Information Processing and Management 54 (2018) 657–673

Contents lists available at ScienceDirect

Information Processing and Management

journal homepage: www.elsevier.com/locate/infoproman

Neural word and entity embeddings for ad hoc retrieval

Ebrahim Bagheri*[,a], Faezeh Ensan[b], Feras Al-Obeidat[c]

[a] Laboratory for Systems, Software and Semantics (LS3), Department of Electrical and Computer Engineering, Ryerson University, Canada
[b] Department of Computer Engineering, Ferdowsi University of Mashhad, Mashhad, Iran
[c] College of Technological Innovation, Zayed University, UAE

ARTICLE INFO

Keywords:
Neural embeddings
Ad hoc document retrieval
TREC
Knowledge graph

ABSTRACT

Learning low dimensional dense representations of the vocabularies of a corpus, known as *neural embeddings*, has gained much attention in the information retrieval community. While there have been several successful attempts at integrating embeddings within the ad hoc document retrieval task, yet, no systematic study has been reported that explores the various aspects of neural embeddings and how they impact retrieval performance. In this paper, we perform a methodical study on how neural embeddings influence the ad hoc document retrieval task. More specifically, we systematically explore the following research questions: *(i)* do methods solely based on neural embeddings perform competitively with state of the art retrieval methods with and without interpolation? *(ii)* are there any statistically significant difference between the performance of retrieval models when based on *word* embeddings compared to when knowledge graph *entity* embeddings are used? and *(iii)* is there significant difference between using locally trained neural embeddings compared to when globally trained neural embeddings are used? We examine these three research questions across both *hard* and *all* queries. Our study finds that *word embeddings* do not show competitive performance to any of the baselines. In contrast, *entity embeddings* show competitive performance to the baselines and when interpolated, outperform the best baselines for both hard and soft queries.

1. Introduction

The area of ad hoc document retrieval has received extensive treatment over the past several years whereby different retrieval models have been proposed to connect query and document spaces. Many of these works build on the foundations of language modeling techniques (Ponte & Croft, 1998) and offer variations that focus on certain aspects of the retrieval process such as impact of smoothing techniques (Zhai & Lafferty, 2001), integration of topic models (Wei & Croft, 2006), including external information in the retrieval process (Li, Luk, Ho, & Chung, 2007; Liu, Liu, Yu, & Meng, 2004), term dependency models (Huston & Croft, 2014), and deep neural networks (Guo, Fan, Ai, & Croft, 2016), just to name a few. More recently two additional directions have been recognized to have the potential to impact the document retrieval process, namely, the use of *knowledge graph* information as well as *neural embeddings*. Both of these techniques are focused on extending language models to move beyond a *hard match* between the query and document spaces, hence addressing issues such as the *vocabulary mismatch* problem.

Techniques based on knowledge graphs explore ways in which additional information related to the query or documents can be included in the retrieval process by systematically traversing through or summarizing the information content of the knowledge graph (Nikolaev, Kotov, & Zhiltsov, 2016; Xiong, Power, & Callan, 2017). As such, entity-centric retrieval models have been explored

* Corresponding author.
E-mail addresses: bagheri@ryerson.ca (E. Bagheri), ensan@um.ac.ir (F. Ensan).

https://doi.org/10.1016/j.ipm.2018.04.007
Received 4 November 2017; Received in revised form 10 March 2018; Accepted 17 April 2018
Available online 25 April 2018

(Foley, O'Connor, & Allan, 2016; Hasibi, Balog, & Zhang, 2017) where entities represent knowledge graph concept mentions within the query or documents, which are often extracted using automated semantic annotators (Jovanovic et al., 2014). Given a set of entities, language models are either extended to support for entity information (Hasibi, Balog, & Bratsberg, 2016) or are interpolated with an additional language model built specifically for entities (Raviv, Kurland, & Carmel, 2016). A benefit of employing entities is they provide means for *soft matching* (Guo, Fan, Ai, & Croft, 2016) where semantic similarity measures (Zhu & Iglesias, 2017) can be used to calculate the distance of query–document pairs.

Neural embedding techniques provide a low dimensional yet dense vector representation of the terms while preserving the geometric relations between them; therefore, these methods provide similar benefits to those provided by the soft matching capability of knowledge graph entities (Ganguly, Roy, Mitra, & Jones, 2015). Various methods have been proposed that learn embeddings for documents (Le & Mikolov, 2014), words (Mikolov, Sutskever, Chen, Corrado, & Dean, 2013), entities (Lin, Liu, Sun, Liu, & Zhu, 2015) as well as jointly for words and entities (Toutanova et al., 2015). Neural embeddings have been used in document retrieval for purposes such as query expansion (Diaz, Mitra, & Craswell, 2016), query classification (Zamani & Croft, 2017), ranking (Kuzi, Shtok, & Kurland, 2016) and text classification (Wang et al., 2016), to name a few. Given the growing role of neural embeddings of words and entities in the retrieval literature, it is important to understand their impact on the performance of ad hoc retrieval.

The challenge that are interested to address in this paper is that although earlier techniques in the literature have reported strong and systematic performance results on standard corpora, no methodical and comprehensive work is yet to comparatively report on the various aspects of neural embeddings in ad hoc retrieval. As such, it is not clear from a practical perspective *how* and to *what extent* neural embeddings can positively or negatively impact the ad hoc retrieval task. Therefore, this motivates us to systematically study the impact of neural embeddings on ad hoc retrieval from several cross-cutting aspects that have not been studied before, including (1) the impact of neural embeddings when learnt on words compared to when trained for knowledge graph entities, (2) the difference between learning neural embeddings at local and global scales, (3) the effect of neural embeddings on hard versus soft queries, and finally, the performance of neural embedding-based retrieval models compared to the state of the art.

More specifically, we study:

1. Whether there is any significant performance difference between the use of *word* or *entity* embeddings in the retrieval process or not. Several recent work have reported the impact of considering entities in ad hoc document retrieval. The observation has been that the consideration of entities and features learnt based on the context of entities within the knowledge graph can enhance the performance of keyword-based retrieval models, for instance, some authors have shown that when entities are present in the query, i.e., *entity-bearing queries*, it would be possible to retrieve effective yet non-overlapping documents to the keyword-based models (Ensan, Bagheri, Zouaq, & Kouznetsov, 2017; Liu & Fang, 2015). However, such distinction has not been studied in the embedding-based retrieval models. In other words, would the performance of a retrieval model that uses *word* embeddings be different from a model that uses *entity* embeddings and whether one of the two embedding types is more effective than the other.
2. If using *globally* trained embeddings results in noticeable difference compared to when the embeddings are *locally* trained for the specific retrieval task. Several authors have explored the impact of embeddings on the retrieval process with mixed results. For instance, Zuccon, Koopman, Bruza, and Azzopardi (2015) reported that globally trained embeddings improve the performance of a translation language model, while Diaz et al. (2016) reported that the use of local embeddings, trained based on relevant documents to the query collection, outperforms the performance of global embeddings. In constrast, Rekabsaz, Lupu, Hanbury, and Zamani (2017) argued that using global embeddings can lead to *topic drift*. Therefore, it is important to investigate the impact of global and local embeddings in the context of and in contrast to strong baselines.
3. Whether there is a significant difference on how embedding-based models impact *harder queries* (Carmel, Yom-Tov, Darlow, & Pelleg, 2006) compared to other queries. Some authors have discussed that some retrieval models such as LES (Liu & Fang, 2015) and SELM (Ensan & Bagheri, 2017) are more effective on harder queries, primarily because they identify relevant documents that suffer from the *vocabulary mismatch* problem. For this reason, we explore whether different embedding-based models impact the performance of hard queries differently from the others and as such would embeddings be most appropriately used within the context of such queries.
4. If the interpolation of an embedding-based model with a strong baseline shows any improvement over the state-of-the-art baselines. Most work in the literature have reported their findings of the performance of the embedding-based models when interpolated with a keyword-based retrieval model. For instance, Diaz et al. (2016) as well as Zamani and Croft (2017) interpolate their embedding models with a query language model based on Kullback–Leibler divergence between the query and document. On this basis, its valuable to explore whether there are cases when non-interpolated embedding-based models have competitive performance to the baselines and also would embedding-based models improve the performance of any baseline with which they interpolate with and would they always provide superior performance over a strong baseline.

In order to study these four aspects, we systematically performed experiments based on two large-scale TREC collections, namely ClueWeb'09B and ClueWeb'12B with their related topics (queries). In the experiments, we employ neural embedding representation of words and entities as a way to compute the distance between the query and document spaces. We use the Word Mover's Distance measure for the purpose of distance calculation between a query and a document based on their neural embedding representation. This distance is then used to rank document relevancy score given an input query. The produced rankings are evaluated based on gold standard human-provided relevance judgments already available in the TREC collection and then compared to several state-of-the-art baselines for comparative analysis.

Briefly, we found that word embeddings do not show competitive performance to the baselines in neither interpolated nor non-interpolated models. We further observed that entity embeddings provide competitive performance to the baselines when used without interpolation and show improved performance over the baselines after interpolation. We will extensively report the breakdown of these observations in this paper.

2. Related work

The work in this paper sits at the intersection of neural embeddings and ad hoc document retrieval. There has already been important work in both of these application areas. For instance, neural embeddings have already found many applications such as the work by Fernández-Reyes, Hermosillo-Valadez, and Montes-y Gómez (2018) that has used global word embeddings to perform query expansion based on the semantics of the query terms, and Su et al. (2018) who have proposed to learn bilingual word embeddings for more accurate translations and the work by Ren, Wang, and Ji (2016) that employs topic-enhanced word embeddings for sentiment analysis. From the perspective of document retrieval, Karisani, Rahgozar, and Oroumchian (2016) have explored various query term re-weighting approaches and their efficiency for document retrieval and ranking, while Capannini et al. (2016) have considered the tradeoff balance between quality and efficiency in document retrieval based on learning to rank techniques.

While acknowledging many other important work that have been reported in both neural embeddings and ad hoc retrieval, we will focus on reviewing those that are directly related to the intersection of neural embeddings and ad hoc document retrieval in this paper. Zuccon et al. (2015) are among the few to systematically study the impact of neural embeddings within the context of a translation language model. These authors reported that the incorporation of word embeddings improves the performance of the language model even if the embeddings were trained on a different corpus. Kuzi et al. (2016) train word embeddings specifically based on the document collection on which queries will be executed. They have evaluated the impact of such word embeddings in selecting terms for query expansion as well as interpolating with a relevance model (RM). The authors reported that (semi)locally trained word embeddings lead to a different set of terms in query expansion that are complementary to the baseline and hence can improve retrieval performance when interpolated. Along the same lines, Diaz et al. (2016) train Local Word embeddings based on the top-k retrieved documents of the queries and showed that locally trained embeddings outperform globally trained models. This is inline with our findings in this paper. However, the work in Diaz et al. (2016) does not compare the performance of the locally trained word embeddings with a strong baseline such as EQFE (Dalton, Dietz, & Allan, 2014) as we do in this paper. Therefore while the superiority of local embeddings over global embeddings has been discussed, the performance of the local embeddings even after interpolation has not been evaluated against a strong baseline. From a somewhat different perspective, Rekabsaz et al. (2017) have argued that word embeddings can cause topic shift in retrieval and suggest global context relatedness measures to avoid topic shifts.

Zamani and Croft (2016, 2017) have also studied the impact of neural embeddings on language models in two consecutive work. In the first work, the authors propose to use word embeddings to perform query expansion. The authors have reported that the use of word embeddings, trained globally, improves retrieval performance on AP, Robust and GOV2. Later, the authors offer an innovative framework for learning word embeddings not based on the term co-occurrence framework but rather based on the relevance of the documents to each query. The work by Xiong, Callan, and Liu (2017) is among the few that considers word and entity embeddings in tandem. However, the focus of this work is primarily on how an attention model can improve ranking performance. As such, the paper uses pre-trained embeddings developed in Bordes, Usunier, Garcia-Duran, Weston, and Yakhnenko (2013).

Our work distinguishes itself from the literature in that it:

1. systematically studies the impact of local and global embeddings while comparing them with strong state-of-the-art baselines both with and without interpolation;
2. explores the impact of word and entity embeddings on retrieval performance and reports how these embeddings can improve the baseline that they have been interpolated with and also how they compare with the best baseline;
3. reports on the impact of the different types of embeddings and their interpolation on the retrieval effectiveness for both harder and softer queries, separately.

3. Background

In this section, we introduce the baselines used for comparative analysis and interpolation, and also provide overview of the technique that has been used for learning word and entity embeddings.

3.1. Retrieval baselines

We benefit from three retrieval baselines in our work, which include the following baselines:

3.1.1. Relevance model (RM3)

The relevance model (Lavrenko & Croft, 2001) is a widely used query expansion method that relies on pseudo-relevance feedback to estimate query topics. The expansion terms are interpolated with the original query to enhance retrieval performance. The interpolated model, known as *RM3*, can be formalized as:

$$P(w|\theta_Q^{RM3}) = (1-\lambda)\underbrace{P(w|\theta_Q)}_{\text{query}}\ \text{language model} + \lambda\underbrace{P(w|Q)}_{\text{relevance}}\ \text{model} \tag{1}$$

Here, λ is the interpolation weight controlling the degree of feedback and $Q = \{q_1, q_2, ..., q_k\}$ is the input query. The relevance model can be estimated as:

$$P(w|Q) \approx \sum_{\theta_D \in R} P(w|\theta_D)P(\theta_D)\prod_{i=1}^{k} P(q_i|\theta_D) \tag{2}$$

where R is the set of document models in the psuedo-relevance feedback document collection and $P(\theta_D)$ is some prior over documents. The original query language model can be assumed to be uniform. It has been systematically shown that RM3 and its variations (Lv & Zhai, 2010) are quite effective and robust for ad hoc retrieval (Lv & Zhai, 2009).

3.1.2. Sequential dependence model (SDM)

The SDM model is a Markov Random Field (MRF)-based model used for modeling dependence between query terms through a graph representation where query terms and documents form the nodes and the edges denote dependency between nodes. On this basis, the retrieval model defines three feature functions (Γ_i): *(i)* exact match of each individual constituting query term in the query (Γ_1), *(ii)* exact match of query bi-grams in the document collection (Γ_2), and *(iii)* exact match of unordered query bigrams within the document collection (Γ_3). Similar to RM3, a linear interpolation of these three feature functions forms the ranking score for each document with regards to the query, which can be formalized as:

$$P(D|Q) \approx \sum_{\gamma \in \{\Gamma_1,\Gamma_2,\Gamma_3\}} \lambda_\gamma \sum_{q_i \in Q} \gamma(q_i, D) \tag{3}$$

The matching of each phrase (word or bigram), such as w, is:

$$\gamma(w, D) = \frac{count(w, D) + \mu\frac{count(w,D)}{|C|}}{|D| + \mu} \tag{4}$$

such that C is the document collection and μ is a Dirichlet prior equivalent to the average document length in C. SDM is often used as an efficient benchmark in ad hoc retrieval literature (Huston & Croft, 2014; Zhiltsov, Kotov, & Nikolaev, 2015).

3.1.3. Entity query feature expansion (EQFE)

EQFE (Dalton et al., 2014) is a more recent query expansion method, which benefits from entity information from knowledge bases. This method is also a linear interpolation of four feature functions that are defined on different aspects of the entities observed in the input query. More succinctly, these four feature functions include: *(i) annotated query*, which considers the extraction of terms from query entity's Wikipedia article, *(ii) knowledge base feedback* that can be seen as implicit query-entity linking where the input query is posed to the collection of Wikipedia articles and the ranking of the retrieved results is considered to be the distribution over relevant entities, *(iii) corpus feedback* defines a variation of the relevance model where linked (entity links) or unlinked (named entity) mentions in the query are used for query expansion, and *(iv) entity context model*, which uses feedback documents to build, for each entity, a distribution over words that have been seen in similar contexts to that entity.

An interesting aspect of EQFE is that the feature function weights are obtained through a log-linear learning-to-rank method such that the retrieval effectiveness of the target metric is optimized on an individual feature function basis. EQFE has shown competitive or better performance to SDM and RM and has also been used as a strong baseline in further entity-based retrieval work (Ensan & Bagheri, 2017).

3.2. Neural embeddings

The idea of neural embedding techniques is to project the terms of a high dimensional vocabulary space onto a lower dimensional dense space while maintaining the same geometric properties of the terms. In other words, embedding techniques perform transformations to represent terms through much smaller vector representations that ensure syntactic and semantic relations between terms are respected after the transformation. In this paper, we consider both Continuous Bag of Words (CBOW) and Skipgram models yet empirically show that CBOW models have stronger performance compared to Skipgram models for ad hoc document retrieval. The idea of CBOW is to model each term within the context of the words that surround it within a given window. More specifically, the objective is to minimize J_θ through gradient descent where:

$$\begin{aligned} J_\theta &= -logP(w_c|w_{c-m}, ..., w_{c-1}, w_{c+1}, ..., w_{c+m}) \\ &= -logP(u_c|\hat{v}) \end{aligned} \tag{5}$$

such that u_c is the vector representation for w_c and $\hat{v}$ is the average of the vectors for $w_{c-m}, ..., w_{c-1}, w_{c+1}, ..., w_{c+m}$, more specifically, $\hat{v} = \frac{u_{c-m}+\cdots+u_{c+m}}{2m}$. In a neural embedding model, the conditional probability is modeled through a softmax layer:

$$
\begin{aligned}
-\log P(u_c|\hat{v}) &= -\log\frac{\exp(u_c^\top \hat{v})}{\sum_{i\in V}\exp(u_i^\top \hat{v})} \\
&= -u_c^\top \hat{v} + \log\left(\sum_{i\in V}\exp(u_i^\top \hat{v})\right)
\end{aligned} \tag{6}
$$

In contrast, the Skipgram model learns the conditional probability of the context words given the central term. As such the objective of Skipgram is to minimize J'_θ using gradient descent as follows:

$$
\begin{aligned}
J'_\theta &= -\log P(w_{c-m},...,w_{c-1},w_{c+1},...,w_{c+m}|w_c) \\
&= \prod_{j=0,j\neq m}^{2m} P(w_{c-m+j}|w_c) \\
&= \prod_{j=0,j\neq m}^{2m} P(u_{c-m+j}|v_c)
\end{aligned} \tag{7}
$$

Similar to the CBOW model, a softmax layer can also be used here as follows:

$$
P(u_{c-m+j}|v_c) = \frac{\exp(u_{c-m+j}^\top v_c)}{\sum_{k\in V}\exp(u_k^\top v_c)} \tag{8}
$$

Neural embeddings have shown to improve various information retrieval tasks (Ganguly et al., 2015). We will systematically study how various neural embeddings alone or through interpolation with the baselines can impact retrieval performance. In our work, we employ neural embeddings based on the intersection of two categories: (1) word-based embeddings vs entity-based embeddings, and (2) locally-trained vs globally trained embeddings. This produces four types of embeddings and we extensively evaluate the impact and performance of each of these embeddings on ad hoc retrieval performance.

As later described in Table 3, for the purpose of globally-trained word-based embeddings, we employ already trained word embeddings based on the Google News and Wikipedia corpora. For globally-trained entity-based embeddings, we benefit from the entity embeddings trained based on the Wikipedia corpus and reported in Li et al. (2016). However, for training local embeddings, we pooled top-1000 retrieved documents by our three baselines for each query and created a document collection. Based on this document collection, we use both Skipgram and CBOW models to learn embeddings. For the word-based embeddings, each document in the pooled document collection is considered without any alterations and hence each word in the whole pooled collection will receive an embedding representation. On the other hand, for the locally-trained entity-based representations, we create an alternative document for each document in the pooled collection such that the alternative document only consists of those entities that were observed in the original document in the order they were observed. The collection of these alternative documents are then used to train both Skipgram and CBOW embeddings, hence, producing locally-trained embeddings for entities.

4. Research framework

4.1. Research questions

The main objective of our work is to empirically study the impact of neural embeddings on the effectiveness of ad hoc document retrieval. To this end, we define a set of research questions that will be systematically studied as shown in Table 1.

In the first two research questions (RQs 1 and 2), we will investigate whether neural embeddings alone without interpolation with any baseline methods can show improved retrieval performance. We will refer to the non-interpolated Embedding-based Retrieval Method as EMR. RQ 1 will explore the performance of EMR on all of the queries of the benchmark datasets while RQ 2 will investigate its performance on harder queries. As already suggested in the literature (Dalton et al., 2014; Ensan & Bagheri, 2017), we define *hard queries* to be those that have a Mean Average Precision (MAP) of lower than 0.05 on SDM.

In the subsequent two research questions, RQs 3 and 4, we interpolate EMR with the baseline methods, i.e., RM3, SDM and EQFE. The performance of the interpolated methods are then evaluated both on all queries (RQ 3) as well as hard queries (RQ 4) to see whether any performance improvements are observed. In both of these research questions, the improvements are measured against the baseline with which EMR was interpolated. For instance, for RM3, we study whether the interpolation of EMR and RM3 shows superior performance compared to RM3 or not.

The last two research questions (RQs 5 and 6) are similar to RQs 3 and 4 except that the performance of the interpolated models

Table 1
Overview of research questions.

	Benchmark	All queries	Hard queries
Not interpolated	Best baseline	RQ 1	RQ 2
Interpolated	Interpolated baseline	RQ 3	RQ 4
	Best baseline	RQ 5	RQ 6

are compared against the best performing baseline. For instance, if EQFE is the best performing baseline for a given benchmark, then the performance of the interpolated models will be measured against EQFE. RQs 5 and 6 will determine whether the interpolated models show significant improvement over the best baselines.

4.2. Methodology

The main idea behind EMR is a two step process whereby neural embeddings are used to (1) expand a given input query; and, (2) perform document retrieval using the expanded query. For both tasks, we assume that a neural embedding matrix is available $\boldsymbol{W} \in \mathbb{R}^{d \times n}$ for a vocabulary size of n and d-dimensional embedding vectors. Each column of $\boldsymbol{W}$, denoted as $\boldsymbol{w}_i \in \mathbb{R}^d$ represents the embedding for word i in the vocabulary.

4.2.1. Query expansion

Given $\boldsymbol{W}$, and an input query such as $Q = \{q_1, q_2, ..., q_n\}$, we define $\boldsymbol{V_Q} = \{\boldsymbol{w}[q_1], \boldsymbol{w}[q_2], ..., \boldsymbol{w}[q_n]\}$ to represent the embedding representation of the query where each query term is replaced by its corresponding vector in $\boldsymbol{W}$. With such representation of the query, query expansion can be accomplished through finding the top-k most similar words in $\boldsymbol{W}$ to $\boldsymbol{V_Q}$. The centroid of the vectors in $\boldsymbol{V_Q}$ can be considered to be a suitable representation of the collective meaning of the query defined as:

$$c_{V_Q} = \frac{\sum_{i=1}^{n} w[q_i]}{n} \tag{9}$$

Based on the singular vector representation for the whole query, c_{V_Q}, EMR retrieves the top-k most similar embeddings to this vector for the purpose of query expansion. Approximate nearest neighborhood search methods (Sugawara, Kobayashi, & Iwasaki, 2016) can be used to find the top-k words in efficient processing time.

4.2.2. Document retrieval

The retrieval of the most relevant documents to a query is most often a function of some distance measure between the input query and the individual documents. Considering a similar embedding representation for the documents, it is possible to find the distance between a query and document that can be used for ranking their relevance. One possible way to compute distance between query Q and a document $D = \{d_1, d_2, ..., d_m\}$, represented as $\boldsymbol{V_D} = \{\boldsymbol{w}[d_1], \boldsymbol{w}[d_2], ..., \boldsymbol{w}[d_m]\}$ in the embedding space, is to calculate the distance between c_{V_Q} and c_{V_D}. We employ a more recent method proposed by Kusner, Sun, Kolkin, and Weinberger (2015), known as the Word Mover's Distance, which considers documents to be a set of weighted embedded words. As such the distance between two documents is calculated by the minimum cumulative distance of the best matching embedded word pairs in the two documents. In the context of EMR, the distance between Q and D will be based on transporting words in Q to words in D. As defined in Kusner et al. (2015), the transportation matrix $\boldsymbol{T}$ is a *flow matrix* in which $\boldsymbol{T}_{i,j}$ shows to what degree word i in Q is transported to word j in D. Matrix $\boldsymbol{T}_{i,j}$, which essentially determines what word pairs from the two documents should be connected to each other, needs to be learnt based on a linear optimization program. To this end, given the distance between $\boldsymbol{w}[q_i]$ and $\boldsymbol{w}[d_j]$, as $d(i,j) = \|\boldsymbol{w}[q_i] - \boldsymbol{w}[d_j]\|_2$, the distance between a query and a document can be calculated by minimizing the following linear optimization function:

$$distance_{EMR}(Q, D) = min \sum_{i=1}^{|Q|} \sum_{j=1}^{|D|} \boldsymbol{T}_{i,j} \times d(i,j) \tag{10}$$

In order for the distance function to consider word frequency information, the objective function is minimized in the context of the following constraints:

$$\sum_{i=1}^{|Q|} \boldsymbol{T}_{i,j} = f(d_i) \tag{11}$$

$$\sum_{j=1}^{|D|} \boldsymbol{T}_{i,j} = f(q_j) \tag{12}$$

where $f(x_i)$ is the function that calculates the normalized frequency of word i in document X. EMR ranks and retrieves documents based on the minimum value computed from optimizing Eq. (10) for every query–document pair. It should be noted that while distance metrics based on the word mover's distance, such as the one introduce in this section, achieve accurate distance calculations, they suffer from high time complexity, which is cubical in the number of unique words in the documents. While the discussion on time complexity analysis of word mover's distance is beyond the scope of this paper, it is important to mention that relaxed versions of this distance measure have been proposed and implemented that have linear time complexity (Atasu et al., 2017) and can be used in large-scale information retrieval.

4.2.3. Interpolation

In order to further study the impact of EMR on improving the results of the baselines, we interpolate EMR with each of the baselines individually through a linear interpolation method. The distance produced by the interpolated query language model between each query–document pair is estimated as:

Table 2
The datasets used in the experiments.

Collection	Size	Topics
ClueWeb'09B	50,220,423	1–200
ClueWeb'12B	52,343,021	1–50

Table 3
Description of the embedding models.

	Name	Source	Vocabulary size
Global	Google News	100 billion-word GN dataset	3,000,000
	Wikipedia	English Wikipedia 2014	400,000
	Wikipedia Entity	Li et al. (2016)	5,188,509
Local	CW09 Local Words	Pages pooled from CW09 baselines	1,986,144
	CW09 local entity	Entities pooled from CW09 baselines	27,399
	CW12 Local Words	Pages pooled from CW12 baselines	568,785
	CW12 local entity	Entities pooled from CW12 baselines	7266

$$distanc\ \ e_{EMR+baseline}(q, d)= (\alpha)distance_{EMR}(q, d) + (1 - \alpha)distance_{baseline}(q, d) \quad (13)$$

where each distance is normalized within the collection of document–query pairs for each retrieval model and document collection, separately. As suggested in Zamani and Croft (2017), the hyperparameter α, the linear interpolation coefficient, as well as the number of expansion terms are determined through a ten-fold cross validation strategy for the queries in each collection. We consider the range for α to be [0.01,0.99] with increments of size 0.01 and the number of expansions to range within [5,50] with increments of 5.

5. Empirical evaluation

5.1. Experimental setup

Similar to Xiong et al. (2017), our experiments were carried out on two datasets from the TREC Web track, namely ClueWeb'09B and ClueWeb'12B datasets. Table 2 summarizes the datasets and queries used in our work.

It is worth mentioning that while Google FACC1 data includes entity links for the documents in these corpora, we had a similar observation to Dalton et al. (2014) in that there are missing and noisy entity links in this data; therefore, we opted to automatically perform entity linking on the corpora using TAGME (Ferragina & Scaiella, 2010). Furthermore, the queries used in our experiments included the title field of 200 TREC Web track topics for ClueWeb'09, and 50 Web track topics for ClueWeb'12. We performed automated entity linking on the queries as well. For the baseline runs, we used those provided by Dalton et al. (2014) and the results are produced based on these runs by `trec_eval` and `RankLib-2.8`.[1]

In terms of the neural embeddings, we employ two classes of embeddings in our work, namely *global* embeddings and *local* embeddings. Table 3 provides an overview of the embeddings. Global embeddings are those that have been trained on publicly available corpora including Wikipedia and Google News. The Google News and Wikipedia embeddings provide representations for words, while Wikipedia Entity offers embeddings for Wikipedia entities and categories. Local embeddings were trained separately for words and entities for ClueWeb'09 and ClueWeb'12. For the *Local Words* embeddings, we pooled the top 1000 results from each of the three baselines for every query repeated over all queries and trained a CBOW model. For the *local entity* embeddings, we collected the entities present in the top 1000 results of each baseline for every query and formed separate documents, which were then pooled together and used to train a neural model. The reason 1000 top documents were selected was that the publicly shared runs for each baseline only consists of 1000 retrieved documents per query. For all models, the dimension of the embeddings is 300. We did not observe any difference by varying the dimension size.

5.2. Findings

We systematically report our findings based on the six research questions introduced earlier in Table 1. Note, statistical significance is determined based on a paired t-test with a p-value < 0.05. It should be noted that in order to be able to perform a paired t-test between the performances of some method a and some other method b, we paired the MAP for each query produced by method a with the MAP of the same query produced by method b. Therefore, for k queries, this would produce k pairs of MAP values. We would then use the k pairs of MAP values to calculate statistical significance based on the paired t-test. We also note that we repeated the statistical significance tests reported in all of our results using the non-parameteric Wilcoxon Signed-Ranks test and obtained the same

[1] All runs of the models in this paper are on Github at https://goo.gl/zySQsR.

Table 4
Comparison of the Skipgram and CBOW methods for both word and entity embeddings in terms of MAP.

		All queries		Hard queries	
		ClueWeb'09	ClueWeb'12	ClueWeb'09	ClueWeb'12
		MAP	MAP	MAP	MAP
Words	Skipgram	0.0517	0.0137	0.0124	0.0061
	CBOW	0.0559†	0.0134	0.0123	0.0063
Entity	Skipgram	0.0894	0.021	0.0099	0.0088
	CBOW	0.0963†	0.0389†	0.0124†	0.0116†

The † symbol shows the difference is statistically significant at p-value < 0.05.

Table 5
Comparison of non-interpolated embedding models with three baselines in terms of MAP over 'All Queries'.

		All Queries			
		ClueWeb'09		ClueWeb'12	
		MAP	MAP Δ%	MAP	MAP Δ%
Google News	SDM	0.0533	−42.63	0.0136	−67.70
	RM3	0.0533	−49.67	0.0136	−62.12
	EQFE	0.0533	−43.54	0.0136	−71.00
Wikipedia	SDM	0.0498	−46.5	0.0154	−63.42
	RM3	0.0498	−52.97	0.0154	−57.1
	EQFE	0.0498	−47.25	0.0154	−67.16
Local Words	SDM	0.0559	−39.83	0.0134	−68.41
	RM3	0.0559	−47.21	0.0134	−62.67
	EQFE	0.0559	−40.78	0.0134	−71.43
Wikipedia Entity	SDM	0.0887	−4.52	0.0264	−37.53
	RM3	0.0887	−16.24	0.0264	−26.46
	EQFE	0.0887†	−6.04	0.0264	−43.71
Local entity	SDM	0.0963	3.66	0.0389†	−7.60
	RM3	0.0963	−9.07	0.0389†	8.36
	EQFE	0.0963†	2.01	0.0389	−17.06

The † symbol shows the difference is *not* statistically significant; all other results are statistically significant at p-value < 0.05.

significance results as the paired t-test.

5.2.1. RQs 1 and 2: performance of non-interpolated EMR

As the first step, we compare the performance of neural embedding models trained based on CBOW to those trained based on the Skipgram model. For this purpose, we learn embeddings for both words and entities using the CBOW as well as the Skipgram methods and compare their performance. The comparative performance of CBOW and Skipgram is depicted in Table 4. As shown in the table, the performance of the two methods has been compared not only for words and entities but also over all of the queries and the hard queries. We observed that out of the eight possibilities, CBOW showed statistically better performance over Skipgram in five cases and for the other three cases the performance of the two methods are not statistically different, showing tied performance. Based on Table 4, it is possible to conclude that the CBOW method shows better performance for the ad hoc document retrieval task and hence we opt to report the rest of the results in this paper based on the CBOW method.

Now, we explore how the ranking produced by EMR compares to the three state of the art baselines. Tables 5 and 6 show the systematic comparison of EMR using the five embeddings on the three baselines. The results show that when word-based embeddings are used (Google News, Wikipedia and Local Words), the results are not comparable to any of the three baselines, and statistically worse outcomes are observed. This poorer performance compared to the baselines for word-based embeddings can be observed over both document collections, ClueWeb'09 and ClueWeb'12, and regardless of query difficulty. The results from the first three embeddings (Google News, Wikipedia and Local Words) show that neural embeddings have a better performance on Clueweb'09 while still with a decrease on MAP of at least 49.63% and 10.59%. The performance decrease observed on Clueweb'12 is more substantial. However, entity-based embeddings, including Wikipedia entity and local entity, show better performance compared to word-based embeddings. For instance, Wikipedia entity embeddings show weaker yet comparable results to the baselines on *hard queries* while local entity embeddings show comparable performance on both all queries as well as hard queries to the baselines. In fact, it can be seen that when used, local entity embeddings outperform SDM on all queries for ClueWeb'09 dataset as well as SDM and RM3 on the hard queries of ClueWeb'09. It should be noted that the absolute MAP values reported for the first three word-based embeddings (Google News, Wikipedia and Local Words) cannot be directly compared with entity-based embeddings (Wikipedia entity and local

Table 6
Comparison of non-interpolated embedding models with three baselines in terms of MAP over 'Hard Queries'.

		Hard Queries			
		ClueWeb'09		ClueWeb'12	
		MAP	MAP Δ%	MAP	MAP Δ%
Google News	SDM	0.0123†	−10.59	0.0065	RM3
	RM3	0.0123†	−17.23	0.0065	−48.86
	EQFE	0.0121	−52.05	0.0065	−71.72
Wikipedia	SDM	0.0141†	2.32	0.0071	−60.40
	RM3	0.0141†	−5.28	0.0071	−44.27
	EQFE	0.0141	−44.29	0.0071	−69.26
Local Words	SDM	0.0123	−10.58	0.0063	−64.71
	RM3	0.0123	−17.21	0.0063	−50.33
	EQFE	0.0123	−51.31	0.0063	−72.53
Wikipedia Entity	SDM	0.0114	79.93	0.0075	−27.27
	RM3	0.0114	104.83	0.0075†	−6.69
	EQFE	0.0114†	13.81	0.0075†	−6.69
Local entity	SDM	0.0124	112.35	0.0116†	12.34
	RM3	0.0124	137.10	0.0116†	44.13
	EQFE	0.0124†	32.81	0.0116†	−13.76

The † symbol shows the difference is *not* statistically significant; all other results are statistically significant at p-value < 0.05.

entity) as the latter embeddings are only tested on *entity-bearing* queries (Liu & Fang, 2015; Pantel, Lin, & Gamon, 2012).[2]

In order to explore how the embeddings have helped/hurt the queries in both datasets, we have visualized the improvement or decrease of MAP values for the hard queries compared to the best baseline (EQFE) in Figs. 1 and 3. From the figures it can be seen that (i) the queries in ClueWeb'12 are harder to improve using embeddings and (ii) local entity embeddings are the most successful in skewing the diagram to the left; hence showing improvement on a higher number of queries compared to the other embeddings. We further report the mean retrieval effectiveness of non-interpolated EMR for different embeddings on hard queries broken down based on SDM percentiles in Figs. 4 and 5.

In ClueWeb'09, as expected word-based embeddings consistently perform worse than the best baseline while Wikipedia entity embedding has a comparable performance to the baseline across the percentiles. However, local entity embeddings outperform the baseline in the [25%, 50%] and [75%, 100%] percentiles. A similar pattern can be observed in ClueWeb'12 where the most improvement was observed in the [25%, 50%] percentile by the local entity embedding. Overall, the best baseline shows a slightly better performance on the hardest percentile of the hard queries on both datasets.

Research Questions 1 and 2.

Entity-based embeddings in general, and *local entity* embeddings in specific, show *competitive* performance to baselines without interpolation on CW09 and CW12 and *improved* retrieval performance on hard queries for CW09.

5.2.2. RQs 3 and 4: performance of interpolated EMR on improving each baseline

In order to evaluate the performance of the interpolated EMR with the baselines, we employ Eq. (13) for the purpose of integration as explained earlier. Tables 7 and 8 show the results of the interpolation of EMR for different embeddings with the three baselines. Compared to the non-interpolated EMR, the interpolation with all global embeddings shows improved performance compared to the non-interpolated models; however, all interpolated models based on global embeddings still have weaker performance than the baselines. It should be noted that a similar observation can be made here on the interpolated model in comparison to the non-interpolated where still the Clueweb'09 document collection is easier to perform on compared to Clueweb'12. There are many case on Google News, Wikipedia and Local Words where while the interpolated model shows a weaker performance compared to the baseline but the decrease in performance is not statistically significant.

From the perspective of entity-based embeddings, both Wikipedia entity and local entity embeddings outperform the baseline with which they are interpolated except for the hard query category of the ClueWeb'12 dataset for the specific case when EMR is interpolated with EQFE, which does not improve EQFE itself. The improvements are in eight cases statistically significant. An interesting observation here is that most of the statistically significant improvements are observed when EMR is interpolated with the baselines for the hard queries of ClueWeb'09. Again similar to the non-interpolated EMR, the hard queries of ClueWeb'12 are harder to improve compared to the hard queries of ClueWeb'09 for the interpolated models. An important observation here is that out of the

[2] The list of entity-bearing queries is available: https://goo.gl/zySQsR

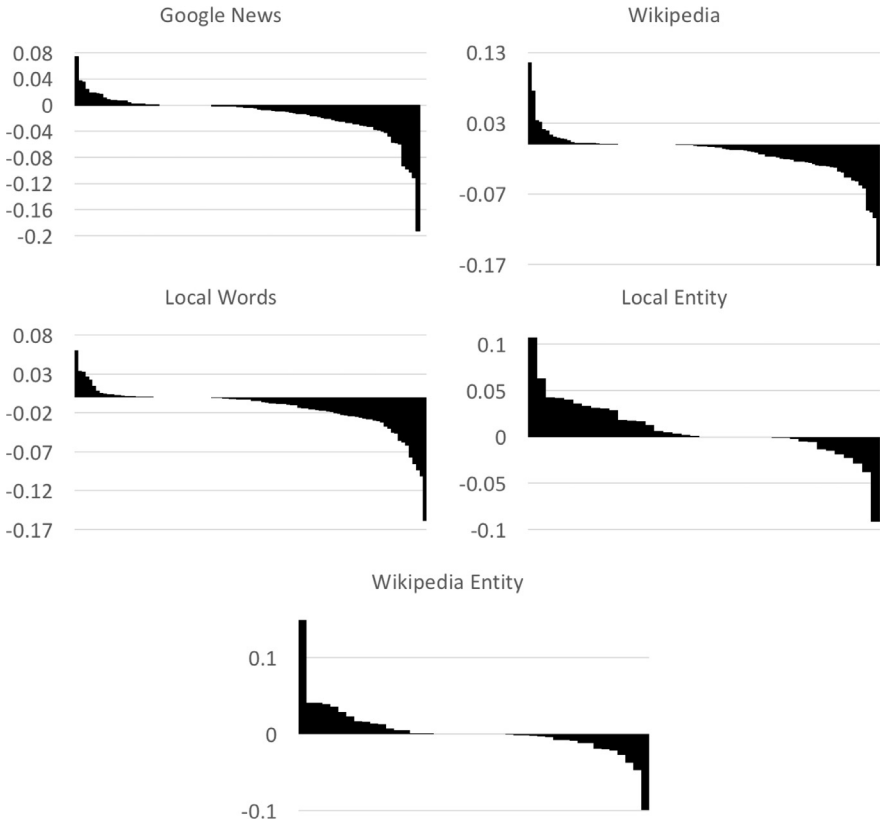

Fig. 1. Improvement of MAP on 'hard queries' (SDM MAP < 0.05) over the best baseline (EQFE) based on *non-interpolated* embeddings over CW09.

Table 7
Comparison of interpolating embedding models with the baselines over 'All Queries'.

		All Queries			
		ClueWeb'09		ClueWeb'12	
		MAP	MAP Δ (%)	MAP	MAP Δ (%)
Google News	SDM	0.0924	−0.54	0.04▽	−4.99
	RM3	0.1057	−0.19	0.0344▽	−4.18
	EQFE	0.0938▽	−0.64	0.0442▽	−5.76
Wikipedia	SDM	0.0924	−0.54	0.04▽	−4.75
	RM3	0.1057	−0.19	0.0344▽	−3.90
	EQFE	0.0938	−0.53	0.0443▽	−5.54
Local Words	SDM	0.0926	−0.32	0.04▽	−4.99
	RM3	0.1057	−0.19	0.0344▽	−4.18
	EQFE	0.0938▽	−0.64	0.0443▽	−5.54
Wikipedia Entity	SDM	0.1172	26.16	0.0469▲	11.40
	RM3	0.1269	19.83	0.0423	17.83
	EQFE	0.114▲	20.76	0.0502▲	7.04
Local entity	SDM	0.1184	27.45	0.0495	17.58
	RM3	0.1282	21.06	0.0431	20.06
	EQFE	0.1118	18.43	0.055	17.27

The ▲ and ▽ symbols show that the increase/decrease is statistically significant at p-value < 0.05; all other results are not statistically significant.

eight statistically significant improvements reported in Tables 7 and 8, five belong to the global entity embeddings learnt based on the Wikipedia corpus pointing to the fact that within the Entity-based neural embeddings models, the embeddings based on global corpora show a reasonable performance.

Table 8
Comparison of interpolating embedding models with the baselines over 'Hard Queries'.

		Hard Queries			
		ClueWeb'09		ClueWeb'12	
		MAP	MAP Δ (%)	MAP	MAP Δ (%)
Google News	SDM	0.0137	−0.93	0.0167▽	−6.44
	RM3	0.0149▽	−41.32	0.0122	−4.73
	EQFE	0.0251	−0.88	0.0209▽	−9.32
Wikipedia	SDM	0.0137	−0.51	0.0167▽	−7.13
	RM3	0.0149	−0.14	0.0122	−4.59
	EQFE	0.0251	−0.84	0.0209▽	−9.47
Local Words	SDM	0.0137	−0.64	0.0167▽	−6.88
	RM3	0.0149	−0.12	0.0122	−4.59
	EQFE	0.0251	1.41	0.0209▽	−9.45
Wikipedia Entity	SDM	0.0069▲	8.84	0.011	6.38
	RM3	0.0065▲	17.31	0.0088	9.29
	EQFE	0.014▲	39.68	0.0126▽	−6.58
Local entity	SDM	0.0061▲	4.66	0.0111	7.67
	RM3	0.0055	4.91	0.0085	5.41
	EQFE	0.0107▲	15.23	0.0152	13.19

The ▲ and ▽ symbols show that the increase/decrease is statistically significant at p-value < 0.05; all other results are not statistically significant.

Research Questions 3 and 4.

Both of the entity-based embeddings have been able to improve the baselines that they have been interpolated with. Hard queries of CW09 have seen the most statistically significant improvement as a result of such interpolation.

We further study the impact of the interpolation co-efficient (α) in Eq. (13) on the performance of the interpolated models. While in all experiments we have used ten-fold cross validation to determine the value for the interpolation co-efficient and number of expansion terms, here and in order to study the impact of the interpolation co-efficient, we determine the best number of expansion terms for different values of α based on a ten-fold cross validation strategy. The results of this study on all queries and hard queries as well as the two different document collections has been reported in Fig. 7. It can be seen that in all four different variations of query type-document collection pairs, the best interpolation co-efficient is consistently 0.1 and values less than or above 0.1 result in worse performance. One interpretation of this could be that while the state-of-the-art baselines carry a strong weight (0.9) in the interpolated models, there is still a role to be played by neural embeddings in terms of covering relevant information that cannot be considered or available to the baselines. As seen in the figure, the two entity-based embeddings show the best performance over all the four cases.

5.2.3. RQs 5 and 6: performance of interpolated EMR on improving the best baseline

The last two research questions investigate whether any of the interpolations have been able to improve the strongest baseline in any of the two datasets for either all or hard queries. Tables 9 and 10 show the results of the most successful interpolation and compares it with the results from the best baseline.

The results show that for *word-based embeddings*, the best performing interpolation is comparable to the baseline on ClueWeb'09,

Table 9
Comparison of the best interpolated embedding model with the best baseline over 'All Queries'.

	All Queries			
	ClueWeb'09		ClueWeb'12	
	MAP	MAP Δ (%)	MAP	MAP Δ (%)
Google News	0.1057	−0.19	0.0443▽	−5.76
Wikipedia	0.1057	−0.19	0.0443▽	−5.54
Local Words	0.1057	−0.19	0.0443▽	−5.54
Wikipedia Entity	0.1269	19.83	0.0502▲	7.04
Local entity	0.1282	21.06	0.055	17.27

The ▲ and ▽ symbols show that the increase/decrease is statistically significant at p-value < 0.05; all other results are not statistically significant.

Table 10
Comparison of the best interpolated embedding model with the best baseline over 'Hard Queries'.

	Hard Queries			
	ClueWeb'09		ClueWeb'12	
	MAP	MAP Δ (%)	MAP	MAP Δ (%)
Google News	0.0251	−0.88	0.0209▽	−9.32
Wikipedia	0.0251	−0.84	0.0209▽	−9.47
Local Words	0.0251	1.41	0.0209▽	−9.45
Wikipedia Entity	0.014▲	39.68	0.0126▽	−6.58
Local entity	0.0107▲	15.23	0.0152	13.19

The ▲ and ▽ symbols show that the increase/decrease is statistically significant at p-value < 0.05; all other results are not statistically significant.

albeit a slightly weaker yet statistically insignificant difference. However, the best performing interpolation using these embeddings shows significantly weaker performance on the ClueWeb'12 dataset. For entity-bearing queries, both Wikipedia entity and local entity embeddings show noticeable improvement when interpolated over the best baseline, except for Wikipedia entity on the hard queries of ClueWeb'12. Local entity embeddings provide consistent improvement over the best baseline in both all and hard queries.

In order to delve deeper into the performance of the interpolations, Figs. 2 and 6 show how much the best interpolated model has been able to help or hurt the best baseline on the hard queries. As seen in Fig. 2 as well as Tables 9 and 10, Wikipedia entity and local entity embeddings show similar behavior in terms of the number and degree of impact they have on the hard queries of the ClueWeb'09 dataset. However, the best interpolation involving Wikipedia entity embeddings shows statistically significant weaker performance compared to the best baseline, while local entity embeddings show an improved performance. This is observed in Fig. 6 where a much larger number of queries have been helped in the interpolation involving the local entity embeddings in comparison to Wikipedia entity embeddings.

We also show the mean retrieval effectiveness of the interpolated EMR models on hard queries of both datasets according to SDM percentiles in Figs. 8 and 9. For the ClueWeb'09 dataset, both entity-based embeddings show comparative performance that exceeds

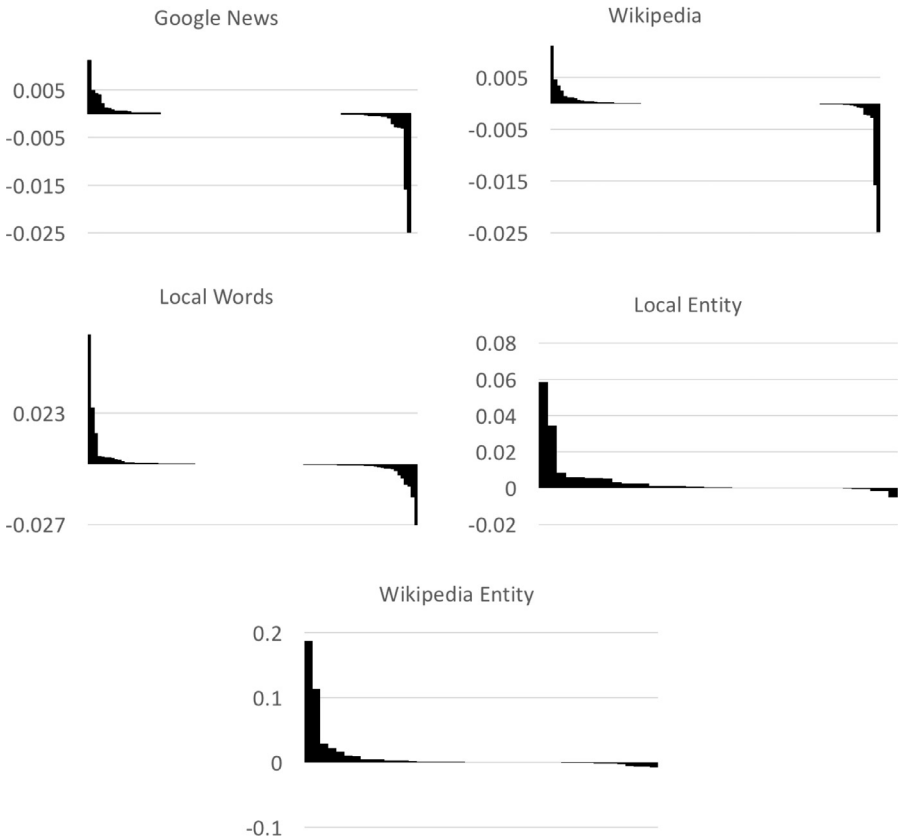

Fig. 2. Improvement of MAP on 'hard queries' (SDM MAP < 0.05) over the best baseline (EQFE) based on *interpolation* with different embeddings over CW09.

Fig. 3. Improvement of MAP on 'hard queries' (SDM MAP < 0.05) over the best baseline (EQFE) based on different non-interpolated embedding features over the ClueWeb 2012 dataset.

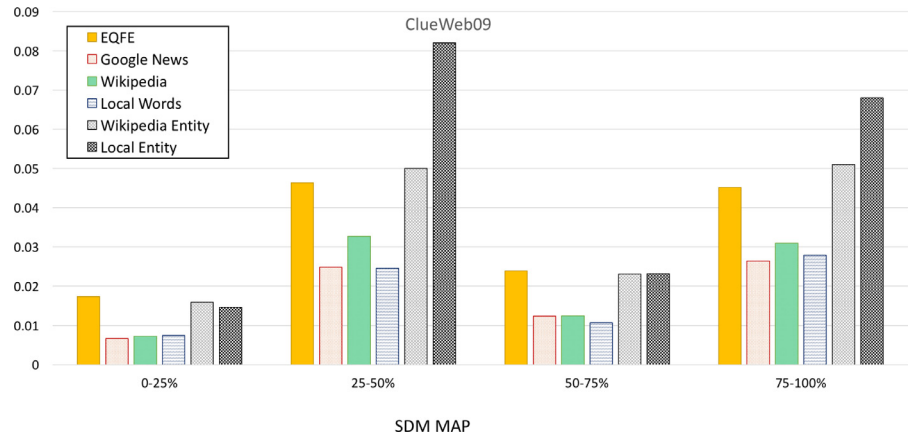

Fig. 4. Mean retrieval effectiveness of the non-interpolated embeddings across SDM percentiles for 'hard queries' on CW09.

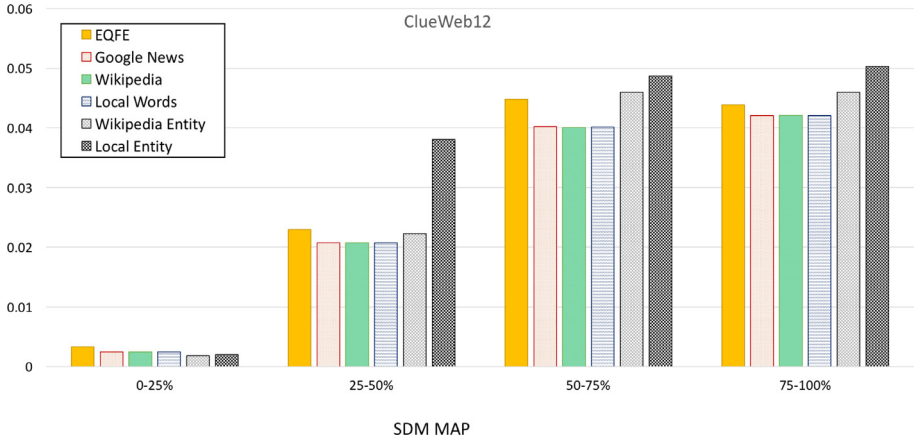

Fig. 5. Mean retrieval effectiveness of the non-interpolated embeddings across SDM percentiles for 'hard queries' on CW12.

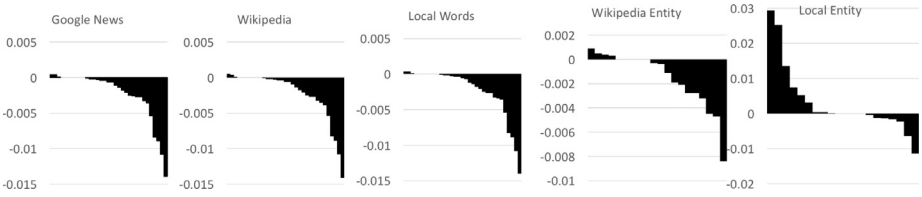

Fig. 6. Improvement of MAP on 'hard queries' (SDM MAP < 0.05) over the best baseline (EQFE) based on interpolation with different embedding features over the ClueWeb 2012 dataset.

the best baseline (EQFE) in all percentiles. An interesting observation can be made when comparing Fig. 8 with Fig. 4 at the [0, 25%] percentile. All interpolated models have been able to significantly outperform the best baseline in this percentile, which did not happen in any of the non-interpolated models. Additionally, it can be seen that the stronger performance of the entity embeddings

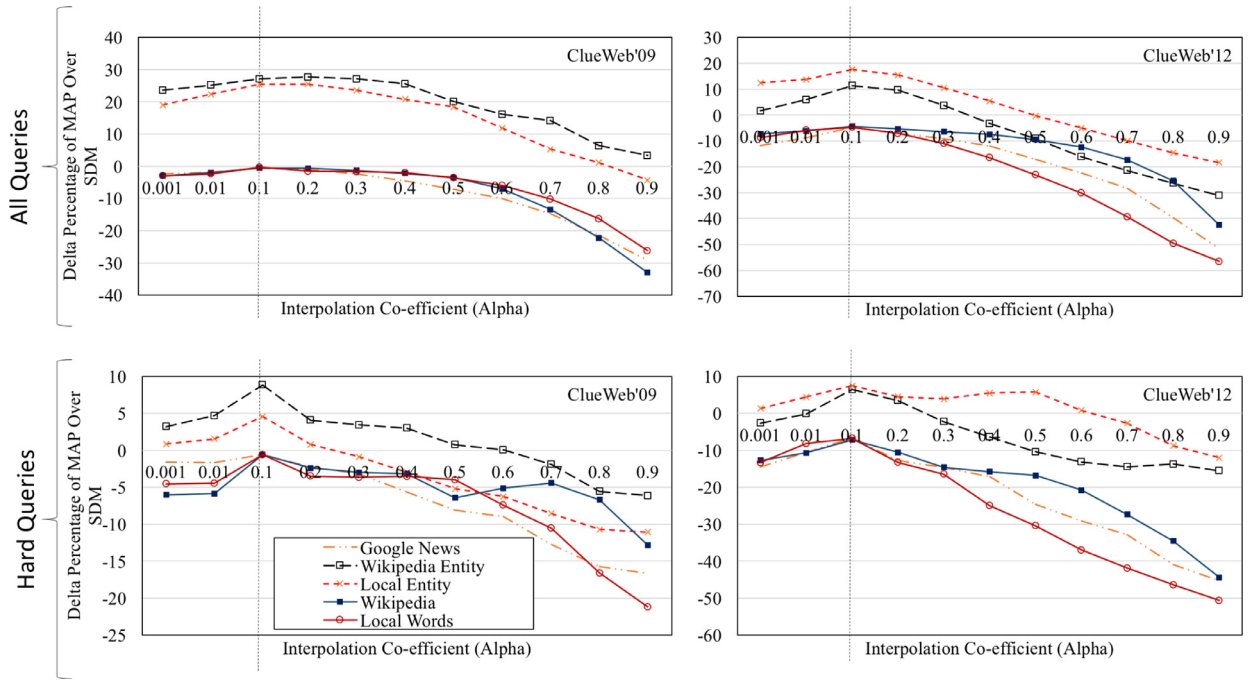

Fig. 7. The impact of the interpolation co-efficient on retrieval performance.

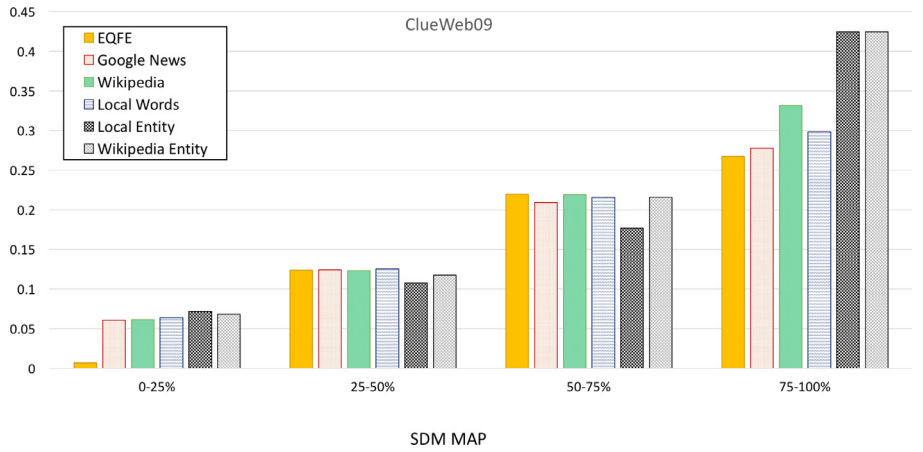

Fig. 8. Mean retrieval effectiveness of interpolation of embeddings across SDM percentiles for 'hard queries' on CW09.

comes primarily from their stronger performance on the queries from the [75%, 100%] percentile. In terms of the ClueWeb'12 dataset, Fig. 9 closely resembles the behavior observed in Fig. 5 where local entity embeddings show consistent stronger retrieval effectiveness compared to the other embeddings and the baseline.

Research Questions 5 and 6.

Regardless of being learnt locally or globally, when interpolated with any of the baselines, word-based embeddings perform weaker than the best baseline. In contrast, entity-based embeddings when interpolated, provide consistent positive improvement over the best baseline for both all and hard queries.

5.2.4. Discussions

We can now present a clear picture of the findings based on the research questions of this paper as follows:

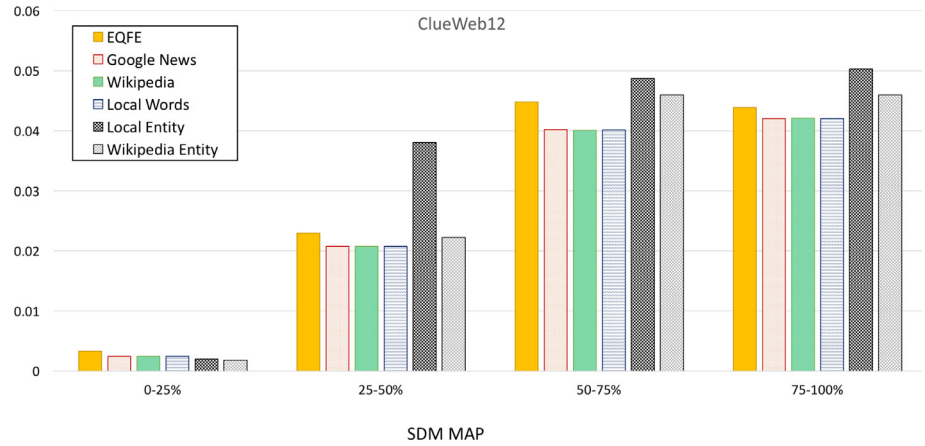

Fig. 9. Mean retrieval effectiveness of interpolation of embeddings across SDM percentiles for 'hard queries' on CW12.

1. Based on RQs 1 and 2, it is clear that neural embeddings, in non-interpolated form, cannot exceed the performance of state-of-the-art baselines. However, looking more carefully at the findings, it is evident that entity embeddings when applied to *hard* queries have competitive and in some cases statistically significant improvement over the state-of-the-art baselines. Therefore, an important finding and contribution of our work is that the application of neural embeddings on *hard* queries can be an effective strategy to increase retrieval performance;
2. A similar observation can be made in the findings related to RQs 3 and 4 where while word-based embeddings do not show competitive performance even after interpolation with state-of-the-art baselines, entity-based neural embeddings can be effective in boosting the retrieval performance of the baselines when interpolated with them. This is especially true on *hard* queries. The findings of RQs 1–4 also point to the fact that a more systematic way of interpolating neural embeddings with the baseline methods could be to automatically determine query difficulty and determine which model to use to address the query based on its difficulty, e.g., *harder* queries to be handled by the EMR model in this case;
3. Finally, based on RQs 5 and 6, we observe that the best interpolated models come from entity-based neural embeddings. Therefore, we conclude that the important factor in the effectiveness of a neural embedding model for ad hoc retrieval is whether it is word-based or entity-based and has less to do with whether it was trained locally or globally. Therefore, given the training of local embeddings can have limitations in practice, the adoption of globally trained Entity-based neural embeddings within the ad hoc retrieval process can be an efficient strategy.

An important point to mention here is the feasibility of implementing the global and local embeddings in practice. This is important because while global embeddings are trained regardless of the input query and document spaces, local embeddings are trained based on an observed set of queries and the top-k associated retrieved documents for each of the queries by some baselines. This raises the question of practical implementation of local embeddings. We notice when reviewing the final interpolation results of the different models (the peaks reached at $\alpha = 0.1$ in Fig. 7) that while the entity-based embeddings are far superior to word-based embeddings, they are themselves (global vs local) quite competitive in terms of performance. In fact, global embeddings, i.e., Wikipedia Entity embeddings, outperform the Local Embeddings on Clueweb'09 and show competitive performance in case of hard queries on Clueweb'12. Therefore, one possible solution would be to use global embeddings for cases when the query has not been observed in the past and periodically retrain the neural embeddings as new batches of queries are observed. We believe it can be an interesting and impactful line of future research to look at query characteristics to determine which queries benefit the most from local or global embeddings. This can be related to query characteristics such as its generality or specificity, among others.

It should be also noted that similar to any other empirical study, the findings of this paper are limited to the extent of the methodology presented here. It might be possible that different forms of interpolation such as query-level interpolation (Ensan & Bagheri, 2017), alternative methods for determining the linear interpolation coefficient (Dalton et al., 2014), use of a different document collection (Rekabsaz et al., 2017; Zuccon et al., 2015) or even the application of other methods for performing query expansion and/or document retrieval based on embeddings (Diaz et al., 2016; Kuzi et al., 2016), could result in varying observations. However, we have ensured reproducibility of our work by publicly sharing all artifacts.

6. Concluding remarks

The main objective of this work has been to systematically study the impact of both local and global word and entity embeddings on the ad hoc document retrieval task. Our empirical study has shown that in word-based embeddings, although local embeddings perform stronger than global embeddings, they do not perform as well as the best state-of-the-art baselines even after interpolation. Furthermore, it was observed that entity-based embeddings not only show competitive performance to the baselines before interpolation but also have the most consistent improvement when interpolated with the baselines. As such, it seems that entity-based

neural embeddings, learnt either globally or locally, can potentially enhance the process of handling entity-bearing queries and positively impact entity-centric information retrieval. Summarily, we find that entity-based embeddings are stronger models compared to word-based embeddings.

References

Atasu, K., Parnell, T. P., Dünner, C., Sifalakis, M., Pozidis, H., Vasileiadis, V., et al. (2017). *Linear-complexity relaxed word mover's distance with GPU acceleration. 2017 IEEE International Conference on Big Data, Big Data 2017, Boston, MA, USA, December 11-14, 2017*889–896. http://dx.doi.org/10.1109/BigData.2017.8258005.

Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., & Yakhnenko, O. (2013). *Translating embeddings for modeling multi-relational data. Proceedings of the 2013 advances in neural information processing systems (NIPS)*2787–2795.

Capannini, G., Lucchese, C., Nardini, F. M., Orlando, S., Perego, R., & Tonellotto, N. (2016). Quality versus efficiency in document scoring with learning-to-rank models. *Information Processing & Management, 52*(6), 1161–1177.

Carmel, D., Yom-Tov, E., Darlow, A., & Pelleg, D. (2006). *What makes a query difficult? Proceedings of the twenty-ninth annual international ACM SIGIR conference on research and development in information retrieval, SIGIR '06*. New York, NY, USA: ACM390–397. http://dx.doi.org/10.1145/1148170.1148238.

Dalton, J., Dietz, L., & Allan, J. (2014). *Entity query feature expansion using knowledge base links. Proceedings of the thirty-seventh international ACM SIGIR conference on research & development in information retrieval*. ACM365–374.

Diaz, F., Mitra, B., & Craswell, N. (2016). *Query expansion with locally-trained word embeddings. Proceedings of the fifty-fourth annual meeting of the association for computational linguistics*367–377.

Ensan, F., & Bagheri, E. (2017). *Document retrieval model through semantic linking. Proceedings of the tenth ACM international conference on web search and data mining*. ACM181–190.

Ensan, F., Bagheri, E., Zouaq, A., & Kouznetsov, A. (2017). *An empirical study of embedding features in learning to rank. Proceedings of the twenty-sixth ACM international conference on information and knowledge management (CIKM)*2059–2062.

Fernández-Reyes, F. C., Hermosillo-Valadez, J., & Montes-y Gómez, M. (2018). A prospect-guided global query expansion strategy using word embeddings. *Information Processing & Management, 54*(1), 1–13.

Ferragina, P., & Scaiella, U. (2010). *TAGME: on-the-fly annotation of short text fragments (by Wikipedia entities). Proceedings of the nineteenth ACM international conference on information and knowledge management*. ACM1625–1628.

Foley, J., O'Connor, B., & Allan, J. (2016). *Improving entity ranking for keyword queries. Proceedings of the twenty-fifth ACM international on conference on information and knowledge management*. ACM2061–2064.

Ganguly, D., Roy, D., Mitra, M., & Jones, G. J. (2015). *Word embedding based generalized language model for information retrieval. Proceedings of the thirty-eighth international ACM SIGIR conference on research and development in information retrieval*. ACM795–798.

Guo, J., Fan, Y., Ai, Q., & Croft, W. B. (2016). *A deep relevance matching model for ad-hoc retrieval. Proceedings of the twenty-fifth ACM international on conference on information and knowledge management*. ACM55–64.

Guo, J., Fan, Y., Ai, Q., & Croft, W. B. (2016). *Semantic matching by non-linear word transportation for information retrieval. Proceedings of the twenty-fifth ACM international on conference on information and knowledge management*. ACM701–710.

Hasibi, F., Balog, K., & Bratsberg, S. E. (2016). *Exploiting entity linking in queries for entity retrieval. Proceedings of the 2016 ACM on international conference on the theory of information retrieval*. ACM209–218.

Hasibi, F., Balog, K., & Zhang, S. (2017). *Nordlys: A toolkit for entity-oriented and semantic search. Proceedings of the annual international ACM SIGIR conference on research and development in information retrieval*1289–1292.

Huston, S., & Croft, W. B. (2014). *A comparison of retrieval models using term dependencies. Proceedings of the twenty-third ACM international conference on conference on information and knowledge management*. ACM111–120.

Jovanovic, J., Bagheri, E., Cuzzola, J., Gasevic, D., Jeremic, Z., & Bashash, R. (2014). Automated semantic tagging of textual content. *IT Professional, 16*(6), 38–46.

Karisani, P., Rahgozar, M., & Oroumchian, F. (2016). A query term re-weighting approach using document similarity. *Information Processing & Management, 52*(3), 478–489.

Kusner, M., Sun, Y., Kolkin, N., & Weinberger, K. (2015). *From word embeddings to document distances. Proceedings of the 2015 international conference on machine learning*957–966.

Kuzi, S., Shtok, A., & Kurland, O. (2016). *Query expansion using word embeddings. Proceedings of the twenty-fifth ACM international on conference on information and knowledge management, CIKM'16*. New York, NY, USA: ACM1929–1932. http://dx.doi.org/10.1145/2983323.2983876.

Lavrenko, V., & Croft, W. B. (2001). *Relevance based language models. Proceedings of the twenty-fourth annual international ACM SIGIR conference on research and development in information retrieval*. ACM120–127.

Le, Q., & Mikolov, T. (2014). *Distributed representations of sentences and documents. Proceedings of the thirty-first international conference on machine learning (ICML-14)* 1188–1196.

Li, Y., Luk, W. P. R., Ho, K. S. E., & Chung, F. L. K. (2007). *Improving weak ad-hoc queries using Wikipedia asexternal corpus. Proceedings of the thirtieth annual international ACM SIGIR conference on research and development in information retrieval, SIGIR '07*. New York, NY, USA: ACM797–798. http://dx.doi.org/10.1145/1277741.1277914.

Li, Y., Zheng, R., Tian, T., Hu, Z., Iyer, R., & Sycara, K. (2016). *Joint embedding of hierarchical categories and entities for concept categorization and dataless classification. Proceedings of the COLING 2016*2678–2688.

Lin, Y., Liu, Z., Sun, M., Liu, Y., & Zhu, X. (2015). *Learning entity and relation embeddings for knowledge graph completion. Proceedings of the 2016 AAAI conference on artificial intelligence*2181–2187.

Liu, S., Liu, F., Yu, C., & Meng, W. (2004). *An effective approach to document retrieval via utilizing wordnet and recognizing phrases. Proceedings of the twenty-seventh annual international ACM SIGIR conference on research and development in information retrieval, SIGIR '04*. New York, NY, USA: ACM266–272. http://dx.doi.org/10.1145/1008992.1009039.

Liu, X., & Fang, H. (2015). Latent entity space: A novel retrieval approach for entity-bearing queries. *Information Retrieval Journal, 18*(6), 473–503.

Lv, Y., & Zhai, C. (2009). *A comparative study of methods for estimating query language models with pseudo feedback. Proceedings of the eighteenth ACM conference on information and knowledge management*. ACM1895–1898.

Lv, Y., & Zhai, C. (2010). *Positional relevance model for pseudo-relevance feedback. Proceedings of the thirty-third international ACM SIGIR conference on research and development in information retrieval*. ACM579–586.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). *Distributed representations of words and phrases and their compositionality. Proceedings of the 2013 advances in neural information processing systems (NIPS)*3111–3119.

Nikolaev, F., Kotov, A., & Zhiltsov, N. (2016). *Parameterized fielded term dependence models for ad-hoc entity retrieval from knowledge graph. Proceedings of the thirty-ninth international ACM SIGIR conference on research and development in information retrieval*. ACM435–444.

Pantel, P., Lin, T., & Gamon, M. (2012). *Mining entity types from query logs via user intent modeling. Proceedings of the fiftieth annual meeting of the association for computational linguistics1. Proceedings of the fiftieth annual meeting of the association for computational linguistics* Association for Computational Linguistics563–571 Long papers

Ponte, J. M., & Croft, W. B. (1998). *A language modeling approach to information retrieval. Proceedings of the twenty-first annual international ACM SIGIR conference on research and development in information retrieval*. ACM275–281.

Raviv, H., Kurland, O., & Carmel, D. (2016). *Document retrieval using entity-based language models. Proceedings of the thirty-ninth international ACM SIGIR conference on research and development in information retrieval*. ACM65–74.

Rekabsaz, N., Lupu, M., Hanbury, A., & Zamani, H. (2017). *Word embedding causes topic shifting; exploit global context!. Proceedings of the 2017 annual international ACM SIGIR conference on research and development in information retrieval*.

Ren, Y., Wang, R., & Ji, D. (2016). A topic-enhanced word embedding for twitter sentiment classification. *Information Sciences, 369*, 188–198.

Su, J., Wu, S., Zhang, B., Wu, C., Qin, Y., & Xiong, D. (2018). A neural generative autoencoder for bilingual word embeddings. *Information Sciences, 424*, 287–300.

Sugawara, K., Kobayashi, H., & Iwasaki, M. (2016). *On approximately searching for similar word embeddings. Proceedings of the annual meeting of the association for computational linguistics (ACL)*.

Toutanova, K., Chen, D., Pantel, P., Poon, H., Choudhury, P., & Gamon, M. (2015). *Representing text for joint embedding of text and knowledge bases. Proceedings of the 2015 empirical methods in natural language processing, EMNLP15. Proceedings of the 2015 empirical methods in natural language processing, EMNLP* 1499–1509.

Wang, P., Xu, B., Xu, J., Tian, G., Liu, C.-L., & Hao, H. (2016). Semantic expansion using word embedding clustering and convolutional neural network for improving short text classification. *Neurocomputing, 174*, 806–814.

Wei, X., & Croft, W. B. (2006). *LDA-based document models for ad-hoc retrieval. Proceedings of the twenty-ninth annual international ACM SIGIR conference on research and development in information retrieval, SIGIR'06*. New York, NY, USA: ACM178–185. http://dx.doi.org/10.1145/1148170.1148204.

Xiong, C., Callan, J., & Liu, T.-Y. (2017). *Learning to attend and to rank with word-entity duets. Proceedings of the annual international ACM SIGIR conference on research and development in information retrieval*763–772.

Xiong, C., Power, R., & Callan, J. (2017). *Explicit semantic ranking for academic search via knowledge graph embedding. Proceedings of the twenty-sixth international conference on world wide web*. International World Wide Web Conferences Steering Committee1271–1279.

Zamani, H., & Croft, W. B. (2016). *Embedding-based query language models. Proceedings of the 2016 ACM on international conference on the theory of information retrieval*. ACM147–156.

Zamani, H., & Croft, W. B. (2017). *Relevance-based word embedding. Proceedings of the annual international ACM SIGIR conference on research and development in information retrieval*505–514.

Zhai, C., & Lafferty, J. (2001). *A study of smoothing methods for language models applied to ad hoc information retrieval. Proceedings of the twenty-fourth annual international ACM SIGIR conference on research and development in information retrieval, SIGIR '01*. New York, NY, USA: ACM334–342. http://dx.doi.org/10.1145/383952.384019.

Zhiltsov, N., Kotov, A., & Nikolaev, F. (2015). *Fielded sequential dependence model for ad-hoc entity retrieval in the web of data. Proceedings of the thirty-eighth international ACM SIGIR conference on research and development in information retrieval*. ACM253–262.

Zhu, G., & Iglesias, C. A. (2017). Computing semantic similarity of concepts in knowledge graphs. *IEEE Transactions on Knowledge and Data Engineering, 29*(1), 72–85.

Zuccon, G., Koopman, B., Bruza, P., & Azzopardi, L. (2015). *Integrating and evaluating neural word embeddings in information retrieval. Proceedings of the twentieth Australasian document computing symposium, ADCS '15*. New York, NY, USA: ACM Article 12, 8 pages. doi: 10.1145/2838931.2838936.