Word Embedding 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*—Similar cases recommendation is more and more popular in the internet inquiry. There have been lots of cases which have been solved perfectly, and recommending them to similar inquiries can not only save the patients' waiting time, but also giving more good references. However, the inquiry platform cannot understand the diversity of description, i.e. the same meaning with different description. This may shield some cases with very high quality answers. In this paper, based on deep learning, we proposed a retrieval model combining word embedding with language models. We use word embedding to solve the problem of description diversity, and then recommend the similar cases for the inquiries. The experiments are based on the data from ask.39.net, and the results show that our methods outperform the state-of-art methods.**

Keywords—internet inquiry; case recommendation; word embedding; data mining

I. INTRODUCTION

In recent years, internet inquiries have bloomed into the supplement and optimization of the traditional medical service system. One of the most important functions is inquiry on line, i.e. one side of the inquiry platform is patients, and the other side is doctors; patients describe their illness and the doctors give professional suggestions on line. To avoid patients' longer waiting time, most internet inquiry platforms provide similar cases recommendation. As a large number of cases with high quality answers have been accumulated on the platform, similar cases recommendation is not only possible, but also a necessity for resource utilization.

The core task of similar cases recommendation is that, given the condition description submitted by the patient, finding the semantic similar cases which could be references in the large scale historical case database. So the task could be converted into computing the semantic similarity of condition description and historical cases. However, because of the problem of "lexical gap" [1], [2], traditional retrieval methods which rely on the total match of words cannot work well now. For example, "what can I do to lose weight" and "how thin my body" (translated from Chinese), the two sentences just describe the same thing while have hardly common words. Traditional methods, like vector space model [3], Okapi BM25 [4], query likelihood language model [5] and so on, require some same words in two sentences, they could not deal the situation of "lexical gap" which can be seen everywhere, for they have no more mining capacity at the semantic level.

In the past few years, the main work about solve the "lexical gap" problem in information retrieval is focus on the statistical translation methods [6], [7], [8], [9]. Since the queries and the answers often express the same meanings with different words, it is natural to use the question-answer pairs as the "parallel corpus" that is used for estimation in machine translation. So, the word-to-word translation probabilities can be learned, and then be combined into the retrieval model. These methods have eased the "lexical gap" problem in a certain degree, and the retrieval results are superior to the traditional methods [6], [7].

However, although the statistical translation based method can find two similar words from the perspective of semantics, it is not intuitive and not so particularly satisfactory. In this paper, we directly convert the words into semantic representation based on deep learning [10]. Deep learning has achieved amazing results in the fields of speech and image [11], [12]. It is also applied into nature language processing, and some results are also made [13], [14]. One is word embedding, which represent a word with a vector from the perspective of semantics [15], [16], [17], [18], [19], [20], [21], [22]. Then we can measure the similarity of two words by computing the distance of the corresponding two vectors. In this paper, we make use of some results of word embedding to understand how "similar" of any two words, then embed the distance into retrieval models to recommend the similar cases for patients' descriptions. Experiments show that our proposed word embedding based language model for similar cases recommendation outperforms the translation-based models significantly.

II. RELATED WORKS

There have been some information retrieval models which could search similar sentences from the perspective of semantics. This section would introduce some methods

mainly based on word translation first, then some related works about word embedding would be given.

A. Information Retrieval Models

At present, some models in community question answering (cQA) could understand semantic to some extent, and many of them are generated from the language model. Query likelihood language models have achieved some good results in finding similar sentences [6], [7], [23], [24], [25], and unigram language model was widely used in practice. For a given query ***q*** and a historic question ***d***, the function for semantic similarity is given as,

$$P(q \mid \mathbf{d}) = \prod_{\omega \in q} P_{LM}(\omega \mid \mathbf{d}) \tag{1}$$

$$P_{LM}(\omega \mid \mathbf{d}) = (1-\lambda) P_{ml}(\omega \mid \mathbf{d}) + \lambda P_{ml}(\omega \mid Conll) \tag{2}$$

where *Conll* is for the whole dataset and P_{ml} represents maximum likelihood estimation. The above equation adopt Jelinek-Mercer smoothing method [5] because of its good perfomance and low computational complexity, and λ here is smoothing parameter.

Jeon et al. first proposed word-based translation model (WTM) for similar questions retrieval [6], [7]. They use question answer pairs as mono parallel corpus, get the word-to-word translation probabilities through statistic translation methods, then use the results to replace the maximum likelihood estimation. They compare this model with VSM, BM25 and LM, and the results show the advantages. On this basis, Lee et al. [26] remove the noise of original data by Textrank [27], then a parallel corpus consists of important words would be gotten. This would make retrieval performance improve greatly. Xue et al. [8] linearly mix two different estimations: maximum likelihood estimation and word-based translation estimation. The purpose is 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.

B. Word Embedding

Word embedding is often the accessory products when training language models with deep learning, it is an idea about distributed representation. It is also called "word representation". Most word embedding works related to language models training.

Bengio et al. construct the language model with a three layers neural network, it is also an *n*-gram model [15]. The input is the continuous *i-1* words' feature vectors, the output is the *i*-th word's feature vector, and the structure of the network can be represented as the following function,

$$f(i, w_{t-1}, \ldots, w_{t-n+1}) = g\left(i, C(w_{t-1}), \ldots, C(w_{t-n+1})\right) \tag{3}$$

where *g* is the neural network and *C(i)* is the *i*-th word feature vector. The function g maps an input sequence of feature vectors for words in context, $\left(C(w_{t-n+1}), \ldots, C(w_{t-1})\right)$, to a conditional probability distribution for the next word w_t. Then optimize the model with stochastic gradient descent method. 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 come into being.

Later, many other researchers start their work according to this idea. Ronan Collobert and Jason Weston rate for the probabilities that *n* consecutive word coocurrences [16], [17]. The higher the rating, the more normal the sentence is. In fact, their purpose is 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 [18], [19] after deep learning was proposed in 2006 [10]. [18] proposed three models, they modified the energy function from the most basic RBM, and then got Log-Bilinear model. [19] proposed a hierarchical idea to replace the matrix multiplication in [15], and the speed was improved. [15] pointed out that the recurrent neural network could be used to reduce the number of parameters, and according this Mikolow proposed RNNLM [20]. 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 [21]. The most direct and most widely used is Tomas Mikolov's work in 2013 [22]. He launched the open-source toolkit word2vec, it could get word vectors simply and efficiently, and in this paper we adopt it.

III. Word Embedding Based Models

Based on the retrieval models and word embedding related works mentioned above, this section we would give some models which considered the characteristics of the internet inquiry platform data.

A. Word Embedding based Model

As mentioned in section II, most retrieval models in cQA are in the framework of query likelihood language models. In this framework, the similarity between query ***Q*** and historical cases ***D*** could be represented as follows,

$$sim(Q, D) \approx P(Q \mid D) = \prod_{\omega \in Q} P(w \mid D) \tag{4}$$

Namely, the similarity of the two cases can be converted into a conditional probability, and *w* represents a single word. To avoid zero probabilities and estimate more accurate language models, documents are smoothed using a background collection,

$$P(w \mid D) = (1-\lambda) P_{ml}(w \mid D) + \lambda P_{ml}(w \mid Conll) \tag{5}$$

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

$$P_{ml}(w \mid D) = \frac{\#(w, D)}{|D|} \tag{6}$$

$$P_{ml}(w \mid Conll) = \frac{\#(w, Conll)}{|Conll|} \tag{7}$$

Here *#(w, D)* is the frequency of occurrence for word *w* in historical case *D*, and |*D*| is the length of ***D***. With word embedding, we could get word vector representation from the semantic aspect, so it is easy to obtain the semantic similarity of any two words. In word embedding based model (WEM), the maximum likelihood estimation P_{ml} in equation (5) is replaced by $\sum_{t\in D} sim(w,t)P_{ml}(t|D)$, so

$$P(w|D)=(1-\lambda)\sum_{t\in D} sim(w,t)P_{ml}(t|D)+\lambda P_{ml}(w|Conll) \quad (8)$$

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

B. Word Embedding based Language Model

The value of *sim*(w,w) may be a flaw in the last model. In general, *sim*(w,w) would be 1 except for polysemous words. The large value would increase the impact of some words in the retrieval model. If the words are those we could remove but unimportant words, the model would not work well. So we try to linearly mix two different estimations: word embedding based estimation and maximum likelihood estimation. So the final function is given as,

$$P(q|D)=\prod_{w\in q} P(w|D) \quad (9)$$

$$P(w|D)=\frac{|D|}{|D|+\lambda}P_{mx}(w|D)+\frac{\lambda}{|D|+\lambda}P_{ml}(w|C) \quad (10)$$

$$P_{mx}(w|D)=(1-\beta)P_{ml}(w|D)+\beta\sum_{t\in D}P(w|t)P_{ml}(t|D) \quad (11)$$

In the word embedding based language model (WELM), the impact of the word semantic similarity could be controlled by β. If we set a very small β, the model would behaves like the query likelihood model, and the importance of matching terms is emphasized.

IV. Experiments

In this section, experiments are conducted on a real internet inquiry platform[1] data. We first reproduce some classic state-of-art methods which are based on translation models, then use our proposed retrieval models to demonstrate the effect.

A. Dataset

Our original data all come from ask.39.net. All cases are written in Chinese, and each case consists of 4 parts: title, disease description, doctor's answer and the department the case belonged to. The original data was so dirty; 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 can 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++[2]. We also make the four parts as a whole corpus, then got word embedding by the open-source toolkit that Tomas Mikolov proposed in 2013 [22]. Furthermore, the semantic similarity of any two words can be obtained by just the cosine distance.

Because there is no publicly test collection about similar cases, we adopt the methods related to [24], [25] to construct test data. In the 1.25 million cases, we specify 10 departments, and take all the cases belonged to them out. For each department, we choose a representative query, i.e. the representative cases, 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 query is with three annotators, if two of them have the same opinion that they are similar, they are similar, 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):

$$MAP(Q_t)=\frac{1}{|Q_t|}\sum_{q\in Q_t}\frac{1}{m_q}\sum_{k=1}^{m_q}Precision(R_k) \quad (12)$$

m_q is defined as the number of questions related to the query ***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 ***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* returning cases to the queries. The precision of single query *q* is,

$$P(\mathbf{q})@N=\frac{k}{N} \quad (13)$$

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

[1] ask.39.net
[2] http://www.fjoch.com/GIZA++.html

precision about the entire test set is the average $P(q)@N$ of all queries,

$$P@N = \frac{\sum_{q=1}^{Q_t} P(\mathbf{q})@N}{Q_t} \quad (14)$$

C. Results and Analysis

This section we compare our models with some state-of-the-art methods. The results are in TABLE II. 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 WEM is our model 1 word embedding based model, while the fourth line WELM is word embedding based language model. There are some clear trends can be found in the result of 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%
WEM	9.55%	7.04%	83.42%	73.86%
WELM	9.05%	7.54%	83.42%	75.74%

1. 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].
2. Our proposed word embedding based models (WEM and WELM) 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.
3. Taken similarity score 1, 2, 3, WELM is slightly outperforms WEM. The improvement is not so obvious as in translation-based models.

Finally, a comparison example is used in TABLE III 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, can it be treated thoroughly. Analyzing the recommended cases according to the subject and its three points, we can find that the cases recommended by WEM are more reasonable than those recommended by WTM.

V. CONCLUSIONS AND FUTURE WORK

In this paper, we proposed novel word embedding based information retrieval models for similar cases recommendation. Compared with traditional translation based models, the proposed approach is more effective in capturing the word similarity from the perspective of semantics. Experiments conducted on real internet inquiry platform data demonstrate that the word embedding 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, in this paper we used the vector to represent the word semantics, the sentence semantics, i.e. the case, may also be represented as a vector, then it could be combined with language models or other strategies.

ACKNOWLEDGMENT

This work was supported by Guiyang Longmaster Information & Technology Co., Ltd. We sincerely thank 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.

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

	WTM	**WEM**
Query	I want to know what **the effective treatments** of **liver cancer** are, and what **the symptoms and characteristics** are? Can it **be treated thoroughly**?	
1	How to treat advanced **liver cancer** which has been transferred to the lungs?	what are **the symptoms and characteristics** of **liver cancer**? How to find **early symptoms** of **liver cancer**?
2	Could the patient of **liver cancer** taking ganoderma lucidum toad soup?	what are **symptoms** of advanced **liver cancer**? what are **characteristics** of advanced **liver cancer**?
3	Hello doctor wang, Could radiotherapy play a role for my father's mid **liver cancer** and pleural effusion?	What are **the effective treatments** of **liver cancer**?

REFERENCES

[1] 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.

[2] 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.

[3] G. Salton, A. Wong, and C.-S. Yang, "A vector space model for automatic indexing," *Communications of the ACM,* vol. 18(11), pp. 613-620, 1975.

[4] A. Singhal, G. Salton, M. Mitra, and C. Buckley, "Document length normalization," *Information Processing & Management,* vol. 32(5), pp. 619-633, 1996.

[5] 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.

[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. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural computation,* vol. 18(7), pp. 1527-1554, 2006.

[11] G. E. Dahl, D. Yu, L. Deng, and A. Acero, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," *Audio, Speech, and Language Processing, IEEE Transactions on,* vol. 20(1), pp. 30-42, 2012.

[12] Krizhevsky A, Sutskever I, Hinton G E, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*. 2012, pp. 1097-1105.

[13] Socher R. Recursive, "Deep Learning for Natural Language Processing and Computer Vision," *Stanford University*, 2014.

[14] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, 2014, pp. 3104-3112.

[15] Bengio Y, Ducharme R, Vincent P, et al. "A neural probabilistic language model". *The Journal of Machine Learning Research*, 2003, 3: 1137-1155.

[16] 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.

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

[18] 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.

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

[20] 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*, pp. 1045-1048.

[21] 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.

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

[23] 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.

[24] 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.

[25] 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.

[26] 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.

[27] R. Mihalcea and P. Tarau, "TextRank: Bringing order into texts," 2004.