Conditional Variational Autoencoder for Query Expansion in Ad-hoc Information Retrieval

Wei Ou[a], Van-Nam Huynh[b]

[a]*International Business School, Zhejiang Gongshang University*
[b]*the School of Knowledge Science, Japan Advanced Institute of Science and Technology*

Abstract

Query expansion (QE) is commonly used to boost the performance of traditional information retrieval (IR) models. Along with the adoption of deep learning techniques in the IR research community, deep neural based QE models have been proposed in recent years. Many of these models suffer difficulties to incorporate the relevance and interactions between queries and documents. In this research we aim to bridge the gap. We model the relevance by treating QE as a generative problem, and assume the relevant documents of a query are generated from language models depending on the query. We propose a query expansion (QE) model, called QECVAE, and a BERT-variant of it, based on the conditional variational autoencoder (CVAE), that take queries and documents as input, and approximate the language models based on their interactions. The proposed models are trained with relevance feedback (RF) data, and generate expansion terms with pseudo relevance feedback (PRF) data based on the language models. The proposed models are evaluated on two standard TREC collections: Robust 04 and TREC 2019 Deep learning. The results suggest that, QECVAE outperforms RM3, which is a robust traditional QE model; the BERT variant outperforms Neural-PRF, a state-of-the-art neural QE model. The results also highlight that combining RM3 and the proposed models leads to even future improvements over the baseline QE models.

Keywords: Information retrieval, query expansion, conditional variational autoencoder

1. Introduction

As Information on the internet is growing exponentially, internet users are faced with increasing difficulties to navigate through the vast ocean of information. Information retrieval, that is serving users the information relevant to their queries, has become an increasingly essential but challenging task in many applications. Many information retrieval models have been proposed over the past decades. Early models usually compute the relevance of a document to a query based on exact matching between the terms in the two. However, since a typical query contains a few terms whereas a document may contain thousands, there exists a possibility that they share no common words even they are highly relevant.

An obvious solution to the problem, which is usually referred to as 'vocabulary gap' , is expanding short queries issued by users with more terms from the document vocabulary [1]. The research of query expansion can be dated back to 1960s. Roccio et al.[2] first proposed to expand a query with the mean of the representation vectors of the query's relevant documents. Lavrenko et al. [3] proposed the relevance model (RM) and its RM3 variant, that essentially use the frequent terms in pseudo relevant documents as expansion terms. Although the idea of these early models is very simple, they have been proven extremely effective and widely adopted in both academia and industry. Even in today's world where the emerging deep learning techniques are overwhelming the traditional probabilistic models in many natural language processing tasks, a well-tuned RM3 is still a very tough baseline to beat [4].

One obvious shortcoming of the relevance models is that the frequent terms in pseudo relevant documents can be possibly background terms, that are loosely relevant to the original queries. To address the problem, methods that employ external knowledge resources [5, 6, 7, 8] or neural based embedding models [9, 10, 11] , which allow for comparing the semantic similarities among words, have been proposed in recent years. Among them, the neural embedding models have received the most attention because they can be trained with unsupervised learning algorithms and take almost no human annotation efforts.

Unfortunately, so far, the embedding models have not shown clear advantages over

RM3 and other traditional QE models [12]. One possible reason is that they usually suffer difficulties to incorporate the relevance and interactions between queries and documents. This subjects the embedding models to high risk of topic drift, though the expansion terms generated by them are semantically similar to the query terms. Guo at el. [13] argued that semantic similarities not necessarily translate into query-document relevance, which is the primary objective of almost all IR systems.

In this research we aim to bridge the gap. We model the relevance by treating QE as a generative problem. Given a query and a relevant document, we assume the terms in the document are generated from a probability distribution over the vocabulary depending upon the query. We propose a novel model based on the conditional variational autoencoder, called QECVAE, and a BERT [14] variant of it, that take query and (pseudo) relevant document pairs as input, and estimate the generative distributions based on their interactions as output.

1.1. Research objectives and contributions

A comprehensive survey of query expansion models shows that the interactions and relevance between queries and documents are usually ignored in selection of expansion terms. This research aims to bridge the gap with the following objectives.

- To find a way to incorporate the relevance and interactions into the generative framework of CVAE.
- To integrate the BERT model, which has shown dominantly strong performance in IR tasks, into the proposed approach for query expansion.
- To evaluate the effectiveness of the proposed models, and compare it against existing QE models.

The contributions of this work can be summarized as follows.

- First, to the best of our knowledge, this paper is the first to discuss the application of CVAE to query expansion for ad-hoc information retrieval.
- Second, to the best of our knowledge, this work is the first neural-based QE model that combines relevance feedback (RF) and pseudo relevance feedback

(PRF) data to generate topic terms to expand queries (though the recent Neural-PRF [15] that also combines the RF and PRF data, it essentially expands original queries with PRF documents, instead of individual topic terms, which are usually more explainable).

- Third, we evaluate the proposed models on the Robust 04 and TREC 2019 Deep Learning datasets. Experiment results suggest that, the proposed models, especially the BERT-variant, outperform a number of robust traditional and neural QE baselines. We also find that the expansion terms generated by the proposed models and RM3 are somehow complementary to each other. Combining the two leads to further performance advantages over the baselines.

The rest of this paper is arranged as follows. Section 2 gives a brief introduction to related work. Sections 3-6 introduce the proposed models in great detail. Section 7 introduces the experimental setup and Section 8 presents the evaluation results. Section 9 gives a brief discussion and Section 10 concludes this paper with future research directions.

2. Related work

We divide existing query expansion models into the following categories based on their working methodologies : external knowledge based, non-neural relevance feedback based, word embedding based and deep learning based. Also, as stated in the Introduction section, deep neural matching models are involved in the proposed model, a brief introduction to these models is also presented at the end of this section.

2.1. External knowledge based QE methods

Liu et al. [5] used WordNet to disambiguate word senses of query terms, and selected their synonyms, hyponyms, and definition words or phrases as the expansion terms. Gong et al. [6] assigned query terms into different groups based on their semantic similarities in WordNet. For each group, the authors used the hypernyms and synonyms of the highest terms in the WordNet hierarchies to expand the original queries. Pal et al. [7] used the terms whose definitions overlap that of query terms the most in

WordNet as the expansion terms. Kotov et al. [8] used ConceptNet to construct concept graph for a query and expanded it with terms associated with the query's adjacent concept nodes. Goslin et at. [16] expanded a query with the frequent terms from its pseudo relevant pages in Wikipedia. Azad et al. [17] found that the expansion terms produced by Wikipedia and WordNet based methods are usually complementary to each other, and proposed an weighting algorithm to combine the expansion results from the two.

2.2. Non-neural Pseudo Relevance feedback based QE models

The Rocchio algorithm [2] is one of the earliest query expansion methods. This method represents queries and documents with vector space models, and expands queries with the centroid of the representation vectors of their pseudo relevant documents. Church et al. [18] expanded queries with terms from pseudo relevant documents that frequently co-occur with the query terms in the entire corpus. The authors proposed a method based on mutual information to measure the co-occurrence. Latiri et al. [19] proposed a method based on association rules to identify expansion terms that frequently co-occur with query terms from pseudo-relevant documents. Carpineto et al. [20] computed the KL divergence between the distributions of a word in a set of pseudo relevant documents and the whole corpus, and chose the ones with the highest divergence to expand the queries. Zhai et al. [21] proposed a mixture model that assumes each word is sampled either from a query-specific topic model or a background topic model. The authors estimated the query-specific topic models by maximizing the likelihood of the pseudo relevant documents and used them to sample expansion terms. Miao et al. [22] expanded the Rocchio algorithm by incorporating the proximity between query and candidate expansion terms. Lavrenko et al.[3] proposed the relevance model (RM) that assumes both query and expansion terms are sampled from the language models of the relevant documents. The joint probability of a query and a candidate term can be estimated by summing over their generative probabilities under each of the language models. RM3 extends RM by incorporating the co-occurrence frequencies between query and candidate expansion terms into the probability framework of RM. Nasir et al. [23] first extracted frequent terms from pseudo relevant documents, then employed external knowledge resources to evaluate semantic relatedness of them

to queries to select expansion terms.

2.3. Embedding based QE models

Embedding models [24] project words, even long pieces of text into continuous semantic spaces, in which the similarity between words or texts can be easily computed. Kuzi et al. [9] chose the terms whose embeddings are the nearest to the average of query term embeddings as the expansion terms (referred to as Average Word Embedding or AWE hereinafter). Diaz et al. [25] argued that word embeddings trained globally on topically unconstrained corpus may produce expansion terms that are too general to capture search intents. The authors proposed to train word embedding models on topic specific corpora and showed that the locally trained word embeddings outperform the globally trained word embedding models. Zamani et al. [10] argued that generally purposed word embedding models focus on semantic proximity between terms, that are not necessarily applicable in IR tasks, where the primary objective is usually to capture query-document relevance. The authors proposed an embedding model that incorporates the relevance into the generation process of document terms. Wang et al. [26] first applied BERT [14] to rank documents to generate high quality pseudo relevant documents, then employed existing PRF based QE models to produce expansion terms. Similarly, Zheng et al. [27] first applied BERT to rank documents, then selected the most relevant chunks from the top N ranked documents to expand queries.

2.4. Deep learning based QE models

Recently, along with wide adoption of deep learning techniques in natural language processing tasks, deep learning models have also been used for query expansion. Imain et al. [28] treated query expansion as a classification problem and proposed a supervised Siamese network to estimate the relevance of a candidate term to a query. For a query, the authors annotated each candidate term with a 'good' or 'bad' quality label. To overcome the lack of training data, the authors generated 2-combinations of the labels to augment the data, and trained the model by an objective of predicting whether each pair of terms belonging to the same quality classes. Liu et el. [29] proposed an

autoencoder model for query expansion in code search in which the encoder takes original queries as the input and the decoder generates method names depending upon the encoding. Instead of expanding queries, Nogueira et al. [30] proposed to expand documents with relevant query terms. The authors trained a sequence-to-sequence transformer model that predicts relevant queries for each document, and concatenated them to the document, in order to increase the chance of the document and an unseen relevant query overlapping. In stead of expanding queries with topic terms, Wang et al.[15] proposed the Neural-PRF model that expands queries with pseudo relevant documents. In the model, the authors first estimated the matching scores of a candidate document to the pseudo relevant documents based on neural matching models, and then weight each of the scores by the relevance between the respective pseudo document and the query, to aggregate them as the final matching score for the candidate document.

2.5. Neural matching models

One of the earliest neural matching models is the DSSM model proposed by Huang et al [31]. In this model the authors projected queries and relevant documents into a semantic space and used the cosine similarities between their representations in the space to predict the marching scores. Inspired by DSSM, a number of extensions that share similar structure but employ more expressive projection functions, such as RNN, CNN, tree-structured neural networks, have also been proposed [32, 33, 34]. Different from DSSM and its extensions that focus on learning good query and document representations for semantic matching, Guo et al. [13] proposed the DRMM model with a focus on encoding the interactions between queries and documents. In the model, for a query and a document, the authors first computed the local interactions between each query term and the document, then aggregated the interactions with a gating mechanism to estimate the matching score. Mitra et al. [35] proposed the DUET model that considers both semantic matching and the interactions. The model consists of two separate deep neural networks, one encodes the local interactions based on exact matching, the other their semantic similarities. The output of the two networks are combined to estimate the matching score.

In recent years a number of neural matching models based on BERT have also been

proposed. Nogueira et al.[36] first applied BERT for passage ranking. The authors fed queries as sentence A and document passages as sentence B into BERT. Based on the resulting [CLS] vectors, the authors built a binary classifier to predict the relevance between the query-document pairs. However, since the inputs of BERT are limited to 512 tokens, it is out of the question to directly use BERT to estimate the relevance scores for documents, which usually contain more than 512 tokens. To deal with problem, Yilmaz et al.[37] proposed the Birch model that segments each document into its constituent sentences and uses BERT to compute the matching scores between the query and each of the sentences. The authors weighed the matching scores using cross-validation to aggregate them into document-level scores. Similarly, Li et al.[38] computed the relevance scores for document passages and used a neural network to learn their weights to aggregate them into document-level scores.

3. Methodology overview

Given a relevant document d of query q, QECVAE assumes d is generated by the following process:

- Draw a continuous latent variable by $z \sim P(z|q)$
- Draw every word w for d from $w \sim P(w|z, q)$

Based on the assumption, the evidence likelihood of d given q is formulated as follows:

$$P(d|q) = \int_z P(z|q) \prod_{w \in d} P(w|z, q) dz \tag{1}$$

In the training process, we are interested in learning the above distributions based on RF data. In the test phase, given a test query $\hat{q}$ and a pseudo relevant document $\hat{d}$, we first sample a z from $P(z|\hat{d}, \hat{q})$, then compute $P(w|z, \hat{q})$, the generative distribution over the vocabulary. The top K terms with the highest weights in the distribution can be used as the expansion terms.

However, because of the presence of the integral operator in Equation (1) , $P(d|q)$ is intractable to compute. Theoretically, it is possible to estimate $P(d|q)$ by sampling

many values for z from $P(z|q)$. However, since only a trivial fraction of the sampled values in a high dimensional space contribute to d, it takes an extremely large amount of samples to arrive at a good estimation and is usually out of the question in practice. A more practical estimation strategy is sampling z from a simple distribution $Q(z|d,q)$, that serves as an approximation of the posterior $P(z|d,q)$, to increase the possibility that the sampled values for z are relevant to the observations. To make $Q(z|d,q)$ a good approximation, the KL divergence $\mathcal{D}[Q(z|d,q)||P(z|d,q)]$ must be minimized.

To minimize $\mathcal{D}[Q(z|d,q)||P(z|d,q)]$, we expand $\log P(d|q)$ as follows:

$$\begin{aligned}
\log P(d|q) &= \log \int_z P(z|q)P(d|z,q)dz && (2)\\
&= \log \int_z P(z|q)P(d|z,q)\frac{Q(z|d,q)}{Q(z|d,q)}dz && (3)\\
&= \log E_{z\sim Q(z|d,q)}\frac{P(z|q)P(d|z,q)}{Q(z|d,q)} && (4)\\
&\geq E_{z\sim Q(z|d,q)}[\log P(d|z,q)] - \mathcal{D}[Q(z|d,q)||P(z|q)] && (5)
\end{aligned}$$

The bottom term is the evidence lower bound (ELBO). It can be easily verified that the difference between $\log P(d|q)$ and the ELBO is equal to $\mathcal{D}[Q(z|d,q)||P(z|d,q)]$. Therefore, minimizing the KL divergence is equivalent to maximizing the ELBO. To maximize the ELBO, for simplicity, $Q(z|d,q)$ is assumed to be a multivariate Gaussian distribution whose parameters depend upon d and q, $P(w|z,q)$ a categorical distribution parameterized by z and q, $P(z|q)$ a multivariate Gaussian distribution parameterized by q.

The ELBO consists of the two terms: $E_{z\sim Q(z|d,q)}[\log P(d|z,q)]$ and $\mathcal{D}[Q(z|d,q)||P(z|q)]$. Because $Q(z|d,q)$ and $P(z|q)$ are assumed to be multivariate Gaussian distributions, the $\mathcal{D}[Q(z|d,q)||P(z|q)]$ part can be expanded into an analytical form that allows for computing the gradients of the ELBO with respect to the parameters of the distributions. However, in the $E_{z\sim Q(z|d,q)}[\log P(d|z,q)]$ part, it is impossible to compute the gradients because it involves sampling from $Q(z|d,q)$ whose parameters are to be learned. A popular approach to overcome the problem is using the 're-parametrization'

technique to rewrite the expectation into such a form :

$$E_{z\sim Q(z|d,q)}[\log P(d|z,q)] = E_{\epsilon\sim\mathcal{N}(0,I)}[\log P(x|z=\mu_Q+\Sigma_Q^{1/2}\epsilon,q)] \qquad (6)$$

where ϵ is a random variable sampled from $\mathcal{N}(0,I)$, μ_Q and Σ_Q are the mean and covariance of the variational distribution. Now the expectation is with respect to $\mathcal{N}(0,I)$, instead of $Q(z|d,q)$. The parameters of $Q(z|d,q)$ are explicitly linked with the ELBO that allows for computing the gradients.

In this research we use a neural network (referred to as 'encoder network' hereinafter) that takes a query-document pair to compute the mean and covariance of $Q(z|d,q)$, and use another network (referred to as 'decoder network' hereinafter) that takes q and z as input to compute the parameters for $P(w|z,q)$. Though the parameters of $P(z|q)$ can also be treated in the same way, we follow the popular approach in existing CVAE research literature by assuming it is the standard multivariate normal distribution. The encoder and decoder networks can be learned by maximizing the ELBO using back-propagation and gradient based optimization algorithms.

4. QECVAE

4.1. Encoder network

Suppose a query consists of N terms, a document M terms. Each term is represented by a L-dimensional embedding vector. For each query term vector, we compute its element-wise product with all the document term vectors. This operation results in a $L * M$ matrix for each query. We feed the matrix to a max-pooling layer to get a L-dimensional vector, which keeps the most salient signals from the matrix, for each query term. We combine the resulting vectors and feed them into a 1-D convolutional layer consisting of F filters, to arrive at a $F * L$ matrix. We keep the most salient signal of each row of the matrix by max-pooling to obtain a F-dimensional vector. We pass the vector to a dense layer to obtain the mean and covariance of the variational distribution $Q(z|d,q)$. The architecture of the encoder network is shown in Figure 1.

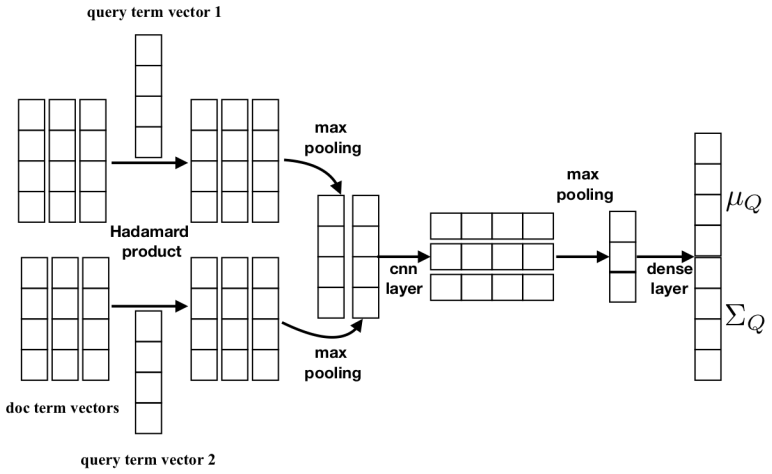

Figure 1: The architecture of the encoder network of QECVAE

4.2. Decoder network

In the decoder, we first combine the query term vectors and pass them into a 1-D convolutional layer consisting of F filters, to arrive at a $F * L$ matrix. We keep the most salient signal of each row of the matrix by max-pooling to obtain a F-dimensional vector. We then concatenate the vector with a latent vector z sampled from the variational distribution $Q(z|d, q)$, and feed the concatenated vector a dense layer to obtain the parameters of the categorical distribution over the vocabulary $P(w|q, z)$. The architecture of the decoder is shown in Figure 2.

5. BERT-QECVAE

The BERT model, which has shown dominantly strong performance in IR, can be easily integrated into the framework of the proposed model.

5.1. Encoder

By following the passage-ranking approach proposed by Nogueira et al. [36], given a query-document pair, we feed the query as sentence A, and the document as sentence B into the BERT model. We pass the resulting [CLS] vector to a dense layer to obtain the parameters of the variational distribution $Q(z|d, q)$. Since the input length of the

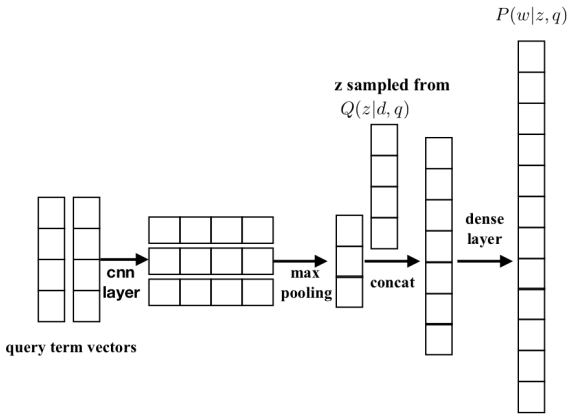

Figure 2: The architecture of the decoder network of QECVAE

BERT model is limited to 512, it is out of the question to feed the entire document into the BERT as a single input. Instead, we truncate each query to 50 tokens, and truncate each document to 400 tokens, in order to reduce the computation burden. (We can also split a document into multiple passages of 512 tokens, and compute the [CLS] vectors for each pair of the query and the passages. However we found this approach significantly increase computation cost).

5.2. Decoder

In the decoder, we feed the query into BERT to obtain the [CLS] vector. We then sample a value for z by $z \sim Q(z|d,q)$ and concatenate it to the [CLS] vector. We feed the result into the dense layer to compute the parameters of the categorical distribution $P(w|q,z)$.

6. Training and query expansion

The proposed models are trained by maximizing the ELBO on RF data. In the training process, for each query-document pair in the RF data, we sample $\epsilon \sim \mathcal{N}(0,I)$, then map it to z by the re-parameterization technique. We compute the gradients of $P(w|z,q) - \mathcal{D}[Q(z|d,q)||p(z|q)]$ with respect to all the distribution parameters. The averages of the gradients at many document-query pair samples converge to the true gradients of the ELBO.

In testing phase, for a test query $\hat{q}$ and a pseudo relevant document $\hat{d}$, we sample a value for z from $Q(z|\hat{d}, \hat{q})$ and feed it to the decoder to compute the term distribution . Assume there are M pseudo relevant documents for the query, we aggregate the respective term distributions by:

$$P(w) = \sum_{m=1}^{M} \frac{s_m}{\sum_j s_j} * P_m(w) \tag{7}$$

where $P_m(w)$ denotes the term distribution computed by the decoder for the m^{th} document, s_m is the matching score of the document in the initial retrieval results. We select the top N terms with the highest weights in the distribution as the expansion terms. We normalize their weights and combine them with the original queries to rerank the retrieved documents based on exact matching based methods, i.e., BM25. It is also possible to combine the proposed QE models with RM3 simply by:

$$P(w) = \alpha * P_{rm3}(w) + (1 - \alpha) P_{qecvae}(w) \tag{8}$$

where $\alpha \in [0, 1]$ is an interpolation coefficient that controls the weights of the two models in the expansion results.

7. EXPERIMENTAL SETUP

7.1. Datasets and evaluation metrics

The proposed QECVAEs are evaluated on two standard benchmark datasets: Robust 04 and TREC 2019 Deep Learning. Robust 04 contains 528,155 documents and 249 'topcis'. Each topic consists of a title, and a description that further explains the title. Both the titles, and descriptions are used separately as queries in our experiment. TREC Deep Learning contains around 3.2 million documents, 367,013 training queries and 43 test queries. Some important statistics of the two datasets are shown in Table 1

For Robust 04, MAP, P@20, NDGC@20 that have been widely adopted in research papers are used to evaluate the performance of all models by the 5-fold cross-validation introduced by [39]. For TREC Deep Learning, all models are evaluated by the official NDCG@10, MRR, and MAP on the test split of the data.

Table 1: Dataset Statistics. T in the Robust 04 row represents the title query, D the description query

	Num of queries	Avg query length	Num of docs	Avg doc length	Num of relevant query-doc pairs
Robust 04 title	250	3(T)/15(D)	528,155	654	17,412
TREC 2019	360,000	6	3,213,834	1218	360,000

7.2. Baselines

In the experiment, query likelihood model (QL) [40] and BM25 [41] are used as the base retrieval models. The proposed QECVAE, and the BERT-QECVAE variant, and their combinations with RM3 are compared with the following QE models and neural matching models:

- Traditional PRF-based QE models
  - RM3
- Neural-based QE models
  - Neural-PRF
  - AWE
- Neural matching models
  - DSSM
  - DRMM
  - DUET
  - Birch

To test RM3 and AWE and the proposed models, we use them to generate terms from PRF documents to expand the original queries and rescore the retrieved documents by BM25. To test Neural-PRF and the neural matching models, we directly use them to rerank the retrieved documents.

7.3. Experimental setup

7.3.1. Baselines

In the experiment, we use the Anserini toolkit [4] to perform tokenization, stemming and indexing. The hyper-parameters of all the non-BERT models are tuned using grid search, based on MAP for Robust 04 and NGCG@10 for TREC Deep Learning . The b and $k1$ in BM25 are set to [0, 1.0] with step size 0.1, and [1.0, 2.0] with step size 0.1, respectively. For QL, the Dirichlet smoothing parameter μ is set to [300, 600] with step size 100. For RM3 and AWE, the number of documents from which the expansion terms are generated is set to [50,100] with step size 10, the number of expansion terms is set to [30, 120] with step size 30, and the weight of the original query term is set to [0.3,1] with step size 0.1. We use Google's W2V embedding model [42] that is pre-trained on the Google News corpus (3 billion words) and fine-tune it on Robust 04 and TREC Deep Learning, respectively, to build the AWE model.

For the neural DSSM, DUET and DRRM models, by following the setup shown in [4], we use their implementations in Match-Zoo with default hyper-parameter settings. Also, the aforementioned W2V embedding vectors are used to initialize their embedding layers if applicable. For Neural-PRF, we use the DRMM model as the base neural matching model. By following the fine-tune procedure shown in the original paper [15], we set the number of PRF documents to [5,30] with step size 5, and the number of terms for each document to [10, 60] with step size of 10. For the Birch matching model we use the fine-tuned models released by the authors [37] with default parameter setting.

For QECVAE, the encoding size is set to [300,1000] with step size 200, the number of PRF documents [1, 10] with step size of 2, the number of expansion terms [30,80] with step size 10, the number of 1-D filters in the convolutional layers [200,600] with step size 200. We extract the top 100 tokens with the highest TF-IDF weights from each document to represent the document. The inputs of the encoder are query tokens and the top document tokens, and the outputs of the decoder are the document tokens. For the BERT-QECVAE variant, we set the encoding size to 400. We use the BERT-Large model [14] as the building block for the encoder and decoder. The inputs of

the encoder are a query and the first chunk of 400 tokens of a relevant document, the outputs of the decoder are the top document tokens. For the combinations of the two models with RM3, the interpolation weight is set to $[0.1, 1]$ with step size 0.2.

7.4. Data scarcity problem

Since Robust 04 is a small dataset with around 20,000 relevant query-document pairs, we augment it with relevance feedback data from TREC 2019. However, since the documents in Robust 04 are articles from some major news outlets, the documents in TREC 2019 are webpages, the word distributions of the two datasets are very different. We cannot simply mix all the relevant query-document pairs from the two datasets. Instead, we propose to use the TREC 2019 queries to retrieve documents from Robust 04 using the exacting matching based BM25 model, and select only those whose top hits enjoy matching scores larger than a certain threshold. In the experiment we retrieve documents for the queries with the Ansirerni toolkit and select approximately 15,000 relevant query-document pairs from TREC 2019 with a matching score threshold of 4.6.

For the BERT-QECVAE variant, we follow the fine-tuning procedure shown in the Birch paper [37]. We first train the variant on TREC 2019, and then use the fine-tuned BERT to initialize the variant for training on Robust 04.

8. Results

In this section we show the performance comparison of QECVAEs and the baselines. Also, we study the quality of the expansion terms generated by the QE models.

8.1. Performance comparison

The performance of all models is summarized in Table 2 and 3 , where the best result in metric column is highlighted in bold. On the Robust 04 dataset, all the QE models lead to performance improvements to the base retrieval model. Among them, AWE leads to the least improvement. One obvious reason is that it does not encode the relevance between queries and documents. QECVAE outperforms RM3 in every metric with statistical significance. Neural-PRF outperforms QECVAE in all the 3

metrics. However, the combination of QECVAE and RM3 outperforms Neural-PRF in terms of NDCG@20 and P@20 with statistical significance. This may indicate the expansion terms generates by the models are somehow complementary to each other. The BERT-QECVAE variant outperforms Neural-PRF by significant margins of more than 2% in P@20 and NDCG@20. This shows that BERT is also very effective in generating query expansion terms. The combination of BERT-QECVAE and RM3 has slight advantages over the BERT-QECVAE variant in all the 3 metrics.

Among the non-BERT neural matching models that , DRMM is the best performing model. This may come from that DRMM employs a gating mechanism that explicitly assigns higher weights to topical words, while DSSM and DUET may have difficulties in identifying important terms on the small dataset. Birch has the best performance among all the neural matching models and enjoys a margin of at least 4% in every metric.

On the description queries, the performance of all the non-BERT QE models decreases, because the description text contains many noisy background terms. RM3 suffers more obvious performance decrease than QCEVAE. One possible reason is that RM3 is essentially based on the simple word-count approach and very susceptible to the noise of background terms. QECVAE and its combination with RM3 enjoy higher values than Neural-PRF in all the 3 metrics, especially in MAP. Unlike the non-BERT QE models, BERT-QECVAE enjoys a gain of approximately 1% in NDCG@20 and P@20 compared to its performance on the title query. This indicates the robustness of BERT in encoding the contextual information of long queries. On the neural matching models, similarly, all non-BERT models suffer decreases in performance, whereas Birch enjoys a performance gain of approximately 2% in NDCG@20.

On the TREC Deep Learning dataset, all query expansion models lead to performance improvements, as they do on Robust 04. Among them, once again, AEW leads to the least improvement. QECVAE outperforms Neural-PRF in MRR and NDCG@10, while the combination of QECVAE and RM3 outperforms the baseline in all the 3 metrics, especially in MRR by a margin of around 2%. This further demonstrates additivity in the performance gains of RM3 and QECVAE. BERT-QECVAE and its combination with RM3 outperform Neural-PRF by margins of 2-4% in all the 3 metrics.

Among all the 3 non-BERT neural matching models, DUET enjoys the strongest performance. This is possibly attributed to that, compared with DRMM and DSSM, it also encodes the exact-matching between queries and documents. However, though it employs complex neural networks, its performance is very close to that of RM3. Birch enjoys the strongest performance among all the neural matching and QE models. However, it is also the most time-consuming one because it computes relevance score of every sentence (of 512 length) in a document to a query. By contrast, the proposed model uses BERT to compute the interaction between the query and only one chunk of 400 tokens from the document. We compare the retrieval time per query each model consumes in Figure 3. As the figure shows, Birch takes more than two times longer than the BERT-QECVAE variant on both datasets. In a realistic ad-hoc search system low latency is a must factor.

Table 2: Performance comparison on Robust 04. For QECVAEs and their combinations with RM3, *denotes statistical significance over RM3, **denotes statistical significance over Neural-PRF (p-value < 0.05)

Model	Title query			Description query		
	P@20	MAP	NDCG@20	P@20	MAP	NDCG@20
BM25	0.3701	0.2545	0.4159	0.3372	0.2405	0.3963
QL	0.3690	0.2539	0.4155	0.3335	0.2466	0.3909
AWE	0.3775	0.2559	0.4229	0.3424	0.2371	0.4062
RM3	0.4027	0.3040	0.4474	0.3792	0.2825	0.4255
DSSM	0.3952	0.3043	0.4412	0.3858	0.2854	0.4213
DRMM	0.4152	0.3193	0.4671	0.4007	0.3055	0.4473
DUET	0.4072	0.3053	0.4500	0.3975	0.2952	0.4393
Birch	**0.4599**	**0.3625**	**0.5165**	**0.4636**	**0.3691**	**0.5355**
Neural-PRF	0.4154	0.3200	0.4656	0.4026	0.2855	0.4479
QECVAE	0.4104 *	0.3124 *	0.4540 *	0.4035 *	0.3029 **	0.4506 *
BERT-QECVAE	0.4362 **	0.3362 **	0.4885 **	0.4492 **	0.3455 **	0.4967 **
RM3+QECVAE	0.4191 **	0.3179 *	0. 4755 **	0.4062*	0.3008 **	0.4510 *
RM3+BERT-QECVAE	0.4394 **	0.3412 **	0.4897 **	0.4496 **	0.3501 **	0.4943 **

8.1.1. Analysis of expansion terms

We analyze the quality of the expansion terms generated by the QE models. We randomly select 4 queries from each dataset and use RM3, QECVAE, BERT-QECVAE to generate expansion terms for the queries. We show the top 10 expansion terms

Table 3: Performance comparison on TREC Deep Learning

Model	TREC 2019		
	MRR	MAP	nDCG@10
BM25	0.3176	0.2311	0.5236
QL	0.2927	0.2275	0.5173
AWE	0.3262	0.2354	0.5422
RM3	0.3639	0.2820	0.5542
DSSM	0.3656	0.2855	0.5554
DRMM	0.3731	0.2985	0.5587
DUET	0.3752	0.3012	0.5632
Birch	**0.4553**	**0.3645**	**0.6667**
Neural-PRF	0.3856	0.3152	0.5904
QECVAE	0.3978	0.3123	0.5972
BERT-QECVAE	0.4198	0.3357	0.6189
QECVAE+RM3	0.4067	0.3256	0.6079
RM3 + BERT-QECVAE	0.4285	0.3359	0.6245

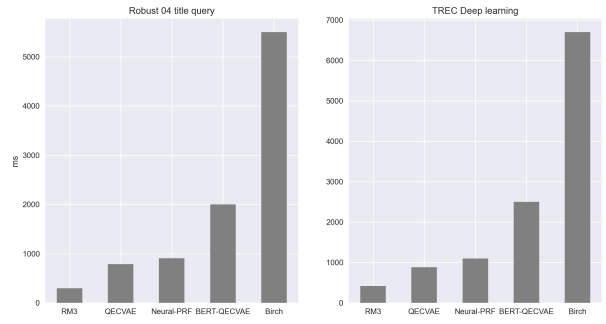

Figure 3: Time efficiency comparison on Robust 04 and TREC 2019 Deep Learning. The y-axis indicates how many milliseconds each model consumes to produce the ranking results

generated by each model in Table 5 and 6 . As shown in the table, among the top 10 terms of RM3, almost half of them are the terms that also appear in the queries . This is caused by that RM3 essentially extracts the most frequent terms from the PRF documents, in which the frequencies of query terms are high. For example, it generates 'sky', 'captain', 'world', 'tomorrow' as the expansion terms for the query 'cast of sky captain and the world of tomorrow' ; 'origin', 'going', 'last-name' for the query 'what origin is the last name goins', though they already exist in the original queries. However, there exists a possibility that some highly ranked PRF documents retrieved by BM25 are not relevant to the original queries at all, therefore, using the frequent terms from these documents increases the risk of introducing irrelevant expansion terms. For example, RM3 introduces 'breast' as an expansion term for the query 'What are the advantages and/or disadvantages of tooth implants?', caused by that BM25 mistakenly ranks some documents related to 'breast implant' highly in its retrieval results.

The expansion terms generated by QECVAE and its BERT variant are more related to the answers to the queries. This suggests their effectiveness in encoding the semantic meaning of the queries. On the TREC 2019 data, the BERT-QECVAE variant ranks 'france' as the top expansion term for query 'what origin is the last name goins', 'republican' the query 'what party is paul ryan in' , which are the correct answers to the queries. QECVAE also shows similar behavior, but suffers risk of producing expansion terms that are loosely relevant to the original queries. For example it produces 'july' for the query 'cast of sky captain and the world of tomorrow', that possibly be the release month of the movie. This suggests that it is less effective than the BERT variant to capture the contextual information of the queries.

We quantize the quality of expansion terms generated by each QE model. For the queries in the test split of TREC Deep Learning data and each held-out fold of Robust 04 (title query), we select the top 1000 frequent terms from the top ranked 100 documents retrieved by BM25, and compute how much performance improvement is caused by each of the words as an expansion term. We select the top 100 terms that cause the largest improvements as the ground truth, and compare them against the top 100 expansion terms generated by the QE models. We measure the quality of the expansion terms by the recall rate of the ground truth terms, and report the result in

Table 4. As shown in the result, the recall rates of QECVAE and the BERT-QECVAE variant are at least 5%, 15%, respectively, higher than that of RM3. The combinations of the proposed models and RM3 have higher recall rates than the proposed models working alone. This further shows the expansion terms generated by the proposed models and RM3 are complementary to each other.

Table 4: Average rates of the ground truth terms retrieved by the QE models

Model	TREC2019	Robust 04
AWE	0.19	0.15
RM3	0.32	0.24
QECVAE	0.37	0.30
QECVAE-RM3	0.40	0.33
BERT-QECVAE	0.46	0.40
BERT-QECVAE-RM3	0.48	0.41

Table 5: The top 10 expansion terms produced by the QE models for 4 randomly selected queries from TREC 2019: 'what party is paul ryan in' (Q1), 'cast of sky captain and the world of tomorrow '(Q2), 'what origin is the last name goins' (Q3), 'how long is recovery from a face lift and neck...' (Q4)

,

Query	RM3	QECVAE	BERT-QECVAE
Q1	ryan, paul, trump said, house gop, republican people, speaker, party	republican ,democratic party, president presidential, vote candidate , washington wisconsin election	republican, house presidential, speaker washington, congress party , senate representative, campaign
Q2	tomorrow, captain film , world sky, legends, 1 new, time	movie , show , actor original , crew award, america series, star , july	film , movie , character hollywood , cast star, actor release , crew ,actress
Q3	familyname, lastname origin ,meaning country , surname family, goin, goins come	country , germany ,name surname , french, old origin , harry england , history	history , surname , name origin, french , roman king , church century, christian
Q4	lift , neck , face surgery , skin , procedure recovery, surgeon patients, plastic	lift , surgery , recovery surgeon, week plastic , time , pain look , doctor	pain, muscle , week surgical , swell , treatment heal , therapy surgeon , month

Table 6: The top 10 expansion terms produced by the QE models for 4 randomly selected queries from Robust 04: 'international criminal activity' (Q1), 'the best living conditions and quality of life for a u.s. '(Q2), 'How do computers get infected by computer viruses?' (Q3), 'What are the advantages and/or disadvantages of tooth implants?.' (Q4)

Query	RM3	QECVAE	BERT-QECVAE
Q1	crime, organized federation, internal criminal , affairs, ministry fight, law,organs	country ,government police, crime law-enforcement european, russia united-states , woman, police	child, drug international, crime law-enforcement government , european county, russia united-states
Q2	retiree, retirement national , u.s money, medical health, living life quality	u.s. , cost food , security retiree ,health medical ,government american , living	retiree , u.s. , living health-care , insurance tax, financial benefits , pension,cost
Q3	computer, virus program ,software security , information system, user personal , data	network , virus security , program , risk, system, pc , computers infected , office	network , infected computer , pc, virus program ,security information ,data, software
Q4	implant , silicone dental, corning , fda , travel, patient, advantage plastic, study	dental , dentist , fda implant, artificial material , health infection ,safety , pound	dental, implant infection, fda , health dentist , jawbone , damage surgery , allergy

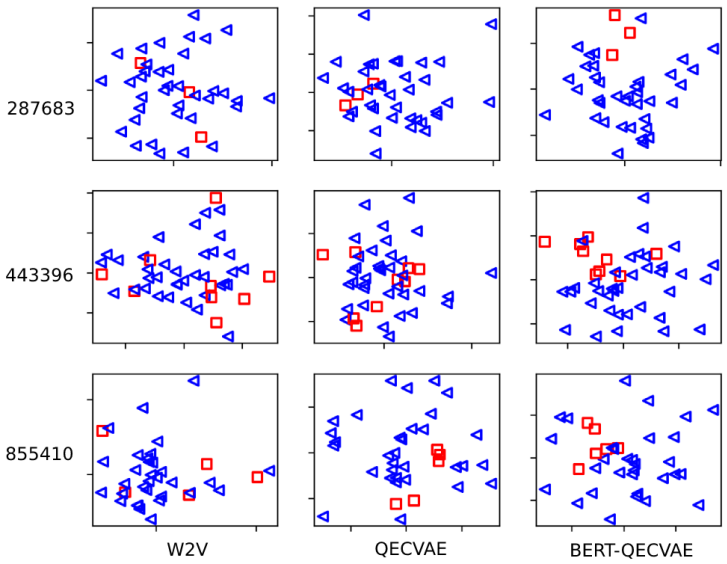

Figure 4: Visualization of encoding vectors of relevant and irrelevant documents. △ denotes the irrelevant query-document pair, □ the relevant pair.

The encoders of QECVAE and the BERT-QECVAE variant are designed to encode query-document relevance. We qualitatively analyze the behavior of its encoding vectors produced by the proposed models. We randomly select 3 queries (287683, 443396, 855410) and their PRF documents from the relevance label file of the test split of TREC 2019. We first represent the documents with AWE and visualize them after dimension reduction by PCA in the left-most panel of Figure 4 . As shown in the figure, the projections of relevant documents are spread out among that of irrelevant documents. This is obviously caused by that the model ignores the relevance between queries and documents.

At the meantime, we pair each query with their respective documents and feed them to the encoder of QECVAE and the BERT-QECVAE variant. We sample values for the latent variables z from the resulting variational distributions $Q(z|d, q)$ and visualize them in the middle and right-most panels of Figure 4 . As the figure indicates, the encodings of relevant query-document pairs are much more locally distributed, whereas that of irrelevant pairs are spread out across whole space. This shows the effectiveness

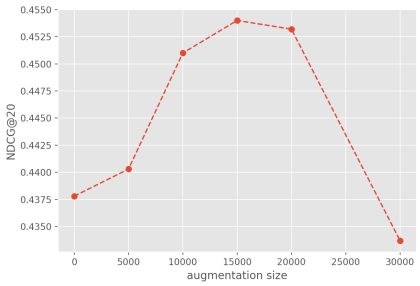

Figure 5: The performance of QECVAE on Robust 04 augmented with different number of samples from TREC 2019

of the proposed models in encoding the relevance.

9. Discussion

In the experiment we augmented Robust 04 with RF data from TREC 2019 to train QECVAE. We compared the performance of QECVAE on the dataset with different augmentation sizes and show the results in Figure 5 . As shown in the figure, the NDCG@20 of QECVAE on the dataset augmented with 15,000 samples is around 1.6% higher than that on the original dataset. Though we only tested this approach on Robust 04, we deem that it is possible to apply the approach on other datasets. However, there may exist an upper limit for the size. In the experiment we found that a size larger than 15,000 yields no further improvement. When the size reaches 30,000, the performance is worse than that on the original dataset.

Compared with RM3 and other PRF based methods that do not use RF data to generate expansion terms, the proposed models are effective to capture user search intentions when queries are ambiguous, with knowledge learned from the RF data. For example, on the Robust 04 dataset, RM3 produces 'product, phone, security, 2012, iphone, service, apple, store, ipad, juice' as the top 10 expansion terms for the query 'apple product'. This is caused by that some of the PRF documents for the query are related to the apple fruit, though the majority of them are related to the Apple company. The existence of 'fruit' may cause the retrieval model to retrieve irrelevant documents. By contrast, BERT-QECVAE produces 'apple, iphone, ipad, computer, consumer, electronics, digital, device, ipod, product' as the expansion terms.

This work is closely related to the relevance embedding model [10]. However, this work differs from it in at least the following 2 ways. First, the focuses are different. The relevance embedding model focuses on learning good representations of words, by taking into account the relevance. Our work focuses on exploiting the relevance for directly generating expansion terms. Second, the working methodologies of generating expansion terms are different. Similar as other embedding based models, the relevance embedding model selects expansion terms by their semantic similarities to original query terms. Our work learns a generative model that can directly predict effective expansion terms based on the interactions between queries and PRD documents.

One criticism we are expecting is that we do not compare the proposed model with the recent BERT-QE model [27]. We acknowledge that the performance reported by the authors are higher than that of BERT-QECVAE. However, BERT-QE involves 3 rounds of BERT-based reranking, therefore, it is much more computationally expensive and time consuming than the proposed models. In a realistic ad-hoc information retrieval scenario, low latency is a critical factor. Also, BRRT-QE is similar as Neural-PRF, it actually does not generate expansion terms. Instead, it expands original queries with PRF document chunks and is focused on approximating the relevance of a candidate document to the chunks. It belongs more to the category of neural matching models, whose predictions results are usually less explainable than that of term-based QE models.

10. Conclusion

In this research we treat QE as a generative problem and assume the relevant documents of a query are generated from language models depending on the query. Based on the assumption, we propose a novel QE model based on CVAE, and a BERT variant of it, to learn the language models for query expansion. We evaluate the proposed models on the Robust 04 and TREC 2019 Deep Learning datasets. Experiment results suggest that, the proposed models, especially the BERT-variant, outperform a number of robust traditional and neural QE baselines. We also find that combining the proposed models with RM3 leads to further improvements over the baselines.

It is also possible to combine the proposed models with state-of-the-art BERT based neural matching models. We are interested in examining the additivity between them and will conduct a compressive investigation in the near future.

References

[1] C. Carpineto, G. Romano, A Survey of Automatic Query Expansion in Information Retrieval, ACM Computing Surveys 44 (1) (2012) 1–50. doi:10.1145/2071389.2071390.

[2] J. Rocchio, Relevance feedback in information retrieval, The Smart retrieval system-experiments in automatic document processing (1971) 313–323Publisher: Prentice Hall.

[3] V. Lavrenko, W. B. Croft, and W.B.Croft. Relevance-based language models, in: In Proceedings on the 24th annual international ACM SIGIR conference, 2001, pp. 120–127.

[4] W. Yang, K. Lu, P. Yang, J. Lin, Critically Examining the "Neural Hype": Weak Baselines and the Additivity of Effectiveness Gains from Neural Ranking Models, in: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'19, Association for Computing Machinery, New York, NY, USA, 2019, pp. 1129–1132. doi:10.1145/3331184.3331340.

[5] S. Liu, F. Liu, C. Yu, W. Meng, An effective approach to document retrieval via utilizing WordNet and recognizing phrases, in: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval, 2004, pp. 266–272.

[6] Z. Gong, C. W. Cheang, Multi-term web query expansion using WordNet, in: International Conference on Database and Expert Systems Applications, Springer, 2006, pp. 379–388.

[7] D. Pal, M. Mitra, K. Datta, Improving query expansion using WordNet, Journal of the Association for Information Science and Technology 65 (12) (2014) 2469–2478, publisher: Wiley Online Library.

[8] A. Kotov, C. Zhai, Tapping into knowledge base for concept feedback: leveraging conceptnet to improve search results for difficult queries, in: Proceedings of the fifth ACM international conference on Web search and data mining, 2012, pp. 403–412.

[9] S. Kuzi, A. Shtok, O. Kurland, Query Expansion Using Word Embeddings, in: Proceedings of the 25th ACM International on Conference on Information and Knowledge Management - CIKM '16, ACM Press, Indianapolis, Indiana, USA, 2016, pp. 1929–1932. doi:10.1145/2983323.2983876.

[10] H. Zamani, W. B. Croft, Relevance-based Word Embedding, in: Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '17, Association for Computing Machinery, New York, NY, USA, 2017, pp. 505–514. doi:10.1145/3077136.3080831.

[11] D. Roy, D. Paul, M. Mitra, U. Garain, Using Word Embeddings for Automatic Query Expansion, arXiv:1606.07608 [cs]ArXiv: 1606.07608 (Jun. 2016).

[12] H. K. Azad, A. Deepak, Query expansion techniques for information retrieval: A survey, Information Processing & Management 56 (5) (2019) 1698–1735. doi:10.1016/j.ipm.2019.05.009.

[13] J. Guo, Y. Fan, Q. Ai, W. B. Croft, A Deep Relevance Matching Model for Ad-hoc Retrieval, in: Proceedings of the 25th ACM International on Conference on Information and Knowledge Management, CIKM '16, Association for Computing Machinery, New York, NY, USA, 2016, pp. 55–64. doi:10.1145/2983323.2983769.

[14] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational

Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186. doi:10.18653/v1/N19-1423.

[15] L. Wang, Z. Luo, C. Li, B. He, L. Sun, H. Yu, Y. Sun, An end-to-end pseudo relevance feedback framework for neural document retrieval, Information Processing & Management 57 (2) (2020) 102182. doi:10.1016/j.ipm.2019.102182.

[16] K. Goslin, M. Hofmann, A Wikipedia powered state-based approach to automatic search query enhancement, Information Processing & Management 54 (4) (2018) 726–739. doi:10.1016/j.ipm.2017.10.001.

[17] H. K. Azad, A. Deepak, A new approach for query expansion using Wikipedia and WordNet, Information Sciences 492 (2019) 147–163. doi:10.1016/j.ins.2019.04.019.

[18] K. Church, P. Hanks, Word association norms, mutual information, and lexicography, Computational linguistics 16 (1) (1990) 22–29.

[19] C. C. Latiri, S. B. Yahia, J. P. Chevallet, A. Jaoua, Query expansion using fuzzy association rules between terms, Proceedings of JIM (2003).

[20] C. Carpineto, R. de Mori, G. Romano, B. Bigi, An information-theoretic approach to automatic query expansion, ACM Transactions on Information Systems 19 (1) (2001) 1–27. doi:10.1145/366836.366860.

[21] C. Zhai, J. Lafferty, Model-based Feedback in the Language Modeling Approach to Information Retrieval 8.

[22] J. Miao, J. X. Huang, Z. Ye, Proximity-based rocchio's model for pseudo relevance, in: Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval - SIGIR '12, ACM Press, Portland, Oregon, USA, 2012, p. 535. doi:10.1145/2348283.2348356.

[23] J. A. Nasir, I. Varlamis, S. Ishfaq, A knowledge-based semantic framework for query expansion, Information Processing & Management 56 (5) (2019) 1605–1617. doi:10.1016/j.ipm.2019.04.007.

[24] J. Camacho-Collados, M. T. Pilehvar, From Word To Sense Embeddings: A Survey on Vector Representations of Meaning, Journal of Artificial Intelligence Research 63 (2018) 743–788. doi:10.1613/jair.1.11259.

[25] F. Diaz, B. Mitra, N. Craswell, Query Expansion with Locally-Trained Word Embeddings, in: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Berlin, Germany, 2016, pp. 367–377. doi:10.18653/v1/P16-1035.

[26] J. Wang, M. Pan, T. He, X. Huang, X. Wang, X. Tu, A Pseudo-relevance feedback framework combining relevance matching and semantic matching for information retrieval, Information Processing & Management 57 (6) (2020) 102342. doi:10.1016/j.ipm.2020.102342.

[27] Z. Zheng, K. Hui, B. He, X. Han, L. Sun, A. Yates, BERT-QE: Contextualized Query Expansion for Document Re-ranking, in: Findings of the Association for Computational Linguistics: EMNLP 2020, Association for Computational Linguistics, Online, 2020, pp. 4718–4728. doi:10.18653/v1/2020.findings-emnlp.424.

[28] A. Imani, A. Vakili, A. Montazer, A. Shakery, Deep Neural Networks for Query Expansion Using Word Embeddings, in: L. Azzopardi, B. Stein, N. Fuhr, P. Mayr, C. Hauff, D. Hiemstra (Eds.), Advances in Information Retrieval, Lecture Notes in Computer Science, Springer International Publishing, Cham, 2019, pp. 203–210. doi:10.1007/978-3-030-15719-7_26.

[29] J. Liu, S. Kim, V. Murali, S. Chaudhuri, S. Chandra, Neural query expansion for code search, in: Proceedings of the 3rd acm sigplan international workshop on machine learning and programming languages, 2019, pp. 29–37.

[30] R. Nogueira, W. Yang, J. Lin, K. Cho, Document Expansion by Query Prediction, arXiv e-prints 1904 (2019) arXiv:1904.08375.

[31] P.-S. Huang, X. He, J. Gao, L. Deng, A. Acero, L. Heck, Learning deep structured semantic models for web search using clickthrough data, in: Proceedings of the 22nd ACM international conference on Information & Knowledge Management, 2013, pp. 2333–2338.

[32] H. Palangi, L. Deng, Y. Shen, J. Gao, X. He, J. Chen, X. Song, R. Ward, Deep sentence embedding using long short-term memory networks: Analysis and application to information retrieval, IEEE/ACM Transactions on Audio, Speech, and Language Processing 24 (4) (2016) 694–707, publisher: IEEE.

[33] Y. Shen, X. He, J. Gao, L. Deng, G. Mesnil, Learning semantic representations using convolutional neural networks for web search, in: Proceedings of the 23rd international conference on world wide web, 2014, pp. 373–374.

[34] K. Sohn, H. Lee, X. Yan, Learning structured output representation using deep conditional generative models, in: Advances in neural information processing systems, 2015, pp. 3483–3491.

[35] B. Mitra, F. Diaz, N. Craswell, Learning to Match using Local and Distributed Representations of Text for Web Search, in: Proceedings of the 26th International Conference on World Wide Web - WWW '17, ACM Press, Perth, Australia, 2017, pp. 1291–1299. doi:10.1145/3038912.3052579.

[36] R. Nogueira, K. Cho, Passage Re-ranking with BERT, arXiv preprint arXiv:1901.04085 (2019).

[37] Z. Akkalyoncu Yilmaz, W. Yang, H. Zhang, J. Lin, Cross-Domain Modeling of Sentence-Level Evidence for Document Retrieval, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, China, 2019, pp. 3490–3496. doi:10.18653/v1/D19-1352.

[38] C. Li, A. Yates, S. MacAvaney, B. He, Y. Sun, PARADE: Passage Representation Aggregation for Document Reranking, arXiv:2008.09093 [cs]ArXiv: 2008.09093 (Aug. 2020).

[39] S. Huston, W. B. Croft, Parameters learned in the comparison of retrieval models using term dependencies, Ir, University of MassachusettsPublisher: Citeseer (2014).

[40] C. Zhai, Statistical language models for information retrieval, Synthesis lectures on human language technologies 1 (1) (2008) 1–141, publisher: Morgan & Claypool Publishers.

[41] S. E. Robertson, S. Walker, M. Beaulieu, P. Willett, Okapi at TREC-7: automatic ad hoc, filtering, VLC and interactive track, Nist Special Publication SP (500) (1999) 253–264, publisher: NATIONAL INSTIUTE OF STANDARDS & TECHNOLOGY.

[42] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, in: Advances in neural information processing systems, 2013, pp. 3111–3119.