ParagraphVector Based Retrieval Model for Similar Cases Recommendation

Yifei Zhao, Jing Wang, Fei-Yue Wang
The State Key Laboratory of Management and Control for Complex Systems
Institute of Automation, Chinese Academy of Sciences
Beijing 100190, China
{yifei.zhao, wangjing2014, feiyue.wang}@ia.ac.cn

Xiaobo Shi
Institute of Smart Healthcare System
Qingdao Academy of Intelligent Industries
Qingdao, China
xiaobo.shi@qaii.ac.cn

***Abstract*—Internet inquiry is playing an increasingly important role as the complement of the traditional medical service system, especially the similar cases recommendation. It can not only save the patients' waiting time, but also make use of the historical resources, for many cases with the same purpose have been solved perfectly. However, because of the diversity and non-standard of the patients' descriptions, the inquiry platform cannot find the cases with similar semantic easily. Most traditional retrieval methods require the overlap of two sentences, and this is not suitable with the diversity and non-standard descriptions. In this paper, we try to utilize the sentences' semantic representation in a continuous space to understand the cases, and then recommend the similar cases. We also incorporate it into query likelihood language models, trying to get better results. Our experimental data are all collected from a real internet inquiry platform, and the results show that our methods significantly outperform the state-of-the-art translation based methods for similar cases recommendation.**

***Keywords*—internet inquiry; similar cases recommendation; distributed representation; data mining**

I. Introduction

In recent years, internet inquiries play an increasingly important role as the supplement and optimization of the traditional medical service system. Internet inquiries platforms are those on which patients can describe their questions or illness, and doctors give their solutions or suggestions. Similar cases recommendation is necessary for this kind of inquiry platform for the reasons, a) there have been plenty of cases with the same purpose which have been solved perfectly in the platform, and these cases could give the patients more references, b) waiting for the doctors' answering is harmful to the users' experience, and some perfectly solved cases should be better than a single doctor's answer which the quality cannot be guaranteed, c) these cases should be fully used from the perspective of resources utilization.

Some traditional information retrieval models may have some effects for this kind of similar cases recommendation. However, when a new case occurs, using traditional methods is sometimes difficult to recommend a perfectly solved case to it for the diversity and non-standard of the patients' descriptions. Some excellent retrieval methods, such as vector space model (VSM) [1], Okapi BM25 [2], query likelihood language model (LM)[3] and so on, require the overlap of two sentences, i.e. the two sentences must have some co-occurrence keywords. When come to the situation of similar cases recommendation, they may be not able to work well. First, there is a problem called "lexical gap" [4, 5], i.e. the words with same meaning but different descriptions, especially most patients are not familiar with professional medical vocabulary, there are lots of dialects and popular words in their description. For example, some may describe insomnia as "lost sleep" while others may be "cannot sleep". Second, except for words, the patients may use various sentences to express the same thing, and most the descriptions are non-standard. With these common phenomena which have to be dealt with, the traditional methods cannot work well, for they have no more mining capacity at the semantic level.

Some work based on statistical translation methods [6-9] have been done to solve the problem of "lexical gap". They are trying to get the semantic similarity of two words from the perspective of word alignment probability. Specifically, they are trying to get many pairs of sentences, which express the same meaning with different descriptions. This is similar to the "parallel corpus" which is used to estimate the word-to-word translation probabilities of two different languages, and the distinguish is that it is monolingual, and the word alignment probabilities gotten is not the translation probability but the semantic similarity of two words. Combining the results with the retrieval models can ease the problem of "lexical gap" in a certain degree, and the retrieval results outperform some practical traditional methods [6, 7].

However, there are two problems about the translation based methods, a) it is not easy to get the "mono parallel corpus" with high quality. Some work get the question-question parallel corpus through similar answers [6], some work see the question-answer pairs directly as parallel corpus [8], all these methods cannot guarantee high quality mono parallel corpus, and without high quality mono parallel corpus accurate word alignment probabilities cannot be gotten. b) even if the word alignment probabilities with high quality could be gotten, it could only get the words semantic similarity, but what we finally need is the cases semantic similarity. Because of the reasons mentioned above, although translation based methods have achieved a certain effect, they are not so particularly satisfied.

Recently, word embedding is an effective way of representing a word from the perspective of semantic [10-13] In their work, each word is represented by a fixed-length vector which is concatenated or averaged with other word vectors in a context, and the resulting vector is used to predict other words in the context. These techniques have gotten satisfied results, and some researchers extended them to phrase-level or sentence-level representations [13, 14]. They can represent a phrase, a sentence, even a document with a fixed-length semantic vector. In this paper, combining the query likelihood language model framework, we make use of the semantic representation of sentence to recommend the similar cases. Compared to the translation based methods which usually assume that the question-answer pairs are "parallel", our models is more practical and easily to implement. In addition, we get the semantic similarity from the overall perspective. We collected data from a real internet inquiry platform, and the experimental results show that our proposed models for similar cases recommendation outperforms the translation-based models significantly.

II. Related Works

There have been many works about information retrieval (IR) would be helpful for similar cases recommendation. This section we would first introduce some related works, especially those works based on translation models, which could address the "lexical gap" problem to some extent. Then some related works about word embedding and paragraph vector in a continuous space would be given, which could solve "lexical gap" problem more effectively.

A. *Information Retrieval Models*

Traditional IR models require the co-occurrence words, especially key words' co-occurrence, such as vector space model [1], Okapi BM25[2], query likelihood language model [3] and so on. They cannot work well when meeting "lexical gap", which could be seen everywhere. The patients' description varies much especially in internet inquiry platform. In an attempt to better retrieve from the perspective of semantic, many significant research efforts have been conducted over the past few years[7-10, 15-17,28-30]. Most of the works focus on recommending the similar questions for the users' queries, and mainly based on the statistic translation methods. Jeon et al. [6, 7] first proposed word-based translation model (WTM) for similar questions retrieval. They used question-answer pairs as mono parallel corpus, automatically fixing the "lexical gap" problem by getting the word-to-word translation probabilities. They compared this model with VSM, BM25 and LM, and the results showed the advantages. To further improve WTM, Lee et al. [9] removed the noise of original data by Textrank [18] to get a more compact parallel corpus consisting with important terms. The experimental results demonstrated better retrieval performance. Xue et al. [8] combined WTM with query-likelihood language model, and proposed a word-based translation language model (WTLM). The purpose was to reduce the problem of self-translations. Although very low or very high self-translations are still possible, this modification gave significant improvements over the original translation model. To capture the contextual information, which could be helpful to different words with the same meaning, Zhou et al. [10] proposed a phrase-based translation model. This model translates phrases as a whole, some contextual information would be included, thus the more accurate translations can better improve the retrieval performance.

Different from the traditional methods mentioned above that assume question-answer pairs are "parallel corpus", which is not so reasonable in practice, our methods and strategies rely on no assumptions. We directly incorporate the continuous word embedding and paragraph vectors into query-likelihood language models for better similar cases recommendation.

B. *Word Embedding*

Word embedding is often the accessory products when training language model. It is an idea about distributed representation. It is also called "word representation". Because of the outstanding performance in various natural language processing work, learning high-quality word representations has attracted researchers' increasing attention, and deep learning were widely applied [20, 22, 24-27].

Bengio et al. [21] firstly proposed a probabilistic neural network language model (NNLM) for word representations. They construct a three layers neural network to represent an n-gram model [21]. The input is the continuous i-1 words' feature vectors, the output is the i-th word's feature vector. What should be mentioned here is that, the input is also parameters which need to be optimized. After optimization, not only the language model, but also the word embeddings came into being. Later, many other researchers started their work according to this idea. Ronan Collobert and Jason Weston rate for the probabilities that n consecutive word coocurrences [22]. The higher the rating, the more normal the sentence is. In fact, their purpose was to complete other tasks in NLP, while not get the word embeddings nor a language model, so the word embeddings they got have two differences. One is they only have lowercase words, and the other is that it is second optimization result. Andriy Mnih and Geoffrey Hinton applied deep learning into NLP [12, 23] after deep learning was proposed in 2006 [24]. [23] proposed three models, they modified the energy function from the most basic RBM, and then got Log-Bilinear model. [12] proposed a hierarchical idea to replace the matrix multiplication in [21], and the speed was improved. Bengio et al. [21] pointed out that the recurrent neural network could be used to reduce the number of parameters, and according this Mikolow proposed RNNLM [25]. He make full use of all context to predict the next word, and got a much better result. Eric H. Huang used global context to train, and got multiple vectors for polysemous words [26]. The most direct and most widely used is Tomas Mikolov's work in 2013 [27]. He launched the open-source toolkit word2vec, it could get word vectors simply and efficiently. In 2014, Quoc Le and Tomas Mikolov proposed Paragraph Vector [27], extended the distributed representations for words to phrase-level, sentence-level and document-level. It is an unsupervised model that learns fixed-length feature representations from variable-length pieces of texts. Similar with the idea mentioned above, the model represents each paragraph by a dense vector which is trained to predict words in the paragraph. Here papragraph means a variable length unit including phrase, sentence and document. In this paper we adopt it to incorprate word embeddings into the language models, and get the case representations in the continous space, trying to get better recommendation performance. To our knowledge, there are few works combine word (case) embeddings with query-likelihood languge models for similar cases recommendation in internet inquery platform.

III. Paragraph Vector Based Models

Based on the retrieval models and distributed representations for semantics mentioned above, this section we would regard a case as a paragraph, and then get the distributed representations for each case and each word, i.e. paragraph vectors and word vectors. Then we would give some models combining the semantic vectors, which also consider the characteristics of the internet inquiry platform data.

A. Paragraph Vector based Model

Based on the idea in [21], we took the strategies in [20] to get the vectors for each word and each paragraph at the same time in the same continous space. Here the vectors are the semantic representations for the words and sentences. More specifically, the paragraph vector was concatenated with several word vectors from the paragraph and the following word in the given context was predicted. These semantic vectors are all trained by the stochastic gradient descent and backpropagation. The word vectors are shared, while the paragraph vectors are unique among paragraphs. At prediction time, the paragraph vectors are inferred by fixing the word vectors and training the new paragraph vectors until convergence.

After converting the paragraph in text form to vectors in math form, it is easily to employ the distance of the vectors to measure the similarity of two cases. In paragraph vector based model (PVM), the similarity between query ***Q*** and historical cases ***D*** could be represented as follows,

$$sim(Q,D)=\cos(Q,D)=\frac{\sum_{i=1}^{n}Q_i\times D_i}{\sqrt{\sum_{i=1}^{n}(Q_i)^2}\times\sqrt{\sum_{i=1}^{n}(D_i)^2}} \tag{1}$$

where n indicates that the semantic vectors are with n dimensions, Q_i and D_i is i^{th} dimension of the semantic vectors which could represent the case which we cannot, and needn't know exactly what semantic it really is.

B. Paragraph Vector based Language Model

Query-likelihood language model framework based retrieve models have achieved many good results in finding similar sentences [6-7, 15-17]. In this framework, the similarity between a query Q and a historical case D is given by the probability of the generating Q from D. i.i.d sampling and unigram language model was widely used in practice, which could be represented as follows,

$$sim(Q,D)\approx P(Q\mid D)=\prod_{\omega\in Q}P(w\mid D) \tag{2}$$

Here w represents a single word. To avoid zero probabilities and estimate more accurate language models, Jelinek-Mercer smoothing method [3] is adopted for its good perfomance and low computational complexity. Documents are smoothed using a background collection and λ here is smoothing parameter,

$$P(w\mid D)=(1-\lambda)P_{ml}(w\mid D)+\lambda P_{ml}(w\mid Conll) \tag{3}$$

In most traditional query likelihood language models, the maximum likelihood estimation is gotten by counting, namely

$$P_{ml}(w\mid D)=\frac{\#(w,D)}{|D|} \tag{4}$$

$$P_{ml}(w\mid Conll)=\frac{\#(w,Conll)}{|Conll|} \tag{5}$$

Here $\#(w, D)$ is the frequency of occurrence for word w in historical case ***D***, and $|D|$ is the length of ***D***. This method for P_{ml} relies heavily on the occurrence of word w. The likelihood estimation is gotten from the external form. With word embedding, we could get word vector and paragraph vector representation in the same continuous space from the semantic aspect. The relation of word w and case ***D*** should be obtained more directly from the inner meaning. In paragraph vector based language model (PVLM), the maximum likelihood estimation P_{ml} in equation (3) should be replaced by $sim(w,D)$, so

$$P(w\mid D)=(1-\lambda)sim(w,D)+\lambda sim(w,Conll) \tag{6}$$

$sim(w,D)$ denotes the semantic similarity of word w and case ***D***, which is the cosine distance between the feature vectors. And, the similarity of two cases is also computed by the conditional probability.

IV. Experiments

In this section, experiments are conducted on a real internet inquiry platform data. We reproduce some classic state-of-the-art methods which are based on translation models, then use our proposed retrieval models to demonstrate the effect.

A. Dataset

Our original data came from ask.39.net. All cases were written in Chinese, and each case consists of 4 parts: title, disease description, doctor's answer and the department the case belongs to. The original data was full of noise, so we cleaned it by limiting the cases' length, the legitimacy of characters and so on. After data cleaning, we got 1.25 million cases which could be used. We made disease description and doctor's answer as parallel corpus, then the word-to-word translation probabilities could be gotten by the open-source toolkit GIZA++. We made the four parts as a whole corpus, and then learnt case vectors and word vectors through the models proposed in [20]. Furthermore, the semantic similarity was obtained by the cosine distance.

Because there was no public test collection about similar cases, we adopted the methods related to [16, 17] to construct test data. In the 1.25 million cases, we specified 10 departments, and took all the cases belonged to them out. For each department, we chose a representative query, i.e. the representative cases, and then we ran 3 different search engines and gathered the top 200 similar cases from each search result. After remove the repeat cases, we got 1204 returned cases. Annotators are all professional medical staff from Peking Union Medical College Hospital, and our annotate rules are as following, a) if the two cases are very similar or has references for the other, annotate the relationship as "3", if they have no relationship annotate "1", and of course the left cases are annotated as "2"; b) annotators manually judged the relevant of the results and each query. Each pair was judged by three annotators, if two of them had the same opinion that they were considered as similar cases, otherwise, the third annotators would join in. Finally the test data statistics are as follows,

TABLE I. STATISTICS ON THE TEST DATA

Query cases	**Returned cases**	**Similar cases**		
		1	2	3
10	1204	117	95	992

B. Evaluation Metrics

We evaluate the performance of the models using two different metrics which are commonly used in information retrieval.

Mean Average Precision (MAP):

$$MAP(Q_t) = \frac{1}{|Q_t|} \sum_{q \in Q_t} \frac{1}{m_q} \sum_{k=1}^{m_q} Precision(R_k) \tag{7}$$

m_q is defined as the number of questions related to the query $\boldsymbol{q}$. R_k is defined as a set which contains the first k questions in the query result. $Precision(R_k)$ is defined as the ratio of R_k and all questions related to $\boldsymbol{q}$. Thus, $MAP(Q_t)$ indicates the average level of the entire test results.

Precision @N (P@N): is defined as the precision of the system about the first N returned cases to the queries. The precision of single query q is,

$$P(\mathbf{q})@N = \frac{k}{N} \tag{8}$$

k is the number of similar cases in the first N returning cases to query. At the same time, N denotes the number of all results that the search system returns. So the system precision about the entire test set is the average $P(q)@N$ of all queries,

$$P@N = \frac{\sum_{q=1}^{Q_t} P(\mathbf{q})@N}{Q_t} \tag{9}$$

C. Results and Analysis

This section we compare our models with some state-of-the-art methods. The results are in TABLE II.

TABLE II. COMPARISON WITH DIFFERENT METHODS FOR SIMILAR CASES RECOMMENDATION

Models	**Precision@20**			**MAP**
	1	2	3	
WTM	10.55%	14.57%	74.87%	58.61%
TRLM	11.17%	9.14%	79.70%	66.92%
PVM	13.00%	6.00%	81.00%	72.30%
PVLM	6.5%	7.5%	86.00%	76.09%

The first line, WTM means word-based translation model proposed in [8], the second line TRLM means the translation based language model in [9], and the third line PVM is paragraph vectors based model, while the fourth line PVLM is paragraph vectors based language model. There are some clear trends can be found in the result of TABLE II.

If we strictly believe that only the score 3 could prove the two cases are similar, word-based translation language model (TRLM) outperforms word-based translation model (WTM) in [7] significantly (row 1 vs. row 2), on both precision and MAP. This means self-translation indeed has the impact on retrieval results in the framework of translation based retrieval models. This conclusion is consistent with the conclusion in [8].

Our proposed distributed representations based models (PVM and PVLM) significantly outperforms the state-of-the-art WTM and TRLM (row 3 and row 4 vs. row 1 and row 2), on both precision and MAP. This means word embedding can grasp semantic of words better than translation-based methods.

The model combining paragraph vectors with query-likelihood language model outperforms simply using the cosine distance of semantic vectors to measure the similarity of two cases (row 3 vs. row 4), this means directly using the paragraph vectors is slightly rough. To better mining the semantic relevance, the word-level semantic information should be combined with query-likelihood language model.

Finally, a comparison example in TABLE III is to show our models' performance on similar cases recommendation (all translated from Chinese). The original case is about liver cancer, and the patient wants to know three points of it: effective treatments, symptoms and characteristics, and whether it could be cured thoroughly. Analyzing the recommended cases according to the subject and its three points, the result indicates that the cases recommended by WEM are more reasonable than those recommended by WTM.

TABLE III. OMPARISON EXAMPLES. TOP 3 RECOMMENDED CASES ARE LISTED FOR THE QUERY

	WTM	PVLM
Query	I want to know what **the effective treatments** of **liver cancer** are, and what **the symptoms and characteristics** are? Can it **be cured thoroughly**?	
1	How to treat advanced **liver cancer** which has been transferred to the lungs?	I want to know what the **symptoms and characteristics** of **liver cancer** are. What does it cause? And could it **be cured thoroughly**?
2	Could the patient of **liver cancer** taking ganoderma lucidum toad soup?	I am suffering from **liver cancer** and which causes troubles in my work, I took the pills but the effect was not obvious. What might cause the **liver cancer**? What are the **effective treatments** of it? And what's the **symptom** of it?
3	Hello doctor wang, Could radiotherapy play a role for my father's mid **liver cancer** and pleural effusion?	I want to know whether the **liver cancer** could **be cured**. What's the **treatment** of it? And what does it cause?
4	My grandpa has an advanced **liver cancer** which has transferred to stomach and he didn't want to eat. What kind of medicine could comfort him and have benefits to his health?	I want to know what may cause the **liver cancer**. And what's the **effective treatment** of it? Should it **be cured thoroughly**?

V. CONCLUSIONS AND FUTURE WORK

In this paper, we proposed novel paragraph vector based information retrieval models for similar cases recommendation. Compared with traditional translation based models, the proposed approach is more effective in capturing the cases similarity from the perspective of semantics. Experiments conducted on real internet inquiry platform data demonstrate that the paragraph based models significantly outperforms the state-of-the-art translation based models.

There are some ways in which this research could be continued. First, consider the weighting of the keywords, especially medical vocabularies, because they are more important in determining what a case focuses on. Second, make the evaluation criteria more standardized. This paper we evaluate the results through professional medical staff, which cannot avoid subjectivity. Like the example in TABLE III, the bold words are the points which could demonstrate the similarity, we may make the similarity quantitative to know more intuitively and accurately how similar two cases are.

ACKNOWLEDGMENT

This work was supported by Guiyang Longmaster Information & Technology Co., Ltd. We sincerely appreciate them. And we also thank the friends from Peking Union Medical College Hospital for their suggestions on medical science and insist on the boring annotating work.

REFERENCES

[1] G. Salton, A. Wong, and C.-S. Yang, "A vector space model for automatic indexing," *Communications of the ACM,* vol. 18, pp. 613-620, 1975.

[2] A. Singhal, G. Salton, M. Mitra, and C. Buckley, "Document length normalization," *Information Processing & Management,* vol. 32, pp. 619-633, 1996.

[3] C. Zhai and J. Lafferty, "A study of smoothing methods for language models applied to ad hoc information retrieval," in *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, 2001, pp. 334-342.

[4] A. Berger, R. Caruana, D. Cohn, D. Freitag, and V. Mittal, "Bridging the lexical chasm: statistical approaches to answer-finding," in *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, 2000, pp. 192-199.

[5] A. Berger and J. Lafferty, "Information retrieval as statistical translation," in *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, 1999, pp. 222-229.

[6] J. Jeon, W. B. Croft, and J. H. Lee, "Finding semantically similar questions based on their answers," in *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, 2005, pp. 617-618.

[7] J. Jeon, W. B. Croft, and J. H. Lee, "Finding similar questions in large question and answer archives," in *Proceedings of the 14th ACM international conference on Information and knowledge management*, 2005, pp. 84-90.

[8] X. Xue, J. Jeon, and W. B. Croft, "Retrieval models for question and answer archives," in *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, 2008, pp. 475-482.

[9] J.-T. Lee, S.-B. Kim, Y.-I. Song, and H.-C. Rim, "Bridging lexical gaps between queries and questions on large online Q&A collections with compact translation models," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2008, pp. 410-418.

[10] G. Zhou, L. Cai, J. Zhao, and K. Liu. Phrase-based translation model for questionretrieval in community question answer archives. In *Annual Meeting of theAssociation for Computational Linguistics (ACL 2011)*, pages 653 – 662, 2011.Bengio et al., 2006

[11] R. Collobert and J. Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning," in *Proceedings of the 25th international conference on Machine learning*, 2008, pp. 160-167.

[12] A. Mnih and G. E. Hinton, "A scalable hierarchical distributed language model," in *Advances in neural information processing systems*, 2009, pp. 1081-1088.

[13] Turian, Joseph, L. Ratinov, and Y. Bengio. "Word representations: A simple and general method for semisupervised learning." *In ACL*2010:384--394.

[14] Mitchell, Jeff, and M. Lapata. "Composition in Distributional Models of Semantics." *Cognitive Science A Multidisciplinary Journal*34.8 (2010) :1388–1429.

[15] X. Cao, G. Cong, B. Cui, C. S. Jensen, and C. Zhang, "The use of categorization information in language models for question retrieval," in *Proceedings of the 18th ACM conference on Information and knowledge management*, 2009, pp. 265-274.

[16] X. Cao, G. Cong, B. Cui, and C. S. Jensen, "A generalized framework of exploring category information for question retrieval in community question answer archives," in *Proceedings of the 19th international conference on World wide web*, 2010, pp. 201-210.

[17] H. Duan, Y. Cao, C.-Y. Lin, and Y. Yu, "Searching Questions by Identifying Question Topic and Question Focus," in *ACL*, 2008, pp. 156-164.

[18] Mihalcea, Rada, P. Tarau, and R. Mihalcea. "TextRank: Bringing Order into Text.." *Unt Scholarly Works*, 2004, pp. 404-411.

[19] Hinton, G E. "Learning distributed representations of concepts." *In Proceedings of CogSci* 1986.

[20] Le, Quoc V., and T. Mikolov. "Distributed Representations of Sentences and Documents." *Eprint Arxiv*, 2014.

[21] Y. Bengio, R. Ducharme, P. Vincent, and C. Janvin, "A neural probabilistic language model," *The Journal of Machine Learning Research,* vol. 3, pp. 1137-1155, 2003.

[22] Collobert R, Weston J, Bottou L, et al. "Natural language processing (almost) from scratch," *The Journal of Machine Learning Research*, 2011, 12: 2493-2537.

[23] A. Mnih and G. Hinton, "Three new graphical models for statistical language modelling," in *Proceedings of the 24th international conference on Machine learning*, 2007, pp. 641-648.

[24] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural computation,* vol. 18, pp. 1527-1554, 2006.

[25] T. Mikolov, M. Karafiát, L. Burget, J. Cernocký, and S. Khudanpur, "Recurrent neural network based language model," in *INTERSPEECH 2010, 11th Annual Conference of the International Speech Communication Association, Makuhari, Chiba, Japan, September 26-30, 2010*, 2010, pp. 1045-1048.

[26] E. H. Huang, R. Socher, C. D. Manning, and A. Y. Ng, "Improving word representations via global context and multiple word prototypes," in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, 2012, pp. 873-882.

[27] Mikolov T, Chen K, Corrado G, et al. "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[28] Bernhard D, Gurevych I. “Combining lexical semantic resources with question & answer archives for translation-based answer finding,” *Proceedings of ACL-IJCNLP*, 2009: 728-736.

[29] Singh A. “Entity based q&a retrieval,” *Proceedings of EMNLP*, 2012: 1266-1277.

[30] Zhang K, Wu W, Wu H, et al. “Question retrieval with high quality answers in community question answering,” *Proceedings of CIKM*, 2014: 371-380.