Query Expansion for Sentence Retrieval Using Pseudo Relevance Feedback and Word Embedding

Piyush Arora(✉), Jennifer Foster, and Gareth J.F. Jones

School of Computing, ADAPT Centre, Dublin City University, Dublin, Ireland
{parora,jfoster,gjones}@computing.dcu.ie

Abstract. This study investigates the use of query expansion (QE) methods in sentence retrieval for non-factoid queries to address the query-document term mismatch problem. Two alternative QE approaches: i) pseudo relevance feedback (PRF), using Robertson term selection, and ii) word embeddings (WE) of query words, are explored. Experiments are carried out on the WebAP data set developed using the TREC GOV2 collection. Experimental results using P@10, NDCG@10 and MRR show that QE using PRF achieves a statistically significant improvement over baseline retrieval models, but that while WE also improves over the baseline, this is not statistically significant. A method combining PRF and WE expansion performs consistently better than using only the PRF method.

Keywords: Query expansion · Pseudo relevance feedback · Word embeddings · Sentence retrieval

1 Introduction

Sentence retrieval is a challenging information retrieval (IR) task which is useful in question answering and summarization systems. Retrieval of sentences relevant to an answer is a difficult task due to short length of the target items which are more likely to suffer from vocabulary mismatch with respect to the query. We focus on the new task of answer passage retrieval for non-factoid queries [3,8]. In this paper we investigate two unsupervised query expansion (QE) methods which seek to address the query-document term mismatch issue. First, we use Robertson's standard Okapi QE relevance feedback method [7]. Second, we propose a word embedding (WE) method [6] to expand the query using words similar to the query based on vector similarity. Finally, we explore the combination of these alternative sources of evidence for QE. We investigate the following research questions:

1. How do unsupervised approaches using traditional retrieval techniques with QE perform for the task of sentence retrieval from a target corpus?
2. Can we leverage word embeddings to improve retrieval effectiveness in a sentence retrieval task?

G.J.F. Jones et al. (Eds.): CLEF 2017, LNCS 10456, pp. 97–103, 2017.
DOI: 10.1007/978-3-319-65813-1_8

2 Related Work

Finding relevant sentences from a given document for a given query is a common task in applications which involve generating summaries and abstracts from individual documents [5]. Our work differs from this since we are focused on retrieval of relevant sentences from a *complete target document collection* for a given query to find answer sentences for non-factoid queries, similar to the work reported in [8].

A related study [1] explored different retrieval models and QE techniques to find novel information from a document collection. Our work is focused on relevancy rather than novelty but we also explore QE models. Supervised approaches such as Learning-to-Rank (L2R) have been used for answer sentence retrieval [3,5,8]. Our work investigates whether we can achieve similar performance using QE techniques as compared to L2R techniques.

Word embedding is a method for forming low-dimensional vector representations of words based on co-occurrence in a reference corpus, thereby providing a means of semantic comparison between words. WE is of increasing interest among the IR community, and there are a number of examples of recent work exploring the use of WE for document-based QE [2,4,9]. [4] investigated different methods for document retrieval, [2] compared different types of embeddings, and [9] explored effective ways of combining embeddings and co-occurrence based statistics for document retrieval. Our work is similar but we focus on the task of sentence retrieval.

3 Experimental Methodology

As our baseline models, we perform sentence retrieval using: (i) a language modeling (LM) method that uses Jelinek-Mercer smoothing [10], and (ii) the BM25 model [11]. We then perform QE using the following methods:

1. **Pseudo Relevance Feedback (PRF):** We use a standard PRF method in which an initial retrieval run is carried out using the selected baseline retrieval system. A number of the top ranked documents are assumed to be relevant, and potential QE terms from within these documents are ranked using the Robertson Offer Weight [7]. A fixed number of these terms are then added to the original query.
2. **Semantic expansion, using word embeddings (WE):** We explore two novel approaches to using WE for QE. The embedding of each word is computed using the *Word2Vec* [6] method, which learns vector representation using a feed-forward neural network by predicting a word given its context (the cbow model).
   For a given query Q consisting of n terms $q_1, \ldots, q_n$, for each term q_i we generate a pool of potential candidate expansion terms c_i, such that $c_i =$ the top z most similar terms $t^1, \ldots, t^z$ for each term q_i. Thus the complete

set of potential expansion candidates for query Q is $C_i^j = c_1, \ldots, c_n$, where i varies from 1 to n and j varies from 1 to z. The most similar terms for each term q_i are obtained by calculating the cosine similarity between its vector representation and the terms in the document corpus. We then select the top k expansion terms for Q from the potential candidates, using one of two methods:

(a) ***QueryWord approach***: We sort all the terms in C_i^j, based on the cosine similarity score between each term t_i^j and the corresponding query term q_i. We select the top k terms from this sorted list as the expansion terms. The use of the cosine similarity in this way biases this list towards expansion terms which are closely related to terms in Q, these terms may be synonyms or words with a close semantic relationship to terms appearing in Q.

(b) ***Centroid approach***: We form a centroid vector (CV) of the query, by summing the vectors of all the query terms q_i in Q. We then sort all the terms in C_i^j, based on the cosine similarity score between each term t_i^j and CV. We select the top k ranked terms as the expansion terms which are most similar to CV. The main goal is to retrieve terms which are related to all the terms in Q as a unit.

3. **Combining WE and PRF**: PRF seeks expansion terms in the top ranked documents following an actual search which is hoped will have retrieved relevant items, while WE seeks to identify words that are similar in meaning to the query words independent of their use in the query. PRF and WE thus provide different sources of information which may be exploited in combination in the QE process. To explore the potential for this combination, we examine an approach which linearly combines terms expanded using PRF and WE calculated as follows,

$$COW = (w_t) * WE_{EQT} + (1 - w_t) * PRF_{EQT} \tag{1}$$

where EQT stands for expanded query terms.

4 Experimental Setup

We use the WebAP dataset [8], which was developed using the TREC Gov2 collection, and has 82 queries with a total of 6,399 documents consisting of 991,233 sentences which are marked at the sentence level on a 5 level scale of topical relevance: *4*: perfect, *3*: excellent, *2*: good, *1*: fair, *0*: none. Following [8], we used precision (P@10), normalized discounted cumulative gain (NDCG@10) and mean reciprocal rank (MRR) to compare the performance of our methods.

The Lucene toolkit[1] was used to perform sentence retrieval. We performed stemming and stopword removal using the Lucene EnglishAnalyzer. Sentence retrieval used lucene's implementation of LM or BM25. We implemented our own

[1] https://lucene.apache.org/core/4_4_0/core/overview-summary.html.

version of Robertson QE. The Gensim tookit was used to learn and incorporate *word embeddings* for use in QE as described above. We used two different types of embeddings:

1. *Global embeddings*: Embeddings trained on Google news, consisting of about 3 million 300 dimension English word vectors which are released for research. [2]
2. *Local embeddings*: Embeddings learnt using subcollection of the WebAP dataset consisting of 6,399 documents, with parameter settings as follows– method: continuous bag-of-words (cbow), embedding size=200, window for training = 10, iterations set for learning = 20, the rest of the parameters were kept as default. [3]

5 Results and Analysis

Baselines: We explored optimization of parameters by varying λ in the LM retrieval model in the range of [0.05, 1.0], and performed grid search in the range of [0.1, 1.5] with an increment of 0.1 for the b and k parameters for the BM25 model. Our best results were obtained for $\lambda = 0.45$ for LM and $b = 0.4$ and $k = 0.4$ for BM25, as shown in Table 2, these values were then fixed for subsequent experiments. Table 1 also presents the best result reported in [8] for their L2R model. Our baseline results are comparable to their results using LM. However, note that they used Dirchlet smoothing techniques whereas our model uses Jelinek-Mercer smoothing, which might be the reason for small variation in the results.

Table 1. Our baseline results and previous results as reported in [8].

	P@10	Ndcg@10	MRR	P@10	Ndcg@10	MRR
	LM retrieval model					
Results from [8]	0.145	0.134	0.339			
Best model from [8]	LearningToRank (only sentence level features)			LearningToRank (features from neighboring sentences)		
Best results from [8]	**0.174**	**0.159**	**0.344**	**0.194**	**0.180**	**0.403**
Our implementation	LM retrieval model			BM25 model		
Baseline	0.149	0.127	0.293	0.158	0.142	0.330

[2] https://github.com/mmihaltz/word2vec-GoogleNews-vectors.

[3] We learnt different embeddings by varying the training method, dimension size, window size, no. of iterations in internal development experiments, but the results obtained showed little variation in performance.

Table 2. PRF based expansion results, the best scores are in boldface. + and ∗ indicate statistically significant improvement over the baseline with p < 0.05 and p < 0.1 respectively, using student's t-test

	P@10	Ndcg@10	MRR	P@10	Ndcg@10	MRR
	LM retrieval model			BM25 model		
Baseline	0.149	0.127	0.293	0.158	0.142	0.330
Weight, w_t, 0.6	**0.153**	**0.140**	0.319	0.170	0.160	**0.352**
Weight, w_t, 0.7	**0.153**	**0.140**	**0.328**	0.173	**0.162***	**0.352**
Weight, w_t, 0.8	0.150	0.137	0.322	**0.174***	**0.162**$^+$	0.347
Weight, w_t, 0.9	0.149	0.133	0.317	0.164	0.153*	0.346

Pseudo Relevance Feedback (PRF): We varied the number of assumed relevant documents R, and expansion query terms (EQT) in the range of {5, 10, 15, 20}. We linearly varied the weight of initial query terms Q and expansion terms EQT.

$$Expanded\ Query = w_t * Q + (1 - w_t) * EQT \tag{2}$$

where w_t varies in range [0.5, 1], with an increment of 0.1.

For both LM and BM25, the best PRF results were obtained using $R = 10$ and $EQT = 10$, as shown in Table 2, with varying weights between initial and expanded query terms. Based on these results, we fixed values $R = 10$ and $EQT = 10$ for further experiments using PRF. PRF shows significant improvement over both baselines, similar effects were reported in earlier work on sentence retrieval by Allan et al. [1]. The BM25 model performed consistently better than LM, thus we report further results using only BM25.

Word Embeddings (WE): We used the value $z = 10$ for all the experiments, where z determines the number of terms entered into the pool of potential candidate expansion terms for each individual query term as described earlier in Sect. 3. The overall number of expansion terms k selected from the pool was varied as {5, 10, 15}. We linearly varied the weight of query terms and expansion terms as shown in Eq. 2. For each combination of embedding (Global and Local) and the expansion technique (QueryWord and Centroid) the best results (based on P@10) are shown for BM25 in Table 3.

Combining WE and PRF: Table 3 shows results using the combined expansion approach. The number of expanded terms k for expansion using WE was varied as {5, 10, 15} while R=10 and EQT=5 was used for performing PRF. As described in Eq. 1, w_t varies in range [0.1, 1] with an increment of 0.1. For each combination of embedding (Global and Local) and the expansion technique (QueryWord and Centroid) the best results (based on P@10) are presented for BM25. The combined expansion approach WEPRF, performs consistently better than using either the WE and PRF approaches.

Table 3. Embedding based WE and WEPRF approach for sentence retrieval, best scores are in boldface. $*$ indicates that the difference in the results compared to the baseline is statistically significant with $p < 0.1$, using student's t-test. NDCG scores are calculated at rank 10.

	Word embedding (WE)			Combined approach (WEPRF)		
	BM25 model			BM25 model		
	P@10	NDCG	MRR	P@10	NDCG	MRR
Baseline	0.158	0.142	0.330	0.158	0.142	0.330
Best PRF result	**0.174***	**0.162**$^+$	0.347	**0.174***	**0.162**$^+$	0.347
QueryWord approach for semantic expansion: using global embeddings						
Best result	0.158	0.144	0.340	**0.179***	**0.166***	0.357
QueryWord approach for semantic expansion: using local embeddings						
Best result	**0.168**	**0.151**	**0.350**	0.165	0.157	0.352
Centroid approach for semantic expansion: using global embeddings						
Best result	0.165	0.144	0.320	0.173	0.162	0.354
Centroid approach for semantic expansion: using local embeddings						
Best result	0.159	0.142	0.316	0.168	0.160	**0.361**

Analysis: We explored sentence retrieval with QE, best results obtained using WEPRF approach are better than Learning-to-Rank results reported in [8] using only sentence level features, but are slightly lower than the best results using information from neighbourhood sentences. However, the simpler unsupervised technique WEPRF is still quite good and can be used for retrieval problems where data is not sufficient to train effective Learning-to-Rank models.

Using only WE techniques shows that the QueryWord approach trained on *Local embeddings* performs relatively better for QE than using *Global embeddings* or the Centroid approach. *Local embeddings* tend to generate better expanded terms which are more related to the query terms and the corpus. In the combined approach (WEPRF), where the PRF method provides potentially in-context expanded terms using top potential relevant sentences, the best results are obtained using QueryWord approach with *Global embeddings*, which generates more diverse expanded terms (number of expanded terms = 5) to improve the retrieval effectiveness.

We performed manual analysis to analyze QE using WE. For TopicId: 704 *goals green party political views*, both Global and Local embeddings were able to expand and identify most of the common terms such as ("democratic, republic, caucus, candidacy, viewpoints, opinions, ideology, politicians"), and are thus able to capture context which improve the task of sentence retrieval. Further, for TopicId: 741 ("artificial intelligence"), *Global embeddings* learnt using Google ngrams drifted towards "intelligence in security, intelligence agencies and counter-terrorism" aspects, whereas *Local embeddings* were able to capture aspects related to the main query more effectively ("neural, bayesian,

combinatorial, acm, aaai, icml etc"). Further analysis indicates that WE techniques are complementary to PRF techniques, and the combination approach performs better as shown in Table 3.

6 Conclusions and Further Investigation

We explored query expansion techniques for sentence retrieval from a corpus of documents, achieving our best performance for an approach that combines the use of word embeddings and pseudo-relevance feedback. We plan to perform further analysis to learn more about how to exploit semantic representations for this task.

Acknowledgments. We thank the reviewers for their feedback and comments. This research is supported by Science Foundation Ireland (SFI) as a part of the ADAPT Centre at Dublin City University (Grant No: 12/CE/I2267).

References

1. Allan, J., Wade, C., Bolivar, A.: Retrieval and novelty detection at the sentence level. In: Proceedings of SIGIR 2003, pp. 314–321 (2003)
2. Diaz, F., Mitra, B., Craswell, N.: Query expansion with locally-trained word embeddings (2016). arXiv preprint arXiv:1605.07891
3. Keikha, M., Park, J.H., Croft, W.B., Sanderson, M.: Retrieving passages and finding answers. In: Proceedings of the 2014 Australasian Document Computing Symposium, p. 81 (2014)
4. Kuzi, S., Shtok, A., Kurland, O.: Query expansion using word embeddings. In: Proceedings of CIKM 2016, pp. 1929–1932 (2016)
5. Metzler, D., Kanungo, T.: Machine learned sentence selection strategies for query-biased summarization. In: SIGIR Learning to Rank Workshop, pp. 40–47 (2008)
6. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. CoRR, abs/1301.3781 (2013)
7. Robertson, S.E.: On term selection for query expansion. J. Documentation **46**(4), 359–364 (1990)
8. Yang, L., et al.: Beyond factoid QA: effective methods for non-factoid answer sentence retrieval. In: Ferro, N., et al. (eds.) ECIR 2016. LNCS, vol. 9626, pp. 115–128. Springer, Cham (2016). doi:10.1007/978-3-319-30671-1_9
9. Roy, D., Ganguly, D., Mitra, M., Jones, G.J.F.: Word vector compositionality based relevance feedback using kernel density estimation. In: Proceedings of CIKM 2016, pp. 1281–1290 (2016)
10. Ponte, J.M., Croft, W.B.: A language modeling approach to information retrieval. In: Proceedings of SIGIR 1998, pp. 275–281 (1998)
11. Robertson, S., Zaragoza, H., et al.: The probabilistic relevance framework: Bm25 and beyond. Found. Trends® Inf. Retrieval **3**(4), 333–389 (2009)