jp

ABSTRACT

This paper describes a novel probabilistic method of measuring semantic similarity for real-world noisy short texts like microblog posts. Our method adds related Wikipedia entities to a short text as its semantic representation and uses the vector of entities for computing semantic similarity. Adding related entities to texts is generally a compound problem that involves the extraction of key terms, finding related entities for each key term, and the aggregation of related entities. Explicit Semantic Analysis (ESA), a popular Wikipedia-based method, solves these problems by summing the weighted vectors of related entities. However, this heuristic weighting highly depends on the rule of majority decision and is not suited to short texts that contain few key terms but many noisy terms. The proposed probabilistic method synthesizes these procedures by extending naive Bayes and achieves robust estimates of related Wikipedia entities for short texts. Experimental results on short text clustering using Twitter data indicated that our method outperformed ESA for short texts containing noisy terms.

Categories and Subject Descriptors

H.3.1 [**INFORMATION STORAGE AND RETRIEVAL**]: Content Analysis and Indexing—*Linguistic processing, Thesauruses*; I.7.1 [**DOCUMENT AND TEXT PROCESSING**]: General

General Terms

Algorithms, Experimentation

Keywords

Semantic similarity; Semantic representation; Naive Bayes; Short text clustering

CIKM'13, Oct. 27–Nov. 1, 2013, San Francisco, CA, USA.

ACM 978-1-4503-2263-8/13/10 ...$15.00.
http://dx.doi.org/10.1145/2505515.2505600.

1. INTRODUCTION

Recently, a great deal of attention has focused on analyzing short texts, such as microblogs, search queries, search results, ads, and news feeds. Semantic similarity measurements between short texts are especially substantial for various applications including text clustering [2] and text classification [26]. To measure semantic similarity between short texts, enriching semantics of short texts is a vital technology because these types of text do not contain enough information for semantic similarity measurements. For example, two short texts *Apple's new product* and *iPhone 5 was launched* refer to similar topics, even though there are no term co-occurrence between them. Measuring their semantic similarity requires additional semantics (i.e., external knowledge sources).

Wikipedia[1], a large-scale collaborative encyclopedia, is a promising knowledge source that can be used to add semantics of short texts. When compared to the Web, Wikipedia has more refined link structures and less noise data, and it widely covers named entities, domain specific entities, and new entities. Thanks to Wikimedia Foundation's contribution to provide the dump data online[2], it can easily be utilized. It has been used as a knowledge source to accomplish various tasks on handling short texts [2, 5, 19].

Explicit Semantic Analysis (ESA) [7], a method that is used to find related Wikipedia entities as semantics of texts, is designed for various purposes rather than a specific task. Additional semantics by Wikipedia entities represent the topic of the texts, enabling accurate semantic similarity measurements for short texts. Adding semantics of texts consists of several subproblems: extracting key terms, finding related entities for each key term, and aggregation of related entities. In order to solve these subproblems, ESA sums the weighted vectors of related entities for each keyword according to the majority rule.

ESA's weighting mechanism, however, is not suited to real-world noisy short texts. To find related Wikipedia entities for a noisy short text, it is important to focus on keywords while filtering out noisy words. Since ESA relies on the majority rule, it does not have a function to filter out noisy words. ESA does not work well if a short text contain few keywords and many noisy words.

[1]http://www.wikipedia.org/
[2]http://dumps.wikimedia.org/

The purpose of this work is to use Wikipedia entities as the semantics of short texts to measure the semantic similarity. In order to achieve this objective, we adapt Wikipedia-based techniques to define the probabilistic scores for key term extraction and related entity finding, and introduce extended naive Bayes (ENB) to aggregate related entities. Our method is more robust for noisy short texts than ESA because its weighting mechanism is based on the Bayes' theorem.

2. RELATED WORK

Short texts vary from traditional documents in their brevity and sparsity, which makes statistical approaches to short texts less effective. Thus, enriching the semantics of short texts using external knowledge, such as Wikipedia, is essential.

Most of the work leveraged Wikipedia for specific tasks on short texts. Ferragina et al. [5] proposed a simple and fast method for entity disambiguation (entity linking) for short texts using Wikipedia. Meij et al. [14] also tackled entity disambiguation by using various features (e.g., anchor texts, links between articles) derived from Wikipedia for machine learning. Phan et al. [19] utilized hidden topics obtained from Wikipedia for learning the LDA [3] classifier of short texts. Hu et al. [8] exploited features from Wikipedia for clustering of short texts. Their work demonstrated that Wikipedia was effective as an external knowledge source.

Contrast to the work described above, research on representing semantics of short texts was used for multiple purposes [7, 15]. Especially, Explicit Semantic Analysis (ESA) [7] has been widely used because of its availability and versatility. ESA was developed for computing word similarity as well as text similarity written in natural languages. ESA builds a weighted inverted index that maps each word into a list of Wikipedia articles in which it appears, and computes the similarity between vectors generated from two words or texts.

Song et al. [24] illustrated the availability of ESA for short text clustering (as a comparative method), i.e., measuring semantic distance (semantic similarity) between short texts using ESA. Banerjee et al. [2] also employed a similar approach to ESA for the purpose of clustering short texts. Sun et al. [26] utilized ESA to classify short texts with a support vector machine (SVM), which is a supervised machine learning technique.

Thus, ESA has been demonstrated to be effective for measuring semantic similarity for short texts. However, ESA has a problem in its weighting system when it comes to analyzing real-world noisy short texts. We will describe this in Section 3.

3. WEAKNESS OF ESA

Explicit Semantic Analysis (ESA) [7] is a method to represent semantics of short texts for semantic similarity measurements. ESA builds a weighted inverted index that maps each word into a list of Wikipedia articles in which it appears, and computes the similarity between vectors generated from two texts or words. To make a short list of related Wikipedia entities for a text that contain multiple words, ESA sums the weighted vectors of related entities for each word. This simple weighting works well for long texts such as news articles and web pages because the scores of related entities belonging to the most dominant topic of the text naturally increase based on the majority rule.

However, we posit that ESA is not well-designed for finding related Wikipedia entities for real-world noisy short texts. Noisy short texts may contain few key terms and many noisy terms, and the majority rule ESA has employed may not work well. It is important to focus on key terms as well as filter out noisy terms to correctly derive related entities from short texts.

Noisy terms cannot be filtered out statically because a noisy term in a text can be a key term in another text depending on the contexts. For example, general term *tree* may be a noisy term in many texts, but it can be a key term that indicates a data structure in the domain of computer sciences. A plant *tree* can also be a key term in the topic of botany. Named entities may be noisy terms in some cases. City name *Liverpool* may only explain *John Lennon*'s birthplace and the main topic of the text may be popular music. Uniformly giving low scores to such noisy terms does not lead to a resolution of the problem.

Another serious issue of ESA is that strongly related entities to a single term tend to remain in the top of the output list. This situation is undesirable especially when the text contains ambiguous terms. For example, *iPhone* is strongly related to term *Apple* but is not related to short text *Apple is a tree*. In this case, summed scores of related entities that are related to both *Apple* and *tree* (e.g. *Fruit* and *Golden Delicious*) should exceed the score of *iPhone*. However, with ESA's weighting mechanism, this hardly occurs if the differential of the scores of *Apple* and *tree* is large. Related entities of the text are then almost derived from only *Apple*. Even if the scores of *Apple* and *tree* are similar, it is likely that few entities related to both *Apple* and *tree* are ranked higher than *iPhone*. That is, related entities belonging to different topics (one of them is actually wrong) coexist in the top of the output list, resulting in deteriorating the performance of semantic similarity measurements.

4. METHOD

To achieve robust finding of related Wikipedia entities for short texts to measure semantic similarity, we propose a method that adapts Wikipedia-based techniques to define probabilistic scores and integrates the scores based on the Bayes' theorem. As described in Section 3, ESA is not suited to real-world noisy short texts because of its simple weighting mechanism of summing weighted vectors. Our method addresses the problem by extending the naive Bayes method, which enable us to emphasize key terms while filter out noisy terms.

Our method obtains probabilistic scores for key terms and related entities by analyzing Wikipedia. After that, our method synthesizes these probabilities and computes the output vector of related entities using extended naive Bayes (ENB). To measure the semantic similarity between two texts, the similarity of their related entity lists ranked by $P(c|T)$, probability that entity c is related to a set of key terms T, is computed using cosine or other metrics.

4.1 Probabilistic Scores from Wikipedia

Our method solves the compound problem of key term extraction, related entity finding, and the aggregation of related entities in a probabilistic schema. In this section, we explain the probabilistic scores of key terms and related en-

tities, as well as the prior probabilities of entities. In Section 4.2, we describe how to aggregate related entities for each key term using the probabilities introduced in this section.

4.1.1 Key Term Extraction

$P(t \in T)$, which is the probability that term t in a text, T, is a key term, is computed using anchor texts in Wikipedia articles [16]. According to the editorial policy of Wikipedia called *wikify*[3], a specific term in Wikipedia articles that indicates another article (entity) should be linked to the article. Here, the more often a term is selected as an anchor text for a corresponding article, the more likely that the term is important. Based on this heuristics, we use the rate of articles that contain a term as an anchor text. According to the literature [16], this method of extracting key terms outperformed other common techniques, such as TFIDF [21] and the χ^2 independence test [13].

Given that $CountArticlesHavingAnchortexts(t)$ is the number of articles that contain term t as an anchor text and $CountArticlesHavingTerms(t)$ is the number of articles that contain term t, the probability is computed as

$$P(t \in T) \approx \frac{CountArticlesHavingAnchortexts(t)}{CountArticlesHavingTerms(t)}. \quad (1)$$

In order to avoid black or white probabilities (i.e., 0 or 1), Laplace Smoothing [11] is introduced.

4.1.2 Related Entity Finding

$P(e|t)$, which is the probability that term t is linked to entity e, is computed using anchor texts and their destination article [18]. Using the policy of *wikify*, a specific term that indicates an entity is linked to a corresponding article. This term then becomes an anchor text for the entity. The relationship between terms and entities can be extracted by analyzing anchor texts. Given that $CountAnchortexts(t, e)$ is the number of times that the anchor text t is linked to entity e, the probability is as follows:

$$P(e|t) \approx \frac{CountAnchortexts(t, e)}{\sum_{e_i \in E} CountAnchortexts(t, e_i)}. \quad (2)$$

E denotes a set of all Wikipedia entities.

$P(c|e)$, which is the probability that entity c is related to entity e, is computed based on incoming and outgoing links of e. We introduce a link-based method that simply uses the number of links between e and c. Given that $CountLinks(e, c)$ is the number of links (regardless of incoming or outgoing links) between two entities, e and c, the probability is computed as

$$P(c|e) \approx \frac{CountLinks(e, c)}{\sum_{c_j \in E} CountLinks(e, c_j)}. \quad (3)$$

By using Equations (2) and (3), $P(c|t)$, the probability that entity c is related to term t, is computed as

$$P(c|t) = \sum_{e_i \in E} P(c|e_i)P(e_i|t). \quad (4)$$

4.1.3 Prior Probability of Entities

$P(c)$, which is the prior probability of entity c, means the generality of c. Because our method computes $P(c|t)$ using the number of links between articles, we also use it for determining prior probability. Namely, prior probability is in proportion to the number of incoming and outgoing links. Given that $CountLinks(c)$ is the number of incoming and outgoing links that an entity c has, the prior probability can be computed as

$$P(c) \approx \frac{CountLinks(c)}{\sum_{c_j \in E} CountLinks(c_j)}. \quad (5)$$

4.2 Extended Naive Bayes

Based on the probabilistic scores extracted from Wikipedia, we attempt to integrate the information to find related Wikipedia entities for texts. First, we start by assuming multiple key terms are input. In other words, we calculate $P(c|T')$ for a set of key terms $T' = \{t_1, ..., t_K\}$[4]. Using $P(c|t)$ and $P(c)$, $P(c|T')$ can be derived using conventional naive Bayes [24]. Given that each term, t, is conditionally independent, the probability is specifically computed by

$$\begin{aligned} P(c|T' = \{t_1, ..., t_K\}) &= \frac{P(c)\prod_{k=1}^{K} P(t_k|c)}{P(T' = \{t_1, ..., t_K\})} \\ &= \frac{\prod_{k=1}^{K} P(c|t_k)P(t_k)}{P(T' = \{t_1, ..., t_K\})P(c)^{K-1}} \\ &= \frac{\prod_{k=1}^{K} P(c|t_k)}{P(c)^{K-1}}. \end{aligned} \quad (6)$$

Next, we tackle the case where members of T cannot be observed, i.e., it is not clear whether a term in a text is a key term or not. Because candidates of the key term in a text can be determined using anchor texts and titles of Wikipedia, this assumption is the same as what we have considered in this work. One of the possible approaches to this challenge may be the two-phase method that first determines key terms and then applies conventional naive Bayes to them. However, this approach gives rise to the problem of how key terms are determined. Threshold-based methods can be employed to select or discard terms, although this requires parameter adjustments. Adjusting thresholds is difficult because optimal thresholds may change along with texts.

Instead of using threshold-based methods, we propose extended naive Bayes (ENB)[5]. ENB can be applied to a set whose members are probabilistically determined. Given a set of key terms T, $P(c|T')$ is computed for all possible states T'. Figure 1 outlines an example of ENB for a set of candidates of the key terms $t_1, ..., t_K$. ENB is used to compute $P(T = T')$, which is the probability that a set of key terms, T, will become state T'. It then computes $P(c|T')$ for each state T' and sums up $P(c|T')$ weighted by $P(T = T')$.

Given that each term, t, is conditionally independent, $P(T = T')$ is computed as

$$\begin{aligned} P(T = T') &= \prod_{t_k \in T'} P(t_k \in T) \prod_{t_k \notin T'} P(t_k \notin T) \\ &= \prod_{t_k \in T'} P(t_k \in T) \prod_{t_k \notin T'} (1 - P(t_k \in T)). \end{aligned} \quad (7)$$

[3]http://en.wikipedia.org/wiki/Wikipedia:WikiProject_Wikify

[4]In this paper, we use T for a set of key terms whose members cannot be observed and T' (with an apostrophe) for a set of key terms whose members can be observed.

[5]The basic idea of ENB was originally proposed in the report [23].

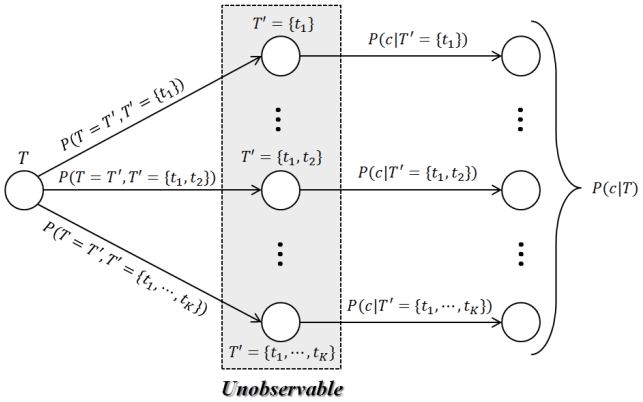

Figure 1: Extended naive Bayes (ENB) for set of key terms whose members cannot be observed.

Therefore, related entities are estimated by using the ENB in Figure 1 utilizing Equations (6) and (7).

$$P(c|T) = \sum_{T'}\Big(P(T=T')P(c|T')\Big) = \sum_{T'}\Big(P(T=T')\frac{\prod_{t_k\in T'}P(c|t_k)}{P(c)^{|T'|-1}}\Big) \quad (8)$$

Here, $|T'|$ denotes the number of key terms contained in T'. The computation of Equation (8) requires exponential time for the number of terms K because it separately applies conventional naive Bayes to each state T'. Equation (8) can be decomposed by dual cases $t_k \in T'$ and $t_k \notin T'$ as

$$\frac{\sum_{T'}\Big(\prod_{t_k\in T'}P(t_k\in T)P(c|t_k)\prod_{t_k\notin T'}\big(1-P(t_k\in T)\big)P(c)\Big)}{P(c)^{K-1}}. \quad (9)$$

The numerator of Equation (9) is then decomposed for each t_k to efficiently compute it.

$$\begin{aligned}
&\sum_{T'}\Big(\prod_{t_k\in T'}P(t_k\in T)P(c|t_k)\prod_{t_k\notin T'}\big(1-P(t_k\in T)\big)P(c)\Big)\\
&=\Big(P(t_1\in T)P(c|t_1)+\big(1-P(t_1\in T)\big)P(c)\Big)\\
&\quad\sum_{T'}\Big(\prod_{t_k\in T'\wedge t_k\notin\{t_1\}}P(t_k\in T)P(c|t_k)\prod_{t_k\notin T'\wedge t_k\notin\{t_1\}}\big(1-P(t_k\in T)\big)P(c)\Big)\\
&=\ \cdots\\
&=\prod_{k=1}^{K}\Big(P(t_k\in T)P(c|t_k)+\big(1-P(t_k\in T)\big)P(c)\Big)
\end{aligned}$$

As a result, the following expression is derived.

$$P(c|T)=\frac{\prod_{k=1}^{K}\Big(P(t_k\in T)P(c|t_k)+\big(1-P(t_k\in T)\big)P(c)\Big)}{P(c)^{K-1}} \quad (10)$$

Consequently, Equation (10) replaces each probability $P(c|t_k)$ in the conventional naive Bayes (Equation (6)) with a linear combination of $P(c|t_k)$ and its prior probability $P(c)$. In the equation, $P(t_k \in T)$ plays a role as the weight for smoothing. That is, ENB naturally includes the smoothing mechanism obtained by $P(t_k \in T)$ to focus on the key terms while filtering out noisy terms.

On the assumption that there is at least one key term, $P(t_k \in T)$ can be normalized by dividing it by the maximum probability. Also, $P(c|T)$ may requires the normalization because $P(t_k \in T)$, $P(c|t_k)$, and $P(c)$ are approximate probabilities. The similarity of related entity lists ranked by $P(c|T)$ obtained from two texts is computed using cosine or other metrics to measure semantic similarity.

5. EVALUATION

5.1 Semantic Similarity Measurements on Benchmark Datasets

We evaluated our method and ESA with a variety of parameter combinations on benchmarks of short text semantic similarity. We particularly leveraged Pilot short text semantic similarity benchmark dataset [10], which contains 30 sentence pairs and their similarity scores rated by 32 participants. Additionally, we created three datasets using ConceptSim [22] and WordNet [4]. We followed the manner of the literature [10] to build short text similarity datasets, i.e., replaced a synset (a single meaning of a word) of WordNet with its definition. As the result, we obtained three datasets based on the gold standards of word similarity datasets: MC [17], RG [20], and WordSim (WS) [1, 6]. Spearman's rank correlation coefficient is used to measure the similarity scores with those by human judgments.

We examined 16 combinations of parameter settings of ESA: keyphraseness [16] (KEY) or IDF [21] for key term extraction, count of anchor texts (A) or logarithmic count of anchor texts (logA) for linking a key term to entities, count of links (L) or logarithmic count of links (logL) for finding related entities from an entity, and cosine-normalized scores of related entities (COS) or unnormalized scores. Moreover, we also implemented original ESA according to the Gabrilovich's work [7]. Since our method and ESA generated a ranked list of entities as output, we used the top 100, 200, 500, 1,000, 2,000, and 5,000 entities to compute similarity scores[6]. We then marked the best score among them per method.

Table 1 shows the results of semantic similarity measurements on the benchmark datasets. Our method outperformed ESA with KEY-A-L (the parameter settings are the same as our method's) for all the datasets. Compared to original ESA, the performance of our method was marginally fine. However, by adjusting parameters, ESA was able to achieve higher scores than our method (e.g. IDF-A-logL-COS, IDF-logA-logL-COS). These datasets consist of formal texts and ESA is accurate enough to measure semantic similarity between the short texts. This means that our method has no significant advantage in these datasets because it is not needed to filter out noisy terms to correctly grasp the topic of the texts.

In spite of these results, our method is more effective than ESA for real-world noisy short texts. In Section 5.2, we will demonstrate that our method can surpass ESA with the best parameter settings that are adjusted based on these datasets (i.e., IDF-A-logL-COS).

[6]We did not evaluate the methods when the number of related entities was less than 100, because the similarity scores became 0 for many unrelated sentence pairs. Spearman's rank correlation coefficient cannot be measured if not a few scores are the same.

Table 1: Spearman's rank correlation coefficient for short text similarity datasets.

Method	Pilot	MC	RG	WS
ESA				
KEY-A-L (ESA-same)	0.733	0.777	0.681	0.506
KEY-A-L-COS	0.824	0.826	0.727	0.542
KEY-A-logL	0.823	0.754	0.690	0.571
KEY-A-logL COS	0.797	0.814	0.710	0.559
KEY-logA-L	0.771	0.814	0.626	0.447
KEY-logA-L COS	0.820	0.856	0.650	0.528
KEY-logA-logL	0.866	0.840	0.713	0.505
KEY-logA-logL COS	0.785	0.866	0.706	0.553
IDF-A-L	0.737	0.893	0.790	0.392
IDF-A-L-COS	0.886	0.835	0.791	0.523
IDF-A-logL	0.845	0.869	0.778	0.509
IDF-A-logL-COS (ESA-adjusted)	0.885	0.894	0.806	0.569
IDF-logA-L	0.692	0.746	0.694	0.364
IDF-logA-L-COS	0.856	0.840	0.768	0.505
IDF-logA-logL	0.838	0.838	0.737	0.484
IDF-logA-logL-COS	0.883	0.897	0.784	0.578
Original ESA [7]	0.797	0.833	0.698	0.562
Our method	0.857	0.840	0.717	0.573

5.2 Tweet Clustering

5.2.1 Setup

We carried out an experiment on clustering of Twitter messages (tweets). In the same clustering algorithm, the performance of clustering depends on how the semantic distance (semantic dissimilarity) is measured. Namely, the performance of semantic similarity measurements can be evaluated using clustering. We employed k-means clustering [12] as the clustering algorithm.

We utilized the *hashtags*, which are defined by Twitter, to create large-scale datasets (i.e., ground truth) for clustering tasks. *Hashtags* are tags, such as *#MacBook* and *#NFL*, that Twitter users intentionally add to their tweets in order to clarify the topic of the tweet [9]. It has actually been demonstrated that hashtags are used to create datasets for short text clustering [24]. In our experiment, we carefully selected independent, unambiguous hashtags (topics) shown in Table 2 so that each cluster contained a maximum of appropriate tweets. Note that collected tweets still contained many ambiguous terms.

The procedure for constructing the dataset was as follows: 1) search tweets by using predefined hashtags and store those written in English, 2) delete tweets that contain more than one predefined hashtag, 3) delete retweets (tweets starting with *RT*), 3) remove URLs in tweets, 4) remove hashtags at the end of tweets (to hide explicit clues for the topic) and the "#" of hashtags not at the end of tweets, and 5) delete tweets that are under four words. Table 3 represents the statistics of the dataset that was used in the evaluation.

We employed a bag-of-words model (BOW) as the baseline and ESA as the comparative method. For BOW, we used all words except stop words in short texts to compute the semantic similarity. For ESA, we employed two parameter settings: the same parameter settings as our method's (ESA-same) and the best parameter settings for the benchmark datasets in Section 5.1 (ESA-adjusted, i.e., IDF-A-logL-COS). We used the top 10, 20, 50, 100, 200, 500, 1,000, 2,000, and 5,000 related entities for measuring semantic similarity as for ESA and our method. We did not use combined methods of BOW and Wikipedia-based methods (ESA and our method) because the purpose of this experiment was to assess the performance of each method for semantic similarity measurements.

Table 2: Predefined hashtags to build the dataset (the number of collected tweets).

#MacBook (1,251)	#Silverlight (221)	#VMWare (890)
#MySQL (1,241)	#Ubuntu (988)	#Chrome (1,018)
#NFL (1,044)	#NHL (1,045)	#NBA (1,085)
#MLB (752)	#MLS (981)	#UFC (991)
#NASCAR (878)		

Table 3: Statistics of the dataset.

Total number of tweets	12,385
Total number of words	174,843
Average number of words per tweet	14.1
Total number of unique words	31,489

We employed normalized mutual information (NMI) [25] as the metric to evaluate the performance. NMI expresses scores based on information theory and is regarded as one of the most reliable metrics for clustering. NMI scores are between 0 and 1, and larger scores indicate better results. we conducted k-means clustering 20 times with random initial clusters and we recorded the average score for each method.

5.2.2 Results

Figure 2 shows the results of tweet clustering (maximum NMI scores are described in figures). In comparison with the bag-of-words (BOW) method, ESA-adjusted and our method achieved good performance because they were able to finely enhance the semantics of short texts to increase the co-occurrences of Wikipedia entities in tweets. From Table 3, the average number of words per tweet is less than 15. This indicates that there are few co-occurrences of terms in tweets and the BOW method often fails to measure semantic similarity between tweets. The same tendency can be observed in the literature [24], which has reported that BOW or statistical approaches, such as LDA [3], are ineffective for computing semantic distance in short text clustering. Generated features by ESA-same were not superior to BOW because of inappropriate parameter settings.

Of the Wikipedia-based approaches, our method outperformed ESA even if the parameters of ESA are well-adjusted. We applied t-test to compare the best performance between our method and ESA methods, and it was significantly different with the p-value < 0.01. From the results, our method is more suited to real-world noisy short texts than ESA.

A good example that illustrates the effectiveness of our method versus ESA is *Kobe's 48 will be the highlight of the Lakers season lol* (the topic is *NBA*). Of all the terms in this sentence, *Kobe* (indicating NBA player Kobe Bryant) and *Lakers* (indicating NBA team Los Angeles Lakers) are key terms, and *highlight* and *lol* are likely to be noisy terms. Additionally, *Kobe* is highly ambiguous as it usually denotes a Japanese city *Kobe*. The output of ESA-adjusted or ESA-same contained many unrelated entities that were derived from the noisy or ambiguous terms. Only the proposed method accurately derived NBA-related entities such as *Kobe Bryant* and *Los Angeles Lakers accomplishments*

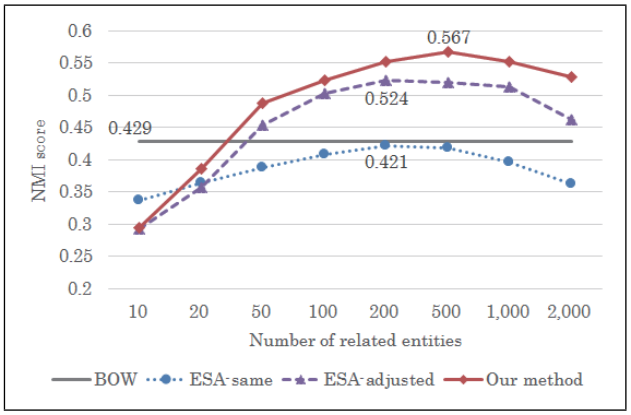

Figure 2: Results of tweet clustering.

and records by filtering out noisy terms and amplifying related entities that were related to multiple key terms.

6. CONCLUSIONS

Using the Bayes' theorem, we proposed a novel probabilistic method to find related Wikipedia entities for short texts to measure semantic similarity. Adding related entities to texts is a compound problem including key term extraction, related entity finding, and the aggregation of related entities. To address the compound problem, our method derives the probabilistic scores for key term extraction and related entity finding using Wikipedia, and aggregates the weighted vector of related entities for each key term by using extended naive Bayes (ENB). The performance of the proposed method on short text similarity datasets was inferior to that of Explicit Semantic Analysis (ESA) with well-adjusted parameters, which solved the compound problem using the heuristic weighting mechanism of summing scores. However, from the experimental results using real-world Twitter data, we confirmed that our method was more robust in measuring semantic similarity for noisy short texts than the ESA.

We plan to develop our method in future work to deal with multi languages at once. Our method uses Wikipedia articles (entities) and does not use any NLP technique. Because Wikipedia articles of different languages are connected each other using inter-language links, our method can be multilingualized using them. We will explore how to incorporate probabilistic scores of language mapping into our probabilistic method.

7. ACKNOWLEDGMENTS

This work was supported in part by CPS-IIP Project (Integrated Platforms for Cyber-Physical Systems to Accelerate Implementation of Efficient Social Systems (FY2012 - FY2016)) in the research promotion program for national level challenges "research and development for the realization of next-generation IT platforms" by the Ministry of Education, Culture, Sports, Science and Technology (MEXT).

8. REFERENCES

[1] E. Agirre, E. Alfonseca, K. Hall, J. Kravalova, M. Pasça, and A. Soroa. A Study on Similarity and Relatedness Using Distributional and WordNet-based Approaches. In *HLT-NAACL*, pages 19–27, 2009.

[2] S. Banerjee, K. Ramanathan, and A. Gupta. Clustering Short Texts Using Wikipedia. In *SIGIR*, pages 787–788, 2007.

[3] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[4] C. Fellbaum. *WordNet: An Electronic Lexical Database*. The MIT Press, 1998.

[5] P. Ferragina and U. Scaiella. TAGME: On-the-fly Annotation of Short Text Fragments (by Wikipedia Entities). In *CIKM*, pages 1625–1628, 2010.

[6] L. Finkelstein, E. Gabrilovich, Y. Matias, E. Rivlin, Z. Solan, G. Wolfman, and E. Ruppin. Placing Search in Context: The Concept Revisited. *ACM TOIS*, 20(1):116–131, 2002.

[7] E. Gabrilovich and S. Markovitch. Computing Semantic Relatedness using Wikipedia-based Explicit Semantic Analysis. In *IJCAI*, pages 1606–1611, 2007.

[8] X. Hu, N. Sun, C. Zhang, and T.-S. Chua. Exploiting Internal and External Semantics for the Clustering of Short Texts Using World Knowledge. In *CIKM*, pages 919–928, 2010.

[9] D. Laniado and P. Mika. Making Sense of Twitter. In *ISWC*, pages 470–485, 2010.

[10] Y. Li, D. McLean, Z. A. Bandar, J. D. O'Shea, and K. Crockett. Sentence Similarity Based on Semantic Nets and Corpus Statistics. *IEEE TKDE*, 18(8):1138–1150, 2006.

[11] G. J. Lidstone. Note on the General Case of the Bayes-Laplace Formula for Inductive or a Posteriori Probabilities. *Transactions of the Faculty of Actuaries*, 8:182–192, 1920.

[12] J. B. MacQueen. Some Methods for classification and Analysis of Multivariate Observations. In *Berkeley Symposium on Mathematical Statistics and Probability*, pages 281–297, 1967.

[13] C. D. Manning and H. Schütze. *Foundations of Statistical Natural Language Processing*. The MIT Press, 1999.

[14] E. Meij, W. Weerkamp, and M. de Rijke. Adding Semantics to Microblog Posts. In *WSDM*, pages 563–572, 2012.

[15] D. Metzler, S. Dumais, and C. Meek. Similarity Measures for Short Segments of Text. In *ECIR*, pages 16–27, 2007.

[16] R. Mihalcea and A. Csomai. Wikify! Linking Documents to Encyclopedic Knowledge. In *CIKM*, pages 233–241, 2007.

[17] G. A. Miller and W. G. Charles. Contextual Correlates of Semantic Similarity. *Language and Cognitive Processes*, 6(1):1–28, 1991.

[18] D. Milne and I. H. Witten. Learning to Link with Wikipedia. In *CIKM*, pages 509–518, 2008.

[19] X.-H. Phan, L.-M. Nguyen, and S. Horiguchi. Learning to Classify Short and Sparse Text & Web with Hidden Topics from Large-scale Data Collections. In *WWW*, pages 91–100, 2008.

[20] H. Rubenstein and J. B. Goodenough. Contextual Correlates of Synonymy. *CACM*, 8(10):627–633, 1965.

[21] G. Salton and C. Buckley. Term-weighting Approaches in Automatic Text Retrieval. *Information Processing and Management*, 24(5):513–523, 1988.

[22] H. A. Schwartz and F. Gomez. Evaluating Semantic Metrics on Tasks of Concept Similarity. In *FLAIRS*, 2011.

[23] M. Shirakawa, H. Wang, Y. Song, Z. Wang, K. Nakayama, T. Hara, and S. Nishio. Entity Disambiguation based on a Probabilistic Taxonomy. Technical Report MSR-TR-2011-125, Microsoft Research, 2011.

[24] Y. Song, H. Wang, Z. Wang, H. Li, and W. Chen. Short Text Conceptualization Using a Probabilistic Knowledgebase. In *IJCAI*, pages 2330–2336, 2011.

[25] A. Strehl and J. Ghosh. Cluster Ensembles – A Knowledge Reuse Framework for Combining Multiple Partitions. *Journal of Machine Learning Research*, 3:583–617, 2002.

[26] X. Sun, H. Wang, and Y. Yu. Towards Effective Short Text Deep Classification. In *SIGIR*, pages 1143–1144, 2011.