Multilingual Short Text Classification via Convolutional Neural Network

Jiao Liu, Rongyi Cui, and Yahui Zhao(✉)

Department of Computer Science and Technology, Yanbian University, 977 Gongyuan Road, Yanji 133002, People's Republic of China
{cuirongyi,yhzhao}@ybu.edu.cn

Abstract. As multilingual text increases, the analysis of multilingual data plays a crucial role in statistical translation models, cross-language information retrieval, the construction of parallel corpus, bilingual information extraction and other fields. In this paper, we introduce convolutional neural network and propose auto-associative memory for the fusion of multilingual data to classify multilingual short text. First, the open-source tool word2vec is used to extract word vector for textual representation. Then, the auto-associative memory relationship can extract the multilingual document semantic, which need to calculate the statistical relevance of word vector between different languages. A critical problem is the domain adaptation of classifiers in different languages and we solve it by transforming multilingual text features. In order to fuse a dense combination of high-level features in multilingual text semantics, we introduce convolutional neural network into the model, and output classification prediction results. This model can process multilingual textual data well. Experiments show that convolutional neural network combined with auto-associative memory improves classification accuracy by 2 to 6% in multilingual text classification, compared to other classic models. Furthermore, the proposed model reduces the dependence of multilingual text on the parallel corpus, thus have good expansibility for multilingual data.

Keywords: Auto-associative memory
Convolutional neural network · Word embedding · Local perception

1 Introduction

With the internationalization of information communication, more and more business institutions are doing international activities. For example, government departments often require to classify documents in different languages, the international e-commerce website needs to classify and recommend the goods described in multiple languages, and digital libraries are supposed to provide multilingual information services based on multilingual classification processing for various language users. Under such circumstances, automatic classification technology of different language documents is particularly important.

X. Meng et al. (Eds.): WISA 2018, LNCS 11242, pp. 27–38, 2018.
https://doi.org/10.1007/978-3-030-02934-0_3

It is difficult for feature extraction of short text due to the single short text has few words and its content is scarce in semantics. Recently, multilingual short text classification has received increasing attention, and many algorithms have been proposed during the last decade. Parallel corpus-based methods [12] are usually categorized on the basis of single language documents, and then the corresponding language documents are divided into the same category. CL-ESA is an extension of parallel corpus-based methods [13,14], which represents documents by similarity vectors between documents and indexed document sets. Gliozzo *et al.* [2] categorize text across languages of English and Italian by using a comparative corpus based on latent semantic analysis and classify the document in low-dimensional projection space. Hanneman *et al.* [4] improve the accuracy of classification by constructing syntactic of full-text translation algorithm. He [6] takes advantage of a bilingual dictionary, WordNet, to translate text feature vectors, and then study the similarity between two Chinese and English texts. Tang *et al.* [15] put forward generalized vector space model for cross-lingual text clustering. Faruqui *et al.* [1,3] exploit the canonical correlation analysis for cross-lingual text analysis to find the largest correlation coefficient in the two language spaces, which aim at building a cross-language bridge. Luo *et al.* [10] improve the method of partial least squares to establish the latent intermediate semantics of multiple languages to classify the text across language in the potential space. Kim [8] proposes a convolutional neural network with multiple convolution kernels to classify texts(TextCNN).

Comparing with previous research, we introduce multilingual associative memory to extend convolutional neural network model. The model is constructed by the auto-associative memory relationship among multilingual languages, by the way counting the co-occurrence degree of multilingual words in the parallel corpus and the spatial relationships embedded in their corresponding words. We deal with word items at the document level. In order to exploit multilingual resources available, those matrixes which have the same semantic are amalgamated into single one. Local perception and weight sharing theory of convolutional neural networks could be applied to classify different documents under the combined language space.

Our work adopts TextCNN to address the issue of text characteristics, which fix convolution and learn the characteristics of multiple-word phrases from a combination of different convolution kernels of different lengths. According to the different characteristics of the deep neural network layer, the TextCNN model is extended with the superposition network layer. The experiment demonstrates that the method can merge the language space of the document, and improves the accuracy of the classification effectively.

2 Related Work

Word2Vec is a tool for computing word vectors based on a large-scale corpus, which is proposed by Mikolov [11]. It includes two structures, CBOW and Skip-Ngram, as shown in Fig. 1.

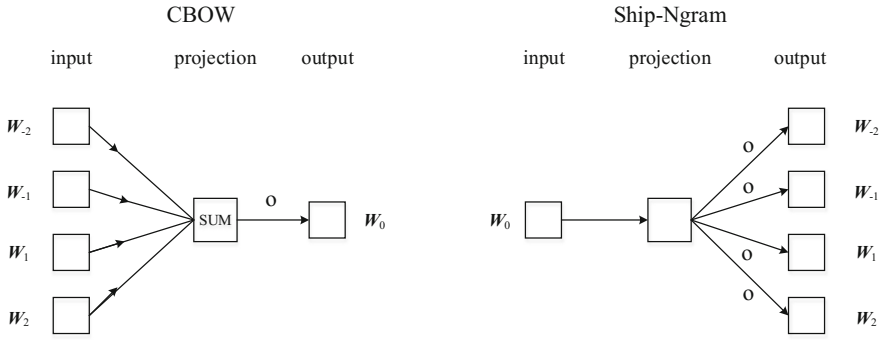

Fig. 1. CBOW and Skip-Ngram models

The model consists of three layers, including an input layer, projection layer and an output layer. For example, the input layer of the CBOW model is composed of $2k$ word vectors in the context of the current word w_0, and the projection layer vector is accumulated by these $2k$ vectors. The output vector could be corresponding with a Huffman tree, every word in the corpus is supposed to be the leaf node, which reference occurrence frequency of each word as weight. Providing that the path from root node to the leaf node is used to represent the word vector of the current word, the goal of this model is to maximize the average logarithmic likelihood function L

$$L = \frac{1}{T} \sum_{t=k}^{T-k} \log p(w_t | w_{t-k}, \cdots w_{t+k}). \tag{1}$$

Equation (1) could be regarded as the prediction of the current word w_t under the context of $w_{t-k}, \cdots, w_{t+k}$. In order to improve training efficiency, the algorithm based on Negative Sampling is proposed, which is suitable for large-scale corpus training. Note that each word is expressed as a word vector in the algorithm. It could be found that the difference between words 'France' and 'Paris' is almost the same as that obtained by 'Italy' minus 'Rome', which proved that the semantic relation between words might be represented by vector linearly.

3 Multilingual Auto-associative Memory

3.1 Co-occurrence Vocabulary Based on Word Embedding

A corpus-based approach to obtain the word co-occurrence is derived from the distribution hypothesis in [5]. In a large corpus, the distribution of words in each document can be indicated as vectors, and the degree of association between words and words can be calculated with this vector. In parallel corpus, if two words belonging to two languages appear in the same semantic document, in

general, it can be deduced that the two words have high semantic relevance. By this relationship, we can find that each word in the vocabulary has a word with the greatest correlation in another language. As a result, we obtain a co-occurrence vocabulary. Not all co-occurrence word pairs can be translated into each other, notwithstanding, they are highly correlated in semantics [9], which has been proved to be suitable for cross-language document retrieval and similarity computation. The computation principle of word2vec reflects the co-occurrence relation between words and its local context when calculating word vectors. The semantic between words could be directly measured by the distance of vector space, which proves that the representation of words might be directly transformed between vectors linearly. However, in different languages, even if there is a translation relationship between source language document d_s and target language document d_t. The words have a similar distribution in the corpus of their respective languages, but they do not have contextual relations, that words in d_s and d_t could not be calculated in the same word context window. Therefore, they could only follow the semantic relations like that "v(France)-v(法国)≈v(Italy)-v(意大利)". By combining the co-occurrence calculation of words and the distance of the word embedding vector, the method generating co-occurrence word pairs is presented as follows:

$$L_{s|t} = \{<x, y> | x \in V_s \wedge y \in V_t \wedge y = T - index(x)\}. \tag{2}$$

where

$$T - index(j) = \arg\max_{i \in V_t}(v_i^T * v_j + \alpha e^{m_{ij}}), j \in V_s. \tag{3}$$

V_S and V_T represent the source language and target language of word items in the document set, respectively. The value of $T - index(j)$ is the number of the word item in the target language co-occurring with the j-th words in the source language. Moreover, v_i and v_j represent the i-th and j-th word vectors of each two languages, α stands for empirical parameters and e is the base of natural logarithms. Furthermore, $m_{i,j}$ indicates the number of occurrences of the two words in the parallel corpus.

3.2 Auto-associative Memory

Auto-associative memory refers to the form or concept of two types of data related to each other have the specific form of knowledge stored in memory. According to this concept, the co-occurrence word table is used as a bridge between the data of two languages. In this paper, the auto-associative memory method is applied to the neural network, which can be described as:

$$f : R^{|v_s| \times n} \rightarrow R^{|v_t| \times n}. \tag{4}$$

$$f(v_{j|s}) = v_{T-index(j)|t}. \tag{5}$$

As shown in Eq. (4), the source language vector of the input can be associated with the correlation vector of the target language. In multilingual tasks, it is only necessary to establish a co-occurrence list among different languages, and the vectors of any language can be associated with any other language.

4 Multilingual Text Classification of Convolutional Neural Network Based on Auto-associative Memory

In the convolutional neural network, the combination of convolution kernel of different sizes can learn the expression way of phrases with a different number of words. Word2vec can generate precise word vector expression, but semantic information of a word needs to be calculated with a whole vector.

As the co-occurrence vocabulary can be regarded as the generation memory of multilingual semantics, text semantic in the source language can be associative to other target languages by mnemonic mapping. The input of the single language is extended to the input memory that contains multiple languages through the auto-associative memory. All samples in different languages space can be calculated by auto-associative memory in multidimensional space. The data generated from the associative memory relationship have a complementary semantic relationship, and the convolutional neural network can be used to extract the salient features and ignore the information that has less effect on the classifier. Therefore, this paper proposes a multilingual text categorization algorithm based on auto-associative memory and convolutional neural network.

4.1 Convolutional Multilingual Mapping Layer

According to the auto-associative memory relationship, the word has a corresponding semantic word in any other language, and each sample data of the input is supposed to be extended as shown in the following Fig. 2.

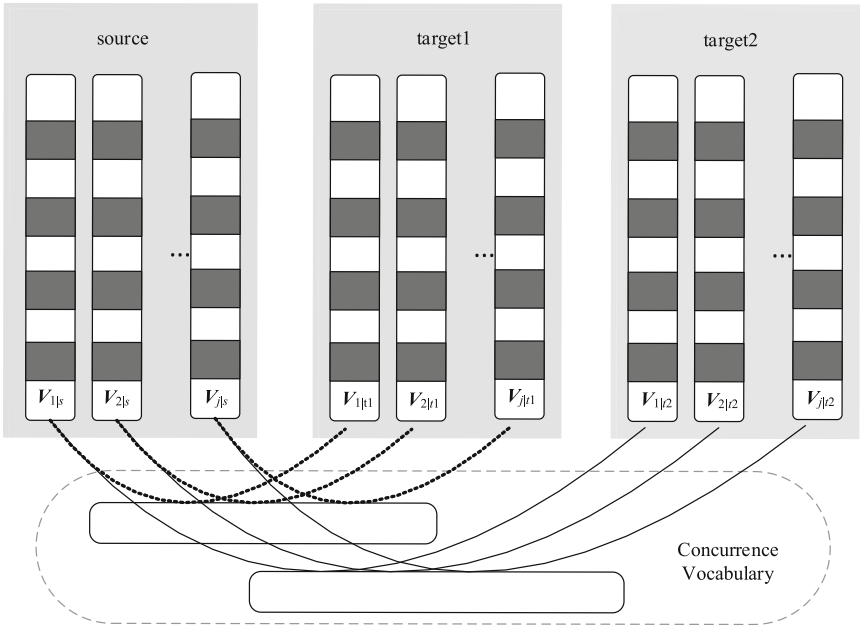

Fig. 2. Language extension based on auto-associative memory

The first frame on the right side of the graph is the output word vector of the source language text, and the co-occurrence vocabulary could find the semantic association words of each word corresponding to the target language. We look for the word vector of the associated word in the same position for the target language space. The text matrix composed of a splice source language text matrix and auto-associative memory is expressed by Eq. (6):

$$d = \begin{bmatrix} d_{i|s} \\ d_{i|t_1} \\ d_{i|t_2} \end{bmatrix}, d_{i|s}, d_{i|t_1}, d_{i|t_2} \in R^{m \times k}, d \in R^{3m \times k} \tag{6}$$

This model adapt to any language resources. The semantic mapping vector based on the input language is taken as a memory to help the model generate multilingual space, and convolutional neural network can extract the local features in that space.

4.2 Convolutional Neural Network Structure

The proposed convolutional neural network model is shown in Fig. 3:

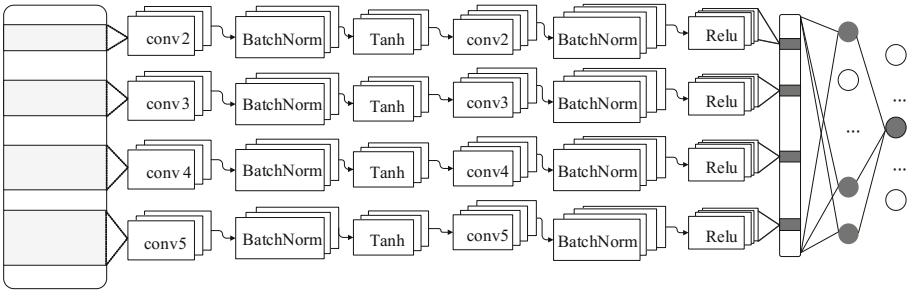

Fig. 3. Extended TextCNN models

As shown, the extended model consists of 9 layers. The input layer is a text matrix being composed of word embedding vectors. Let $d_{i:i+j}$ refer to the concatenation of words $d_i, d_{i+1}, \cdots, d_{i+j}$, a window that moving backward from the first row of the input matrix. Each convolution kernel is a window with h rows and k columns, and it is applied to produce a new feature.

$$s_i = w * d_{i:i+h-1} + b, w \in R^{h \times k}. \tag{7}$$

Among them, d represents input, w represents the weight parameter of convolution kernel, and b is a bias item. The window width k is consistent with the width of the word vector. Convolution kernel with row 2 represents it can extract phrase information composed of two words. In the same way, the convolution

kernel of other lengths can also represent the extraction of phrase characteristics of the corresponding number of words.

The calculation process of the batch normalization layer is calculated as follows:

$$\mu_\beta = \frac{1}{m}\sum_{i=1}^{m} s_i, \tag{8}$$

$$\sigma_\beta^2 = \frac{1}{m}\sum_{i=1}^{m}(s_i - \mu_\beta)^2, \tag{9}$$

$$\tilde{s}_i = \frac{s_i - \mu_\beta}{\sqrt{\sigma_\beta^2 + \varepsilon}}, \tag{10}$$

$$z_i = \gamma \tilde{s}_i + \beta. \tag{11}$$

where, μ_β is the mean value of the input, and σ_β^2 is the variance of input, m is the quantity of input data. Equation (10) is used to compress the distribution range of data [7] so that results have fixed mean and variance. Neural network is a parameterized model essentially, different data distributions are supposed to be better fitted with different parameter models. When the distribution gap between training data and test data is large, the performance of the model will be greatly reduced. In addition, as the number of network layers increases, the impact of changes in lower-layer network parameters on higher-level networks will increase. Normalization of data can solve this problem, but the expression ability of network will be weakened. So Eq. (11) is used to zoom and translate normalized data.

Activation layer functions is f_{tanh} and f_{relu}.

$$f_{\tanh}(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}, \tag{12}$$

$$f_{relu}(z) = \begin{cases} z & (if z > 0) \\ 0 & (if z < 0) \end{cases}. \tag{13}$$

The output range of $f_{tanh}(z)$ is $[-1, 1]$, and the function value is saturated when the absolute value of the input data $\boldsymbol{X}$ is very large, which makes the function close to the biological neuron and can suppress or stimulate the information transmission of the neuron. And since it has a mean value of 0, it converges faster. Relu function solves the problem saturation function encountered, that is, when the function value is saturated to 0 or 1, the network layer's derivative is close to 0, so it will affect the reverse transfer of the gradient. In high layer, Relu function is suitable to ensure the transmission of gradient and alleviate the problem of gradient disappearance.

The Chunk-Max Pooling method is adopted in pooling layer, which means to divide the vectors into equal length segments. After that, only the most significant eigenvalues of each subsegment is preserved.

The last layer is the output layer of the classification result, which combined the full-connection layer with the softmax layer constitutes a softmax regression

classifier. Assumed that the convolution layer, the activation layer and the pool layer can map the vector from the original input to the hidden layer feature space, then the function of the full connection layer shall map the distributed feature vector in the hidden layer space to the sample label to complete the classification task.

The formula for the softmax function is:

$$p_i = \frac{e^{y_i}}{\sum\limits_k e^{y_k}}, \tag{14}$$

where y_i represents the output of the i-th unit on the previous layer, and the value of p_i is the output of the i-th neuron on the output layer, which represents the probability that the classification label belongs to the i-th class.

5 Experiments

5.1 Datasets

The experimental data set of this paper is collected from a multilingual document management system project, including 13 categories scientific abstracts which contain more than 90,000 texts in Chinese, English, and Korean. Each language contains more than 30,000 texts, which form a translation corpus for content alignment. The data set is randomly divided into ten parts, of which are used as the training set, and the rest are used as the test set, repeat it and take the average value of the experimental results.

5.2 Experimental Setup and Baselines

We use accuracy and cross entropy to verify the performance of the proposed model. The concept of cross entropy comes from information theory. Suppose that the same sample set has two kinds of label probability distribution p and q. The cross entropy represents the average encoding length from the truly distributed p to the error distribution q, so that the cross entropy loss function can measure the similarity between the p and the q. According to the duality of entropy, when p and q are equally distributed, the following formula is minimized.

$$H = -\sum_{\mathrm{i}} p_i \log_2 q_i. \tag{15}$$

For the dataset, we compare our method against state-of-the-art methods including Bilingual Word Embedding (BWE) [17], Canonical Correlation Analysis (CCA) [1], Machine Translate (MT) [4].

(1) Bilingual Word Embedding (BWE) disarranges the mixed training and use the TextCNN algorithm as the classifier to measure the effect of this algorithm.

(2) Canonical Correlation Analysis (CCA) map two languages into a single space for classification and perform correlation calculations on the word vectors between the two languages to obtain a linear transformation matrix between the word vectors. We average classification results between each two languages pairs.
(3) Based on Machine Translate (MT) model, Google translation is used to translate Chinese and Korean abstracts into English for training.
(4) In this paper, multilingual text classification of convolutional neural network based on auto-associative memory (Me-CL-CNN) has built convolution kernel set in [2, 3, 4, 5]. Proper phrase expression can give expression to semantic combination and weaken unimportant features. The depth of convolution kernel is 64, and the dropout rate of hidden nodes is equal to 0.5. According to empirical value. We choose L2 regularization method and set the regularity coefficient equal to 0.05.

5.3 Accuracy for Different Embedding Dimensions

In different Natural Language Processing tasks, there are different requirements for the length of the word vector, so the purpose of the first experiment is to determine the length of the word vector that fits the problem of short text classification. The TF-IDF algorithm is used to weigh word vectors to get text vectors [16] as input of classifier commonly used in machine learning, including SVM, KNN and RBF.

To tell the significant difference in accuracy of different embedding dimensions, We first compare the classifer in terms of classification accuracy on our Chinese datasets. The results are shown in Table 1, and the best dimension of each classifer is highlighted in bold.

Table 1. Classification accuracy (%) for different embedding dimensions

Embedding dimensions	Accuracy (%)		
	KNN	SVM	RBF
50	59.83	68.03	65.21
100	61.50	70.91	67.45
150	62.16	71.79	**69.83**
200	**62.24**	**72.03**	69.69
250	62.24	72.01	69.67

As can be seen, the classification accuracy of the text increases slowly with the increase of the word embedding dimension, which indicates that the higher dimension of the word embedding is more able to express the semantic information. When the vector dimension exceeds 200, the classification performance will no longer continue to increase. Without loss of generality, we set length of input word embedding is 200 for our experiment considering classfication performance.

5.4 Experiments and Analysis of Results for Multilingual Text Classification

We compare our method against the selected baselines in terms of classification accuracy, and the results of the experiment are shown in Table 2:

Table 2. Classification accuracy (%) and cross entropy on datasets

Method	Result evaluation indexes	
	Cross entropy	Accuracy (%)
BWE	0.97	78.53
CCA	0.64	80.8
MT	1.12	76.5
Me-CL-CNN	**0.45**	**82.41**

Generally, the bilingual transformation method based on the Machine Translation and the Canonical Correlation Analysis is similar. The former is the translation of the text, and the latter uses the relevance of the word vector itself to carry out the language transformation. In a specific field of language, the Machine Translation tool is very poor in the translation of special terms in the scientific literature, so only 76.5% of the correct rate is obtained, while CCA has an average accuracy of 80.80%. The method based on BWE is slightly worse than CCA. This is because the alignment statements of the three languages produced differences in grammar and word position during the random fusion of windows, resulting in poor training of word vectors. The width of the set context window contains three kinds of words, so it will have an obvious influence on the semantic expression of the documents.

For the single label datasets, the model we proposed shows stronger performance in classification. The sensitivity of the convolutional neural network to local features makes it more capable of complying with the multilingual semantic information based on the auto-associative memory model, thus completing the classification task well. The advantage of the model is that it does not need the aid of external tools, only needs parallel corpus to obtain the relationship between multilingual features. It has good extension ability, strong generalization ability and strong portability for each language. Moreover, the model can obtain any language text to be input, and it will help the model to generate semantic category labels by using the mapping vector of the single language semantic, so this model can counteract the effect of data imbalances due to the scarcity of language resources.

6 Conclusion

This paper proposes a way constructing the semantic auto-associative memory relationship between multilingual languages in combination with the co-occurrence degree of multilingual words in the parallel corpus and the distance

of word embedding. Without language restrictions, we provide a basis for the detection and fusion of multilingual text semantics. Moreover, this paper extends the text classification model of convolutional neural network and superimpose two convolution, pooling and activation layers to extract the higher level abstract semantics. The normalization layer is added to adjust and speed up the model, which extends the input of every short text by using auto-associative memory. The experiment illustrates that the auto-associative memory model and the extended convolutional neural network can extract the deep semantic information of the multilingual feature, which can be performed in a particularly efficient way for the classification of multilingual short text.

Acknowledgment. This research was financially supported by State Language Commission of China under Grant No. YB135-76. We would like to thank editor and referee for their careful reading and valuable comments.

References

1. Faruqui, M., Dyer, C.: Improving vector space word representations using multilingual correlation. In: Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, pp. 462–471 (2014)
2. Gliozzo, A., Strapparava, C.: Exploiting comparable corpora and bilingual dictionaries for cross-language text categorization. In: Proceedings of the 44th Annual Meeting of the Association for Computational Linguistics, pp. 553–560. Association for Computational Linguistics (2006)
3. Guo, J., Che, W., Yarowsky, D., Wang, H., Liu, T.: Cross-lingual dependency parsing based on distributed representations. In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics, Long Papers, vol. 1, pp. 1234–1244 (2015)
4. Hanneman, G., Lavie, A.: Automatic category label coarsening for syntax-based machine translation. In: Proceedings of the Fifth Workshop on Syntax, Semantics and Structure in Statistical Translation, pp. 98–106. Association for Computational Linguistics (2011)
5. Harris, Z.S.: Mathematical structures of language. In: Tracts in Pure and Applied Mathematics (1968)
6. He, W.: Research on wordNet based Chinese English cross-language text similarity measurement. Master's thesis, Shanghai Jiao Tong University (2011)
7. Ioffe, S., Szegedy, C.: Batch normalization: accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167 (2015)
8. Kim, Y.: Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882 (2014)
9. Liu, J., Cui, R.Y., Zhao, Y.H.: Cross-lingual similar documents retrieval based on co-occurrence projection. In: Proceedings of the 6th International Conference on Computer Science and Network Technology, pp. 11–15 (2017)
10. Luo, Y., Wang, M., Le, Z., Lu, X.: Bilingual latent semantic corresponding analysis and its application to cross-lingual text categorization. J. China Soc. Sci. Tech. Inf. **32**(1), 86–96 (2013)
11. Mikolov, T., Le, Q.V., Sutskever, I.: Exploiting similarities among languages for machine translation. arXiv preprint arXiv:1309.4168 (2013)

12. Peng, Z.: Research of cross-language text correlation detection technology. Master's thesis, Central South University (2014)
13. Potthast, M., Stein, B., Anderka, M.: A Wikipedia-based multilingual retrieval model. In: Macdonald, C., Ounis, I., Plachouras, V., Ruthven, I., White, R.W. (eds.) ECIR 2008. LNCS, vol. 4956, pp. 522–530. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-78646-7_51
14. Sorg, P., Cimiano, P.: An experimental comparison of explicit semantic analysis implementations for cross-language retrieval. In: Horacek, H., Métais, E., Muñoz, R., Wolska, M. (eds.) NLDB 2009. LNCS, vol. 5723, pp. 36–48. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-12550-8_4
15. Tang, G., Xia, Y., Zhang, M., Zheng, T.: Cross-lingual document clustering based on similarity space model. J. Chin. Inf. Process. **26**(2), 116–120 (2012)
16. Tang, M., Zhu, L., Zou, X.C.: Document vector representation based on word2vec. Comput. Sci. **43**(6), 214–217 (2016)
17. Vulić, I., Moens, M.F.: Monolingual and cross-lingual information retrieval models based on (bilingual) word embeddings. In: Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 363–372. ACM (2015)