Using Word Embeddings to Retrieve Semantically Similar Questions in Community Question Answering

Nouha Othman, Rim Faiz, Kamel Smaïli

Using Word Embeddings to Retrieve Semantically Similar Questions in Community Question Answering

NOUHA OTHMAN[1], RIM FAIZ[2], AND KAMEL SMAÏLI[3]

[1] *LARODEC, University of Tunis - Tunisia*
[2] *LARODEC, University of Carthage - Tunisia*
[3] *LORIA, University of Lorrraine - France*
[1] *othmannouha@gmail.com*
[2] *rim.faiz@ihec.rnu.tn*
[3] *smaili@loria.fr*

Compiled February 23, 2018

This paper focuses on question retrieval which is a crucial and tricky task in Community Question Answering (cQA). Question retrieval aims at finding historical questions that are semantically equivalent to the queried ones, assuming that the answers to the similar questions should also answer the new ones. The major challenges are the lexical gap problem as well as the verboseness in natural language. Most existing methods measure the similarity between questions based on the bag-of-words (BOWs) representation capturing no semantics between words. In this paper, we rely on word embeddings and TF-IDF for a meaningful vector representation of the questions. The similarity between questions is measured using cosine similarity based on their vector-based word representations. Experiments carried out on a real world data set from Yahoo! Answers show that our method is competetive.

1. INTRODUCTION

Community-based Question Answering (cQA), which provides platforms for people with different backgrounds to share knowledge, has become an increasingly popular mean of information seeking on the web. In cQA, users can interact and respond to other users' questions or post their own questions for other participants to answer [1]. Over the last years, with the boom of Web 2.0, cQA emerges as an exciting form of online service for producing user-generated content, such as Yahoo! Answers[1], Stackoverflow[2], MathOverflow[3], LinuxQuestions[4] and so forth. Such community services have built up huge archives of question-answer pairs that are continuously increasing accumulating duplicated questions. Consequently, users cannot easily find the good answers among hundreds of possible answers and then post new questions that already exist in the archives. In order to reduce the time lag required to get a new answer, cQA should automatically search the community archive to check if equivalent questions

[1] http://answers.yahoo.com/

[2] http://stackoverflow.com/
[3] http://www.mathoverflow.net
[4] http://www.linuxquestions.org/

have previously been posted. If a similar question is detected, its associated answer can be directly returned as a relevant answer to the new query.

Recently, numerous interesting studies have been done along this line [2–7] with the aim of answering new questions with past answers. As a matter of fact, question retrieval is a non trivial task facing several challenges as questions in cQA vary significantly in terms of vocabulary, length, style, and content quality. The greatest challenge is the lexical gap between the queried questions and the existing ones in the archives [2], which constitutes a real obstacle to traditional Information Retrieval (IR) models since users can formulate the same question employing different wording. For instance, the questions: *What are the characteristics of your work?* and *How can you describe your job ?*, have the same meaning but they are lexically different. The word mismatching is a critical issue in cQA since questions are relatively short and similar ones usually have sparse representations with little word overlap. From this, it is clear that effective retrieval models for question retrieval are strongly needed to take full advantage of the sizeable community archives. In order to bridge the lexical gap problem in cQA, most state-of-the-art studies attempt to improve the similarity measure between questions while it is hard to set a compelling similarity function for sparse and discrete representations of words. More importantly, most existing approaches neither take into account the contextual information nor capture enough semantic relations between words. Recent efforts in learning distributed semantic representations, also called word embedding, have been shown to be a great asset for a large variety of Natural Language Processing (NLP) and IR tasks, such as word analogy [8], recommender systems [9] and question retrieval [10]. Word embeddings is an emerging technique which aims at mapping words from a vocabulary into real vectors in a low-dimensional (compared to the vocabulary size) space. In this space, close vectors are supposed to indicate high semantic and syntactic similarity between the corresponding words. Although word embeddings have shown significant performance in many challenging tasks, there is less known about the use of word embeddings to improve the question retrieval task.

Motivated by the recent success of these emerging methods, in this paper, we propose a word embedding-based method for question retrieval in cQA. Instead of representing questions as a bag of words (BoW), we suggest representing them as Bag of-Embedded-Words (BoEW) in a continuous space using word2vec, the most popular word embedding model. The representation of words using semantic word embeddings should grasp most of the semantic information in the questions. The generated word embeddings of a question are then weighted through the use of TF-IDF (term frequency - inverse document frequency) information and averaged to get an overall representation of the question. Interestingly, the use of word embedding to represent words along with TF-IDF weighting has shown promise in finding an effective vector representation for a short text fragment [11]. Questions are therefore ranked using cosine similarity based on the vector based word representation for each question. A previous posted question is considered to be semantically similar to a queried question if their corresponding vector representations lie close to each other according to the cosine similarity measure. The previous question with the highest cosine similarity score will be returned as the most similar question to the new posted one. We test the proposed method on a large-scale real data from Yahoo! Answers. Experimental results show that our method is promising and can outperform certain state-of-the-art methods for question retrieval in cQA.

The rest of this paper is organized as follows: In Section (2), we give an overview of the main related work on question retrieval in cQA. Then, we describe in Section (3) our proposed word embedding based-method for question retrieval. Section (4) presents our experimental evaluation and Section (5) concludes the paper and outlines some perspectives.

2. RELATED WORK

Recently, along with the flourishing of community question answering (CQA) services, there has been growing interest in question retrieval in cQA. Significant research efforts have been conducted in order to detect semantically similar questions that can be adequately answered by the same answer.

Several works were based on the vector space model referred to as VSM to calculate the cosine similarity between a query and archived questions [3, 12]. However, the major limitation of VSM is that it favors short questions, while cQA services can handle a wide range of questions not limited to concise or factoid questions. In order to overcome the shortcoming of VSM, BM25 have been employed for question retrieval to take into consideration the question length [3]. Okapi BM25 is the most widely applied model among a family of Okapi retrieval models proposed by Robertson et al. in [13] and has proven significant performance in several IR tasks. Besides, Language Models (LM)s [14] have been also used to explicitly model queries as sequences of query terms instead of sets of terms. LMs estimate the relative likelihood for each possible successor term taking into consideration relative positions of terms. Nonetheless, such models might not be effective when there are few common words between the user's query and the archived questions.

To overcome the vocabulary mismatch problem faced by LMs, the translation model was used to learn correlation between words based on parallel corpora and it has obtained significant performance for question retrieval. The basic intuition behind translation models is to consider question-answer pairs as parallel texts, then relationship of words can be constructed by learning word-to-word translation probabilities such as in [2, 4]. Within the same context, [15] presented a parallel dataset for training statistical word translation models, composed of the definitions and glosses provided for the same term by different lexical semantic resources. In [16], the authors tried to improve the word-based translation model by adding some contextual information when building the translation of phrases as a whole, instead of translating separate words. In [5], the word-based translation model was extended by incorporating semantic information (entities) and explored strategies to learn the translation probabilities between words and concepts using the cQA archives and an entity catalog. Although, the aforementioned basic models have yielded good results, questions and answers are not really parallel, rather they are different from the information they contain [6].

Advanced approaches based on semantic similarity were required to bridge the lexical gap problem in question retrieval toward a deep understanding of short text to detect the equivalent questions. For instance, there were few attempts that have exploited the available category information for question retrieval like in [3, 14, 17]. Despite the fact that these attempts have proven to significantly improve the performance of the language model for question retrieval, the use of category information was restricted to the language model. Wang et al [18] used a parser to build syntactic trees of questions, and rank them based on the similarity between their syntactic trees and that of the query question. Nevertheless, such an approach is very complex since it requires a lot of training data. As observed by [18], existing parsers are still not well-trained to parse informally written questions.

Furthermore, many attempts have been made in the past to model the semantic relationship between the searched questions and the candidate ones with deep question analysis such as [12] who proposed to identify the question topic and focus for question retrieval. Within this context, some studies relied on a learning-to-ranking strategy like [19] who presented an approach to rank the retrieved questions with multiple features, while [20] rank the candidate answers with a single word information instead of the combination of various features. Latent Semantic Indexing (LSI) [21] was also employed to address the given task like in [22]. While being effective to address the synonymy and polysemy by mapping words about the same concept next to each other, the efficiency of LSI highly depends on the data structure and both its training and inference are computationally expensive on large vocabularies.

Otherwise, other works focused on the representation learning for questions, relying on an emerging model for learning distributed representations of words in a low-dimensional vector space namely Word Embedding. This latter has recently been subject of a wide interest and has shown promise in numerous NLP tasks [23, 24], in particular for question retrieval [10]. The main virtue of this unsupervised learning model is that it doesn't need expensive annotation; it only requires a huge amount of raw textual data in its training phase. As we believe that the representation of words is vital for the question retrieval task and inspired by the success of the latter model, we rely on word embeddings to address the question retrieval task in cQA.

3. DESCRIPTION OF WECOSIM

The intuition behind the method we propose for question retrieval, called *WECOSim*, is to transform words in each question in the community collection into continuous vectors. Unlike traditional methods which represent each question as Bag Of Words (BOWs), we propose to represent a question as a Bag-of-Embedded-Words (BoEW). The continuous word representations are learned in advance using the continuous bag-of-words (CBOW) model [25]. Each question can, therefore, be defined as a set of words embedded in a continuous space. The word embeddings of a question are weighted through the use of TF-IDF information and averaged to get an overall representation of the question. Besides, the cosine similarity is used to calculate the similarity between the average of the word vectors corresponding to the queried question and that of each existing question in the archive. The historical questions are then ranked according to their cosine similarity scores in order to return the top ranking question having the maximum score, as the most relevant one to the new queried question. As depicted in Figure 1, the proposed method for question retrieval in cQA consists of four steps namely, question preprocessing, word embedding learning, embedding vector weighting and question ranking.

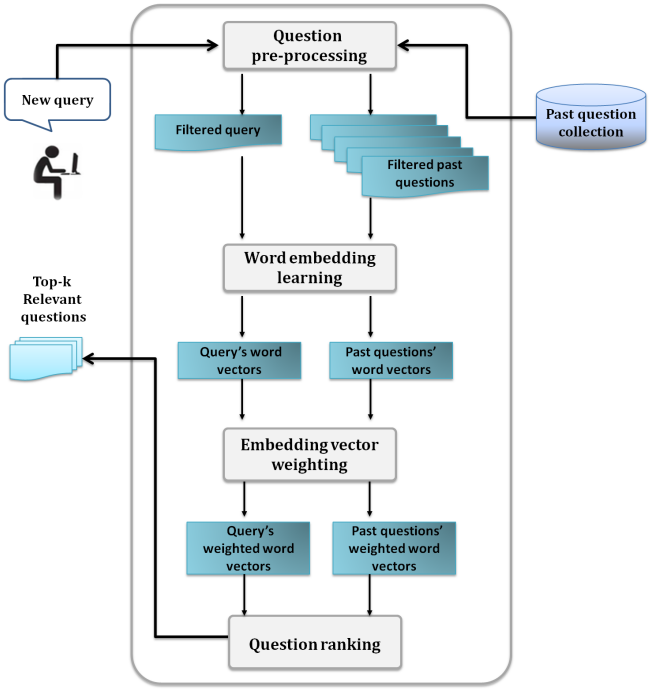

Fig. 1. Overview of the proposed method

A. Question Preprocessing

The question preprocessing module intends to process the natural language questions and extract the useful terms in order to generate formal queries. These latter are obtained by applying text cleaning, tokenization, stopwords removal and stemming. Thus, at the end of the question preprocessing module, we obtain a set of filtered queries, each of which is formally defined as follows: $q = \{t_1, t_2, ..., t_Q\}$ where t represents a separate term of the query q and Q denotes the number of query terms.

B. Word Embedding Learning

Word embedding techniques, also known as distributed semantic representations play a significant role in building continuous word vectors based on their contexts in a large corpus. They learn a low-dimensional vector for each vocabulary term in which the similarity between the word vectors can show the syntactic and semantic similarities between the corresponding words. Basically, there exist two main types of word embeddings namely Continuous Bag-of-Words model (CBoW) and Skip-gram model. The former one consists in predicting a current word given its

context, while the second does the inverse predicting the contextual words given a target word in a sliding window. It is worthwhile to note that, in this work, we consider the CBOW model [25] to learn word embeddings, since it is more efficient and performs better with sizeable data than Skip-gram. As shown in Figure 2, the CBOW model predicts the center word given the representation of its surrounding words using continuous distributed bag-of-words representation of the context, hence the name CBOW. The context vector is got by averaging the

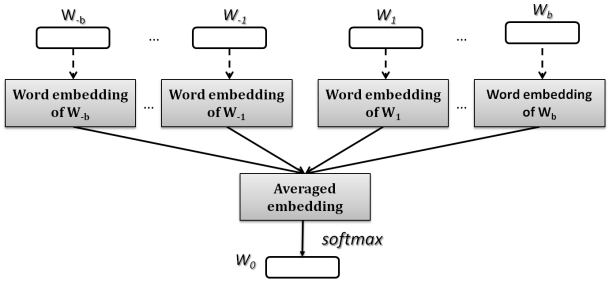

Fig. 2. Overview of the Continuous Bag-of-Words model.

embeddings of each contextual word while the prediction of the center word w_0 is obtained by applying a softmax over the vocabulary V. Formally, let d be the word embedding dimension, the output matrix $O \in \Re^{|V| \times d}$ maps the context vector c into a $|V|$-dimensional vector representing the center word, and maximizes the following probability:

$$p(v_0 \mid w_{[-b,b]-\{0\}}) = \frac{\exp v_0^T O_c}{\sum_{v \in V} \exp v^T O_c} \quad (1)$$

where b is a hyperparameter defining the window of context words, O_c represents the projection of the context vector c into the vocabulary V and v is a one-hot representation. The strength of CBOW is that it does not rise substantially when we increase the window b.

C. Embedding Vector Weighting

Once the questions are presented as Bag of-Embedded-Words (BoEW), the generated vectors are weighted using TF-IDF, which is one of the most widely used term weighting schemes in information retrieval systems owing to its simplicity and effectiveness. In other words, each embedding word is multiplied by the TF-IDF of the word it represents. TF-IDF is a statistic weighting function to calculate the importance of a word based on its relative frequency in a specific document and the inverse proportion of documents containing the word over the entire document collection. As we work on questions, we adapt the basic function to our context by simply replacing documents with questions. Given a question collection C , a word w and a question q, TF-IDF is defined as follows:

$$tfidf(w,q,C) = tf(w,q) * idf(w,Q) = f_{w,q} * \log(\frac{|C|}{f_{w,C}}) \quad (2)$$

where $f_{w,q}$ is the number of times w appears in a question q, $|C|$ is the size of the question collection and $f_{w,C}$ is the total number of questions that contain the word w. We use TF-IDF to estimate how important is a word not only in a particular question, but rather in the whole collection of questions. Actually, some common words may occur several times in questions but they are not relevant as key-concepts to be indexed or searched. Intuitively, words that are common in a single or small set of questions will be assigned higher scores while words which appear frequently in questions tend to have low scores.

D. Question Ranking

The weighted embedding vectors of the query words are employed to calculate the average vector V_q of the queried question as follows:

$$V_q = \frac{\sum_{i=1}^{|V|} (v_{w_i} \times tfidf(w_i,q,C))}{\sum_{i=1}^{|V|} tfidf(w_i,q,C)} \quad (3)$$

where v_{w_i} is the embedding vector of the word w_i generated by word2vec and $|V|$ is the number of word vectors in a given question q . Similarly, for each historical question, we compute its average vector V_d. The similarity between a queried question and a historical one in the vector space is calculated as the cosine similarity between V_q and V_d. Questions are ranked using cosine similarity scores based on their weighted vectors in order to return the top ranking questions having the maximum score, as the most relevant ones to the new queried question.

4. EXPERIMENTS

A. Dataset

In our experiments, we used the dataset released by [26] for evaluation. In order to construct the dataset, the authors crawled questions from all categories in Yahoo! Answers, the most popular cQA platform, and then randomly splitted the questions into two sets while maintaining their distributions in all categories. The first set contains 1,123,034 questions as a question repository for question search, while the second is used as the test set and contains 252 queries and 1624 manually labeled relevant questions. The number of relevant questions related to each original query varies from 2 to 30. The questions are of different lengths varying from two to 15 words, in different structures and belonging to various categories e.g. Computers and Internet, Yahoo! Products, Entertainment and Music, Education and Reference, Business and Finance, Pets, Health, Sports, Travel, Diet and Fitness. Table 1 shows an example of a query and its corresponding related questions from the test set. Annotators

Table 1. *Example of questions from the test set.*

Query:	How can I get skinnier without getting in a diet?
Category:	Diet and Fitness
Topic:	Weight loss
Related questions	- How do I get fit without changing my diet? - How can i get slim but neither diet nor exercise? - How do you get skinny fast without diet pills? - I need a solution for getting fit (loosing weight) and I must say I cant take tough diets ?

were asked to label each query with “relevant" if a candidate question is considered semantically similar to the query or “irrelevant" otherwise. In case of conflict, happen, a third annotator will make judgement for the final result. Note that the questions in the test data do not overlap with those in the retrieval data. To train the word embeddings, we resorted to another large-scale data set from cQA sites, namely the Yahoo! Webscope dataset[5], including 1,256,173 questions with 2,512,345 distinct words. Some preprocessing was performed before the experiments; all questions were lower cased, tokenized, stemmed by Porter Stemmer[6] and all stop words were removed.

B. Learning of Word Embedding

We trained the word embeddings on the whole Yahoo! Webscope dataset using word2vec in order to represent the words of the training data as continuous vectors which capture the contexts of the words. The training parameters of word2vec were set after several tests: the dimensionality of the feature vectors was fixed at 300 (size=300), the size of the context window was set to 10 (window=10) and the number of negative samples was set to 25 (negative=25).

C. Evaluation Metrics

In order to evaluate the performance of our method, we used Mean Average Precision (MAP) and Precision@n (P@n) as they are extensively used for evaluating the performance of question retrieval for cQA. Particularly, MAP is the most commonly used metric in the literature assuming that the user is interested in finding many relevant questions for each query. MAP rewards methods that not only return relevant questions early, but also get good ranking of the results. Given a set of queried questions Q, MAP represents the mean of the average precision for each queried question q and it is set as follows:

$$\text{MAP} = \frac{\sum_{q \in Q} \text{AvgP}(q)}{\mid Q \mid} \tag{4}$$

where $AvgP(q)$ is the mean of the precision scores after each relevant question q is retrieved, and it is defined as:

$$\text{AvgP} = \frac{\sum_{r} P@r}{R} \tag{5}$$

where r is the rank of each relevant question, R is the total number of relevant questions, and $P@r$ is the precision of the top-r retrieved questions.

Precision@n returns the proportion of the top-n retrieved questions that are relevant. Given a set of queried questions Q, P@n is the proportion of the top n retrieved questions that are relevant to the queries, and it is defined as follows:

$$\text{P@n} = \frac{1}{\mid Q \mid} \sum_{q \in Q} \frac{\text{Nr}}{N} \tag{6}$$

where Nr is the number of relevant questions among the top N ranked list returned for a query q. In our experiments, we calculated P@10 and P@5.

[5]The Yahoo! Webscope dataset Yahoo answers comprehensive questions and answers version 1.0.2, available at “http://research.yahoo.com/Academic_Relations"

[6]http://tartarus.org/martin/PorterStemmer/

D. Main Results

We compare the performance of WECOSim with the following competitive state-of-the-art question retrieval models tested by Zhang et al. in [26] on the same dataset:

- **TLM** [2]: A translation based language model which combines the translation model estimated using the question and the language model estimated using the answer part. It integrates word-to-word translation probabilities learned by exploiting various sources of information.
- **ETLM** [5]: An entity based translation language model, which is an extension of TLM by replacing the word translation with entity translation in order to incorporate semantic information within the entities.
- **PBTM** [16]: A phrase based translation model which employs machine translation probabilities and assumes that question retrieval should be performed at the phrase level. TLM learns the probability of translating a sequence of words in a historical question into another sequence of words in a queried question.
- **WKM** [27]: A world knowledge based model which used Wikipedia as an external resource to add the estimation of the term weights to the ranking function. A concept thesaurus was built based on the semantic relations extracted from the world knowledge of Wikipedia.
- **M-NET** [10]: A continuous word embedding based model, which integrates the category information of the questions to get the updated word embedding, assuming that the representations of words that belong to the same category should be close to each other.
- **ParaKCM** [26]: A key concept paraphrasing based approach which explores the translations of pivot languages and expands queries with the paraphrases. It assumes that paraphrases contributes additional semantic connection between the key concepts in the queried question and those of the historical questions.

From Table 2, we can see that PBTM outperforms TLM which demonstrates that capturing contextual information in modeling the translation of phrases as a whole or consecutive sequence of words is more effective than translating single words in isolation. This is because, by and large, there is a dependency between adjacent words in a phrase. The fact that ETLM (an extension of TLM) performs as good as PBTM proves that replacing the word translation by entity translation for ranking

Table 2. *Comparison of the question retrieval performance of different models.*

	TLM	ETLM	PBTM	WKM	M-NET	ParaKCM	WECOSim-tfidf	WECOSim
P@5	0.3238	0.3314	0.3318	0.3413	0.3686	0.3722	**0.3432**	**0.4339**
P@10	0.2548	0.2603	0.2603	0.2715	0.2848	0.2889	**0.2738**	**0.3646**
MAP	0.3957	0.4073	0.4095	0.4116	0.4507	0.4578	**0.4125**	**0.5038**

improves the performance of the translation language model. Although, ETLM and WKM are both based on external knowledge resource e.g. Wikipedia, WKM uses wider information from the knowledge source. Specifically, WKM builds a Wikipedia thesaurus, which derives the concept relationships (e.g. synonymy, hypernymy, polysemy and associative relations) based on the structural knowledge in Wikipedia. The different relations in the thesaurus are treated according to their importance to expand the query and then enhance the traditional similarity measure for question retrieval. Nevertheless, the performance of WKM and ETLM are limited by the low coverage of the concepts of Wikipedia on the various users' questions. M-NET, also based on continuous word embeddings performs well owing to the use of metadata of category information to encode the properties of words, from which similar words can be grouped according to their categories. The best compared system is ParaKCM, a key concept paraphrasing based approach which explores the translations of pivot languages and expands queries with the generated paraphrases for question retrieval.

The results show that our method WECOSim significantly outperforms all the aforementioned methods on all criteria by returning a good number of relevant questions among the retrieved ones early. A possible reason behind this is that context-vector representations learned by word2vec can effectively address the word lexical gap problem by capturing semantic relations between words, while the other methods do not capture enough information about semantic equivalence. We can say that questions represented by bag-of-embedded words can be captured more accurately than traditional bag-of-words models which cannot capture neither semantics nor positions in text. This good performance indicates that the use of word embeddings along with TF-IDF weighting and cosine similarity is effective in the question retrieval task. However, we find that sometimes, our method fails to retrieve similar questions: Out of 252 test questions, only 12 questions get P@10 values equal to zero. Most of these questions contain misspelled query terms. For instance, questions containing *sofwar* by mistake cannot be retrieved for a query containing the term *software*. Such cases show that our approach fails to address some lexical disagreement problems. Furthermore, there are few cases where WECOSim fails to detect semantic equivalence. Some of these cases include questions having one single similar question and most words of this latter do not appear in a similar context with those of the queried question, such as: *Which is better to aim my putter towards, the pole or the hole?* and *How do I aim for the target in golf?*. Obviously, further experiments with the dimensions of the embeddings are needed to improve the results.

On the other hand, we tested our method with and without TF-IDF weighting (In Table 2, WECOSim and WECOSim-tfidf respectively) to examine its effect on question retrieval results. Through our experiments, we found that the use of TF-IDF allows to slightly increase the P@5, P@10 and the MAP values. The reason behind this is that TF-IDF can detect questions that make frequent use of specific words and determine if they are relevant in the question. We can say that the discriminatory power of TF-IDF enables the retrieval engine to find relevant questions that could likely be similar to the new query. However, there are some cases when a word can be relatively common over the whole collection but still holds some importance throughout the question like the words *date* and *system*. Such common words get a low TF-IDF score, and thus are pretty much ignored in the search. Furthermore, TF-IDF doesn't take into account synonymy relations between terms. For example, if a user posted a question including the word *dwelling*, TF-IDF would not consider questions that might be similar to the query but instead use the word *bungalow*. TF-IDF can not resolve lexical ambiguity which is frequent in our community collection of informal and heterogeneous questions where the same concept may be expressed in different ways. It is also worth mentioning that the computation complexity of TF-IDF is $O(nm)$, where n is the total number of words and m is the total number of questions in the corpus. For large collections like yours, this could present an escalating problem.

5. CONCLUSION

Our work falls within the human-generated content spirit of the cQA and reuse of past questions and answers. We focus on the question retrieval task, presuming that the corresponding answers to a similar past question should meet the new question needs. In this paper, we propose a word embedding based method to address the lexical gap problem in question retrieval from cQA archives. Specifically, we suggest incorporating an embedding of words in a continuous space for question representations. The word embeddings are learned in advance using the CBOW model and weighted based on the frequency of the words. To find semantically similar questions to a new query, historical questions are ranked using cosine similarity based on their vector-based word representations in a continuous space. Experiments conducted on large-scale cQA data show the effectiveness of the use of semantic word embeddings along with TF-IDF to represent question words. Our method can significantly outperform existing ones in finding similar questions even if they share few common words. We have shown evidence that the TF-IDF weighting, though simple, can improve the search efficiency and the quality of the retrieval results. Nevertheless, there is a limit to represent word as one vector without considering lexical ambiguity. For future research, we will consider enhancing the simple TF-IDF scheme and incorporating various types of metadata information into the learning process in order to enrich word representations. It would also be of interest to investigate the use of different similarity measures for computing the semantic similarity between questions.

REFERENCES

1. Y. Liu, J. Bian, and E. Agichtein, "Predicting information seeker satisfaction in community question answering," in "Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval," (ACM, 2008), pp. 483–490.
2. 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," (ACM, 2008), pp. 475–482.

3. 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," (ACM, 2010), pp. 201–210.
4. L. Cai, G. Zhou, K. Liu, and J. Zhao, "Learning the latent topics for question retrieval in community qa." in "Proceedings of 5th International Joint Conference on Natural Language Processing," (2011), pp. 273–281.
5. A. Singh, "Entity based q&a retrieval," in "Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning," (ACL, 2012), pp. 1266–1277.
6. K. Zhang, W. Wu, H. Wu, Z. Li, and M. Zhou, "Question retrieval with high quality answers in community question answering," in "Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management," (ACM, 2014), pp. 371–380.
7. P. Nakov, D. Hoogeveen, L. Màrquez, A. Moschitti, H. Mubarak, T. Baldwin, and K. Verspoor, "Semeval-2017 task 3: Community question answering," in "Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)," (2017), pp. 27–48.
8. T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in "Advances in neural information processing systems," (2013), pp. 3111–3119.
9. C. Musto, G. Semeraro, M. de Gemmis, and P. Lops, "Learning word embeddings from wikipedia for content-based recommender systems," in "European Conference on Information Retrieval," (Springer, 2016), pp. 729–734.
10. G. Zhou, T. He, J. Zhao, and P. Hu, "Learning continuous word embedding with metadata for question retrieval in community question answering," in "Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing," (2015), pp. 250–259.
11. C. De Boom, S. Van Canneyt, T. Demeester, and B. Dhoedt, "Representation learning for very short texts using weighted word embedding aggregation," Pattern Recognit. Lett. **80**, 150–156 (2016).
12. H. Duan, Y. Cao, C.-Y. Lin, and Y. Yu, "Searching questions by identifying question topic and question focus." in "ACL," , vol. 8 (2008), vol. 8, pp. 156–164.
13. S. E. Robertson, S. Walker, S. Jones, M. M. Hancock-Beaulieu, M. Gatford *et al.*, "Okapi at trec-3," Nist Special Publ. Sp **109**, 109 (1995).
14. 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," (ACM, 2009), pp. 265–274.
15. D. Bernhard and I. Gurevych, "Combining lexical semantic resources with question & answer archives for translation-based answer finding," in "Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP," (ACL, 2009), pp. 728–736.
16. G. Zhou, L. Cai, J. Zhao, and K. Liu, "Phrase-based translation model for question retrieval in community question answer archives," in "Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1," (ACL, 2011), pp. 653–662.
17. G. Zhou, Y. Chen, D. Zeng, and J. Zhao, "Towards faster and better retrieval models for question search," in "Proceedings of the 22nd ACM international conference on Conference on information & knowledge management," (ACM, 2013), pp. 2139–2148.
18. K. Wang, Z. Ming, and T.-S. Chua, "A syntactic tree matching approach to finding similar questions in community-based qa services," in "Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval," (ACM, 2009), pp. 187–194.
19. M. Surdeanu, M. Ciaramita, and H. Zaragoza, "Learning to rank answers on large online qa collections." in "ACL," , vol. 8 (2008), vol. 8, pp. 719–727.
20. B. Wang, X. Wang, C. Sun, B. Liu, and L. Sun, "Modeling semantic relevance for question-answer pairs in web social communities," in "Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics," (ACL, 2010), pp. 1230–1238.
21. S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by latent semantic analysis," J. Am. society for information science **41**, 391 (1990).
22. X. Qiu, L. Tian, and X. Huang, "Latent semantic tensor indexing for community-based question answering." in "ACL (2)," (2013), pp. 434–439.
23. J. Turian, L. Ratinov, and Y. Bengio, "Word representations: a simple and general method for semi-supervised learning," in "Proceedings of the 48th annual meeting of the association for computational linguistics," (ACL, 2010), pp. 384–394.
24. R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, "Natural language processing (almost) from scratch," J. Mach. Learn. Res. **12**, 2493–2537 (2011).
25. T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," arXiv preprint arXiv:1301.3781 (2013).
26. W.-N. Zhang, Z.-Y. Ming, Y. Zhang, T. Liu, and T.-S. Chua, "Capturing the semantics of key phrases using multiple languages for question retrieval," IEEE Transactions on Knowl. Data Eng. **28**, 888–900 (2016).
27. G. Zhou, Y. Liu, F. Liu, D. Zeng, and J. Zhao, "Improving question retrieval in community question answering using world knowledge." in "IJCAI," , vol. 13 (2013), vol. 13, pp. 2239–2245.

Nouha Othman received the B.Sc degree in Computer Science from Institut Supérieur de Gestion (ISG) of the University of Tunis, Tunisia, and two master's degrees in Computer Science from ISG and Polytech Nantes of the University of Nantes, France. Currently, she is a Ph.D. candidate in Computer Science at ISG Tunis, University of Tunis, belonging to LARODEC laboratory since 2015. Her research interests include information retrieval, natural language processing and machine learning.

Rim Faiz obtained her Ph.D. in Computer Science from the University of Paris-Dauphine, LAMSADE Lab., in France. She is currently a Professor in Computer Science at the Institute of High Business Study (IHEC), LARODEC Lab., University of Carthage, in Tunisia. Her research interests include Information Retrieval, Big Data, Text Mining, Machine Learning, Natural Language Processing, and Semantic Web. She has published several papers and has served as PC member and reviewer for several international conferences and journals. Dr. Faiz is also responsible of the Master "E-Commerce and Technological Innovation" and the Master "Business Intelligence" at IHEC, University of Carthage.

Kamel Smaïli is Professor at the university of Lorraine since 2002, he obtained a PhD from the university of Nancy 1 in 1991 on automatic speech recognition. He defended his HDR in 2001 (Statistical language modelling: from speech recognition to machine translation). His research interest since more than 20 years concerns statistical language modelling for automatic speech recognition. Since 2000 he oriented his research towards speech to speech translation. He participated to several European and national projects concerning automatic speech recognition: COCOS, MULTIWORKS, COST, MIAMM, IVOMOB (RNRT project) and CMCU. He advised 14 PhD and HDR students and he participated to more than 35 PhD committees in France, Germany, Spain and Algeria. He took part to several program committees: Interspeech, Eurospeech, ICSLP, ICASSP, TALN, ICWMI, SIIE, TAIMA, Machine Translation, Computer speech and language, Speech communication, Journal of Natural Language Engineering,... He has been invited several times to give talks as invited speakers in Japan, France, Tunisia, Algeria and Morocco. He published 90 papers in international conferences and journals and 20 papers in francophone conferences. Furthermore, Mr Smaili was the head of MIAGE (Master and Bachelor) department for 7 years and the head of UFR (equivalent to a faculty) Mathematics and Informatics for 5 years where he managed more than 30 permanent people and 120 temporary positions, 500 students.