Short Text Categorization by Smoothing Word Distribution

Fumiyo Fukumoto*, **Yoshimi Suzuki***

*University of Yamanashi
Takeda 4-3-11, Kofu, Yamanashi, Japan
{ysuzuki, fukumoto}@yamanashi.ac.jp

Abstract

We present a method for short text categorization that maximizes the impact of informative words due to the sparseness of short texts. We used a method of smoothing word distribution. We identified semantically similar words by using the Word2vec, and constructed classes with related words based on the spectral clustering. As a result of clustering, for each class, we randomly selected one word and regarded it as a representative words. We represented each training and test text as a vector, each dimension of a vector is a word/representative word appeared in the text, and applied LibSVM for categorization. The method was tested on scientific paper titles provided by Japan Science and Technology Agency (JST), and the results showed the effectiveness of the method.

Keywords: Short Text, Word Smoothing, Word2vec, Spectral Clustering

1. Introduction

With the exponential growth of information on the Internet, short texts such as search snippets, Web page titles and product reviews are widely available on the Internet. Similarly, in today's academia, publish or perish policy results in an enormous body of publications (Davietov et al., 2014). Automatic categorization of such short texts supports many applications such as building topic directory, creating digital libraries and information retrieval. A growing number of machine learning (ML) techniques have been applied to the text categorization task (Gopal and Yang, 2010). For reasons of both efficiency and accuracy, feature selection is often used since applying machine learning methods to text categorization (Yang and Pedersen, 1997; Dumais and Chen, 2000). Each document is represented using a vector of selected words. Then, labeled documents are used to train classifiers. Each test document is classified by using these classifiers. A basic assumption in the categorization task is that the distributions of words between training and test documents are identical. When the assumption is not hold, the classification accuracy was worse. However, it is often the case that the word distribution in the training data is different from that of the test data when the categorization data consists of short texts. For instance, the number of different noun words within two sets of 1,000 documents randomly collected from Reuters International news, *i.e.*, training data set A and test set B is 8,506, and 8,944, respectively. The number of different noun words appeared in both A and B is 5,667, and their union is 11,783. The ratio is 0.481 (5,667/11,783). In contrast, in the titles collected from the same data A and B, the ratio, *i.e.* the number of different noun words appeared in both A and B was 1,392, and 1,377, respectively. The number of words occurred in both A and B was 629, and the ratio is 0.294 (629/2,140). The observation shows that short texts does not provide enough words co-occurrence between training and test data. The methodology for accurate categorization of the new short text by making the maximum use of tagged data is needed. To alleviate the sparseness of short text, state-of-arts works mainly focus on expanding short texts with knowledge extracted from auxiliary long text corpus (Jin et al., 2011). However, this process is usually domain dependent and thus requires remarkable human efforts in collecting and tuning the data (Zhang et al., 2013).

In this paper, we present an auxiliary-resource-free method for short text categorization that maximizes the impact of informative words in texts. We focused on noun words which frequently appeared in texts. We collected semantically related words by smoothing the word distribution. We replaced these to a representative words in order to regard that these words are equally salient between training and test data. Each document is represented by using a vector of words including representative words, and classifiers are trained. We applied LibSVM to train and classify short texts.

2. Related Work

Short text categorization is widely studied since the early 2000s when growing online social network applications and e-commerce(Deng and Peng, 2006). Unlike normal documents such as news articles and academic papers, short texts are less topic-focused and informative words in texts. Major attempts to tackle these problems are to exploit an external corpus including thesaurus. Chen *et al.* attempted to extracts multiple granularity topics to generate features for short text (Chen et al., 2011). They assume that the quality of topics with a certain granularity depends on two aspects: one is their capability in helping discriminate short text with different class labels; the other is the distance between them and topics with other granularities. They used auxiliary corpus, *i.e.*, Wikipedia page collected by Phan (Phan et al., 2008). Wang *et al.* presented a method of short text classification by using bag-of-concepts in short text representation (Wang et al., 2008). Given the training text per class, they first attempted to construct a concept model for each class. During this construction, they used a large taxonomy knowledge base, Probase (Wu et al., 2012) to convert terms to concepts. Then, given a short text to be classified, the short text is assigned to the relevant concepts. Finally, they used a concept-based similarity to classify and rank short text. All of these approaches mentioned above aimed at utilizing auxiliary data to leverage less informative words of short texts to construct a high-quality classification model. However, when the data distribution of the external knowledge and test data are not identical the classification accuracy

might perform worse. Manual collection and tuning the data is very expensive and timeconsuming. The methodology for accurate classification of short text by making the maximum use of test data only is needed to improve categorization performance.

Long *et al.* addressed this issue, and attempted to use transfer learning to leverage both labeled and unlabeled external data (Long et al., 2012). They used transfer learning to avoid the noise in the large set of unlabeled external data. The transfer learning is a learning technique that retains and applies the knowledge learned in one or more domains to efficiently develop an effective hypothesis for a new domain. The earliest discussion is done by ML community in a NIPS-95 workshop [1], and more recently, transfer learning techniques have been successfully applied in many applications including text classification task (Raina et al., 2006; Dai et al., 2007; Sparinnapakorn and Kubat, 2007). Zelikovitz *et al.* proposes a semi-supervised short text classification method (Zelikovitz and Hirsh, 2000). They used the unlabeled corpus as background knowledge, and applied transductive latent semantic indexing (LSI) to classify text. They reported that the use of LSI increases the accuracy rates for short text classification. However, SVD costs more computing time and space complexity in high-dimensional space (Song et al., 2014).

To our knowledge, there have been only a few previous work on short text classification utilizing short texts only. One attempt is Zhang *et al.* method. They classified short text by detecting information path. They assume that ordered subsets of short text, called information path consists of sequential subsets in the test dataset, and according to this path, instances of former classified subsets can assist classification of later subset, and thus would achieve better classification results than classifying the entire dataset directly. The results by using the search snippets and the paper titles datasets show the effectiveness of the method. However, term selection for detecting information path is entropy-based simple statistical method, and they mentioned that it needs to involve some existing methods, including those leveraging auxiliary resource into their framework for further accuracy gains.

In contrast with the aforementioned works, here we propose a method for short text categorization that maximizes the impact of informative words by using a word smoothing method.

3. Smoothing Word Distributions

A basic assumption in the categorization task is that the distributions of words between training and test documents are identical. When the assumption is not hold, the classification accuracy was worse. However, because of the sparseness of short texts, it is often the case that the word distribution in the training data is different from that of the test data. We then identified words with semantically related, and replaced these to a representative word in order to regard that these words are equally informative across training and test sets.

[1] http://socrates.acadiau.ca/courses/comp/dsilver/NIPS95_LTL/transfer.workshop.1995.html.

Our method for smoothing word distributions consists of two procedures: (i) identification of semantically similar words, and (ii) construction of word classes by using spectral clustering. In the first procedure, we used Word2vec (Mikolov et al., 2013). The Word2vec first constructs a word from the training text data and then learns vector representation of words. It is provided two main model architectures, continuous bag-of-words and skip-gram. We used skip-gram model as it gives better word representations when the data is small (Mikolov et al., 2013). The skip-gram model's objective function L is to maximize the likelihood of the prediction of contextual words given the center word. Given a sequence of training words w_1, w_2, $\cdots$, w_T, the objective of the model is to maximized L:

$$L = \frac{1}{T}\sum_{t=1}^{T}\sum_{-k\leq j\leq k, j\neq 0} \log p(w_{t+j} \mid w_t)$$

where k is a hyperparameter defining the window of the training words. Every word w is associated with two learnable parameter vectors, input vector I_w and output vector O_w of the w. The probability of predicting the word w_i given the word w_j is defined as:

$$p(w_i \mid w_j) = \frac{\exp({I_{w_i}}^\top O_{w_j})}{\sum_{l=1}^{V}\exp({I_l}^\top O_{w_j})}$$

where V refers to the number of words in the vocabulary. For larger vocabulary size, it is not efficient for computation, as it is proportional to the number of words in the V. Word2vec uses the hierarchical softmax objective function to solve the problem. The learned vector representations can be used to find the closest words for a user-specified word. For each category, we collected a small number of documents and create a training data. We applied Word2vec to each training data. As a result, we obtained the number of n models where n is the number of category. Then, for each word in the training data, we collected a certain number of related words whose similarity value is larger than a certain threshold value θ. We created a list of word pairs which satisfy RNN. RNN is so-called Reciprocal Nearest Neighbors that two words are each other's most similar word (Hindle, 1990).

The second procedure is construction of word classes from a list of word pairs. We recall that we identified semantically related word pairs by using both training and test data. The result is large number of word pairs even in short texts. We then applied spectral clustering technique (Ng et al., 2002) to the input of a list of word pairs to attack the problem dealing with the high dimensional spaces. Similar to other clustering algorithms, the spectral clustering takes as input a matrix formed from a pairwise similarity function over a set of data points. Given a set of points $S = \{s_1, \cdots, s_n\}$ in a high dimensional space, the algorithm is as follows:

1. Form a distance matrix $D \in R^2$. We used cosine similarity as a distance measure.
2. D is transformed to an affinity matrix A_{ij}.

Cat	2003	2004	Cat	2003	2004
Measurement	6,859	7,708	Nucleus	117,076	124,255
Chemistry	120,890	127,732	Cosmology	22,636	18,224
Biotechnology	125,214	129,328	Agriculture	44,889	42,752
Medicine	86,509	89,773	Metals	9,173	9,252
Information science	10,220	11,075	Computer	31,785	35,934
Industrial engineering	7,629	7,723	Energy	2,084	2,544
Social problem	3,780	3,800	Device	51,668	65,852
Heat transfer	8,672	9,678	Lubrication	18,025	19,077
Chemical engineering	18,989	18,033	Environment	24,363	25,598
Traffic	3,343	3,464	Extraction	2,433	2,561
Machinery	17,008	19,155	Disasters	8,759	9,890
Ceramic	33,048	34,625	Printing	1,035	1,125

Table 1: Data used in the Experiments

$$A_{ij} = \begin{cases} \exp(-(\frac{D_{ij}^2}{\sigma^2}), & \text{if } i \neq j \\ 0, & \text{otherwise} \end{cases}$$

σ^2 is a parameter and controls the rate at which affinity drops off with distance.

3. The matrix $L = D^{-frac12}AD^{-frac12}$ is created. D is a diagonal matrix whose (i, i) element is the sum of A's i-th row.

4. The eigenvectors and eigenvalues of L are computed, and a new matrix is created from the vectors associated with the number of l largest eigenvalues.

5. Each item now has a vector of l coordinates in the transformed space. These vectors are normalized to unit length.

6. K-means is applied to S in the l-dimensional space.

Finally, for each set obtained by K-means, we randomly selected one word and regareded it as a representative word. If each word appeared in the documents is listed in a set, we replaced the word in the documents to its representative word.

4. Short Text Categorization

So far, we made use of labeled training data and test data in smoothing word distributions. The final step is short text categorization. We trained the model and classified short texts by using LibSVM (Chang and Lin, 2011).

We represented each training and test text as a vector, each dimension of a vector is a word/representative word appeared in the text, and each element of the dimension is a word frequency. We applied LibSVM to the training data, and obtained classifiers. A test text is classified by using these classifiers.

5. Experiments

5.1. Experimental setup

We choose the paper titles dataset provided by JST[2] to test our categorization method. The dataset consists of scientific paper titles published by mainly Japan and USA. The data set is collected from 1981 to 2013. It consists of 9, 798,218 English titles organized into 5 level hierarchies, 3,210 categories in all. Of these, we used data collected from 2004 as smoothing word distribution data, and used training and test data collectd from 2003. we used 24 categories in the top level hierarchy. Table 1 shows the number of titles in each category. All titles were tagged by using Tree Tagger (Schmid, 1995) and noun words are extracted. For each category in 2004 year of Table 1, we applied smoothing word distribution, and obtained clusters of words. We used Word2vec tool released by Google in 2013.

[2]www.jst.go.jp/EN/index.html

5.2. Smoothing results

Table 2 shows statistics obtained by word smoothing. "Words" and "l" in Table 2 show the number of words obtained by Word2vec, and the number clusters in the spectral clustering, respectively. "Cls" refers to the actual number of clusters obtained by the spectral clustering. The similarity value θ is 0.65. Table 3 shows some examples of clusters. The categories in Table 3 are the tompost and lowest two categories.

We can see from Table 3 that several semantic relations between two words are obtained. For instance, hypernym and hyponym words such as {cholinesterase, acetylcholinesterase}, word and its abbriviated word such as {bpa, bisphenol}, symonym words, *e.g.* {technique, method}, and antonym words such as {open-loop, closed-loop}.

5.3. Categorization results

We divided titles shown in Table 1 (2003 year) into five folds, and tested our method by using five cross validation. The results are shown in Table 4. Each value in Table 4 shows the micro-averaged F-score against five folds, and the bottom of Table 4 refers to the macro-averaged F-score against all categories. "w" and "w/o" indicates with and without smoothing word distribution(SWD). We empirically selected values of two parameters, the threshold value θ of the similarity between two words calculated by Word2vec, and the number of clusters l in the spectral clustering. Each value in Table 4 shows the maximized F-score

Cat	Words	l	Cls	Cat	Words	l	Cls
Measurement	839	500	191	Nucleus	6,594	3,000	1,036
Chemistry	2,225	1,000	430	Cosmology	751	700	172
Biotechnology	2,826	2,000	538	Agriculture	1,513	800	348
Medecine	1,445	800	316	Metals	202	100	46
Information science	839	2,000	185	Computer	2,115	2,000	431
Industrial engineering	711	500	163	Energy	269	900	69
Social problem	375	200	85	Device	3,765	1,000	570
Heat transfer	208	200	44	Lubrication	364	200	85
Chemical engineering	2,576	1,000	465	Environment	4,405	2,000	846
Traffic	831	2,000	188	Extraction	512	2,000	117
Machinery	1,795	800	350	Disasters	1,425	2,000	303
Ceramic	4,220	2,000	835	Printing	176	100	43

Table 2: The number of clusters

Cat	Examples of clusters
Nucleus	{nanochains, nanoclusters, nanocubes, nanodisks, nanoflowers, nanoneedles, nanopillars, nanorod, nanoscrolls, nanotubes} {cyclohexanone, benzaldehyde}, {disulphide, disulfide}
Chemistry	{biotransformation, bioconversion}, {bpa, bisphenol}, {cholinesterase, acetylcholinesterase}, {doner, acceptor}, {perovskite, perovskite-type, perovskites}, {disulphide, disulfide}
Industrial engineering	{customization, customisation}, {measurement, calibration}, {open-loop, closed-loop} {precision, accuracy}, {recommendation, guide-line}, {co, corp, corporation, industry, manufacturer}
Printing	{silver, gold}, {technique, method}, {athlete, player}, {colour, color}, {controller, control} {printing, billet, cyanine}, {web, pages}, {regulator, regulation } {microscopy, microscope}

Table 3: The results of clusters

obtained by varying these parameters. The θ value is 0.65, and l is 2,000. "$*$" in Table 4 denotes that "w SWD" is statistical significance t-test with the "$*$" marked method, P-value ≤ 0.05. As can be seen clearly from Table 4, the results with SWD is better to the results wtihout SWD except for the categories "biotechnology" and "agriculture". This shows that smoothing word distributions by Word2vec and spectral clustering contributes categorization performance.

We examined how the θ value of Word2vec affects overall performance of categorization. Figure 1 shows the results. Each bar is the best F-score among different numbers of l in the spectral clustering. Figure 1 shows that when the θ value ranges from 0.9 to 0.65, the F-score increased. However, the θ value is smaller than 0.6, the F-score decreased, and the performance was worse than "w/o" SWD when the value attained at 0.5. This is reasonable because the lower value of θ leads to have a noisy data including semantically dissimilar words.

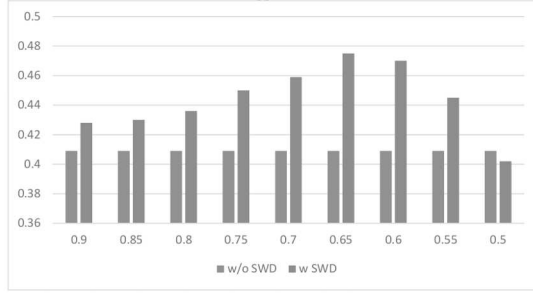

Figure 1: F-score against the θ value of the Word2vec

6. Conclusion

We have presented a method for short text categorization that maximizes the impact of informative words due to the sparseness of short texts. The results using scientific paper titles provided by JST show the efffectiveness of the method. Future work includes (i) extending the method to use hierarchical structure of categories for further improvement, (ii) comparing the method with other related methods (Chen et al., 2011; Zhang et al., 2013), and (iii) evaluating the method by using other data such as the search snippets (Phan et al., 2008) and titles of news articles.

7. References

Chang, C. C. and C. J. Lin, 2011. Libsvm: A Library for Support Vector Machines. *ACM Transactions on Intelligent Systems and Technology*, 2(27):1–27.

Chen, M., X. Jin, and D. Shen, 2011. Short Text Classification Improved by Learning Multi-granularity Topics. In *Proc. of the 22nd International Joint Conference on Artificial Intelligence*.

Dai, W., Q. Yang, G.R. Xue, and Y. Yu, 2007. Boosting for Transfer Learning. In *Proc. of the 24th International Conference on Machine Learning*.

Davietov, F., A. S. Aydin, and A. Cakmak, 2014. High Impact Academic Paper Prediction Using Temporal and

Cat	w/o SWD	w SWD	Cat	w/o SWD	w SWD
Measurement	.120∗	.241	Nucleus	.626∗	.751
Chemistry	.641∗	.749	Cosmology	.606∗	.633
Biotechnology	.665	.685	Agriculture	.545	.531
Medecine	.688	.702	Metals	.351∗	.541
Information science	.415∗	.480	Computer	.436	.439
Industrial engineering	.232	.236	Energy	.229∗	.318
Social problem	.324∗	.463	Device	.547∗	.483
Heat transfer	.411∗	.484	Lubrication	.418∗	.473
Chemical engineering	.459∗	.544	Environment	.403∗	.526
Traffic	.262∗	.302	Extraction	.297	.311
Machinery	.473∗	.549	Disasters	.203∗	.305
Ceramic	.355∗	.426	Printing	.120∗	.218
Macro-avg F				.409	.475

Table 4: Categorization Results with 2003 Data

Topological Features. In *Proc. of the 23rd ACM International Conference on Information and Knowledge Management*.

Deng, W. W. and H. Peng, 2006. Research on a Naive Baysian based Short Message Filtering System. In *Proc. of the 5th International Conference on Machine Learning and Cybernetics*.

Dumais, S. and H. Chen, 2000. Hierarchical Classification of Web Contents. In *Proc. of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*.

Gopal, S. and Y. Yang, 2010. Multilabel Classification with Meta-level Features. In *Proc. of the 33rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*.

Hindle, D., 1990. Noun Classification from Predicate-Argument Structures. In *Proc. of the 28th Annual Meeting of the Association for Computational Linguistics*.

Jin, O., N. Liu, K. Zhao, Y. Yu, and Q. Yang, 2011. Transferring Topical Knowledge from Auxiliary Long Texts for Short Text Clustering. In *Proc. of the 20th ACM International Conference on Information and Knowledge Management*.

Long, G., L. Chen, X. Zhu, and C. Zhang, 2012. TCSST: Transfer Classification of Short & Sparse Text Using External Data. In *Proc. of the 21st ACM International Conference on Information and Knowledge Management*.

Mikolov, T., K. Chen, G. Corrado, and J. Dean, 2013. Efficient Estimation of Word Representations in Vector Space. In *Proc. of the International Conference on Learning Representations Workshop*.

Ng, A. Y., M. I. Jordan, and Y. Weiss, 2002. On Spectral Clustering: Analysis and an Algorithm. In T. G. dietterich, S. Becker, and Z. Ghahramani (eds.), *Advances in Neural Information Processing systems 14*. Cambridge MA MIT Press.

Phan, X. H., L. M. Nguyen, and S. Horiguchi, 2008. Learning to Classify Short and Sparse Text & Web with Hidden Topics from Large-Scale Data Collections. In *Proc. of the 17th International World Wide Web Conference*.

Raina, R., A. Y. Ng, and D. Koller, 2006. Constructing Informative Priors using Transfer Learning. In *Proc. of the 23rd International Conference on Machine Learning*.

Schmid, H., 1995. Improvements in Part-of-Speech Tagging with an Application to German. In *Proc. of the EACL SIGDAT Workshop*.

Song, M., G. E. Heo, and S. Y. Kim, 2014. Analyzing topic evolution in bioinformatics: Investigation of dynamics of the field with conference data in dblp. *Scientometrics*, 101(1):397–428.

Sparinnapakorn, K. and M. Kubat, 2007. Combining Subclassifiers in Text Categorization: A DST-based Solution and a Case Study. In *Proc. of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

Wang, F., Z. Wang, Z. Li, and J. R. Wen, 2008. Concept-based Short Text Classification and Ranking. In *Proc. of the 23rd ACM International Conference on Information and Knowledge Management*.

Wu, W., H. Li, H. Wang, and K. Q. Zhu, 2012. A Probabilistic Taxonomy for Text Understanding. In *Proc. of the 2012 ACM SIGMOD International Conference on Management of Data*.

Yang, Y. and J. O. Pedersen, 1997. A Comparative Study on Feature Selection in Text Categorization. In *Proc. of the 14th International Conference on Machine Learning*.

Zelikovitz, S. and H. Hirsh, 2000. Improving Short Text Classification Using Unlabeled Background Knowledge to Assess Document Similarity. In *Proc. of the 17th International Conference on Machine Learning*.

Zhang, S., X. Jin, D. Shen, B. Cao, X. Ding, and X. Zhang, 2013. Short Text Classification by Detecting Information Path. In *Proc. of the 22rd ACM International Conference on Information and Knowledge Management*.