# Embedding-based Query Expansion for Weighted Sequential Dependence Retrieval Model

Saeid Balaneshin-kordan
Wayne State University
saeid@wayne.edu

Alexander Kotov
Wayne State University
kotov@wayne.edu

## ABSTRACT

Although information retrieval models based on Markov Random Fields (MRF), such as Sequential Dependence Model and Weighted Sequential Dependence Model (WSDM), have been shown to outperform bag-of-words probabilistic and language modeling retrieval models by taking into account term dependencies, it is not known how to effectively account for term dependencies in query expansion methods based on pseudo-relevance feedback (PRF) for retrieval models of this type. In this paper, we propose Semantic Weighted Dependence Model (SWDM), a PRF based query expansion method for WSDM, which utilizes distributed low-dimensional word representations (i.e., word embeddings). Our method finds the closest unigrams to each query term in the embedding space and top retrieved documents and directly incorporates them into the retrieval function of WSDM. Experiments on TREC datasets indicate statistically significant improvement of SWDM over state-of-the-art MRF retrieval models, PRF methods for MRF retrieval models and embedding based query expansion methods for bag-of-words retrieval models.

## CCS CONCEPTS

•**Information systems →Query reformulation;**

## KEYWORDS

Weighted Sequential Dependence Model; Term Dependencies; Word Embeddings; Query Expansion; Pseudo-Relevance Feedback

## 1 INTRODUCTION

Designing retrieval models and addressing the problem of vocabulary mismatch via query and document expansion have traditionally been two orthogonal directions of information retrieval (IR) research. In particular, separate methods for query [4] or document [3] expansion are typically employed in conjunction with bag-of-words probabilistic, such as BM25 [15], and language modeling based, such as Query Likelihood [14], retrieval models. These methods typically identify query expansion terms in the collection itself using statistical measures of semantic similarity between pairs of terms pre-computed in advance [2, 9], top-retrieved documents [11], external resources [10] or their combination [1].


*SIGIRfi17, August 7–11, 2017, Shinjuku, Tokyo, Japan*

DOI: http://dx.doi.org/10.1145/3077136.3080764

Markov Random Fields (MRF) retrieval models [12], such as Sequential Dependence Model (SDM) and Weighted Sequential Dependence Model (WSDM), consider retrieval function as a graph of dependencies between the query terms and a document and calculate document retrieval score as a linear combination of the potential functions defined on the cliques in this graph. Although these retrieval models have been shown to outperform probabilistic and language modeling retrieval models by going beyond bag-of-words assumption and taking into account term dependencies, it is not known how to effectively incorporate term dependencies into query expansion methods based on pseudo-relevance feedback (PRF) for this type of retrieval models. Due to sparsity of $n$-grams, accounting for term dependencies in query expansion based only on term co-occurrence statistics within the collection itself is quite challenging. For this reason, only unigrams have been utilized for query expansion in Latent Concept Expansion (LCE) [13] and Parameterized Query Expansion (PQE) [6], state-of-the-art PRF methods for SDM and WSDM, respectively.

Word embeddings are distributed low-dimensional vector representations, which have been successfully utilized in different IR contexts, such as estimation of translation models [3, 21] and query expansion for bag-of-words retrieval models [19], as well as in retrieval models based on neural architectures [7, 16] and proximity search [8]. Since $n$-grams typically appear in a limited number of contexts in a collection, the utility of $n$-gram embeddings for IR is limited. For example, based on the word embeddings trained on a Google News corpus with 100 billion words[1], the most semantically similar words to "*human*" are "*humankind*", "*mankind*" and "*humanity*", all of which can be good query expansion terms. The most semantically similar $n$-grams to "*human*", however, are "*human beings*", "*impertinent flamboyant endearingly*", "*employee Laura Althouse*", and "*Christine Gaugler head*"[2]. It is obvious that these $n$-grams would only cause topic drift and degrade the retrieval results, if used for query expansion. Furthermore, due to sparsity, bigram embeddings are also ineffective for computing their importance weight in SDM [20].

Our proposed model, **S**emantic **W**eighted **D**ependence **M**odel (SWDM), mitigates the potential vocabulary mismatch between queries and documents in WSDM via query expansion. Similar to SDM and WSDM, the retrieval score of a document, according to SWDM, depends on the matching query unigrams, ordered and unordered bigrams. However, unlike SDM, SWDM finds *the closest unigrams and bigrams to query terms in embedding space* and directly incorporates them into the retrieval function of WSDM.

[1] https://drive.google.com/file/d/0B7XkCwpI5KDYNlNUTTlSS21pQmM/

[2] The similarity of trigrams "*employee Laura Althouse*" and "*Christine Gaugler head*" was deduced from the fragments "*… said Christine Gaugler, head of human resources …*" and "*… and human resources employee Laura Althouse …*", which appear in multiple places within this corpus.

To overcome the $n$-gram sparsity problem, SWDM takes into account only the dependencies between those pairs of terms, in which at least one term is semantically similar to one of the query terms and both terms have appeared within multiple windows in the collection and top retrieved documents. For example, for the query "*human smuggling*", SWDM identifies "*trafficking*" as one of the expansion unigrams and "*human trafficking*" as one of the expansion bigrams, since the unigram "*trafficking*" is semantically similar to "*smuggling*". Proximity to query terms in the embedding space as well as their frequency in the collection and top-retrieved documents are used as features for weighting the importance of original and expansion concepts (unigrams or bigrams).

## 2 RELATED WORK

Word embeddings are typically utilized in retrieval models to calculate distributional similarity between terms, quantify relevance of documents to queries or as an input to neural architectures for relevance matching.

**Calculation of distributional similarity:** cosine similarity between word embeddings is used in two scenarios. In the first scenario, distributional similarity between word vectors is used to find semantically similar words to expand queries [19] or documents [3, 21]. Components of a difference vector between embeddings of query unigrams and embeddings of the entire query have been utilized as features to estimate the weights of query unigrams in SDM [20]. Relevance of documents to queries can be quantified by aggregating cosine similarity scores between pairs of the most similar query and document term embedding vectors [8]. An alternative approach to quantifying relevance of documents to queries involves aggregating the embeddings of all document and query words into a single embedding vector for the entire document and a single embedding vector for the entire query and calculating their relevance score as a cosine similarity between these vectors [17]. The proposed method also falls into this category, since it relies on cosine similarity between word embeddings to find the most semantically similar unigrams to query terms and uses these unigrams for query expansion.

**Input to neural architectures:** distributed representations of query and document terms have also been used as input to neural architectures based on Convolutional Neural Network [16] or Recurrent Neural Network [18], which directly calculate the relevance score of documents to queries. In [7], a histogram of cosine similarities between embeddings of a query and documents terms is used an input to a feed-forward neural network with term gating, which directly computes the relevance score.

## 3 METHOD

Retrieval function of SDM calculates the relevance score of document $D$ to query $Q$ as follows:

$$P(D|Q) \overset{rank}{=} \sum_{q_i \in Q} \lambda_T f_T(q_i, D) + \sum_{q_i, q_{i+1} \in Q} \lambda_B f_B(q_i q_{i+1}, D) + \sum_{q_i, q_{i+1} \in Q} \lambda_U f_U(q_i q_{i+1}, D) \quad (1)$$

where $q_i$ is a query unigram and $q_i q_{i+1}$ is a query bigram, and $f_T(q_i, D)$, $f_B(q_i q_{i+1}, D)$ and $f_U(q_i q_{i+1}, D)$ are the potential (i.e., matching) functions for query concept types (unigrams, ordered and unordered bigrams), respectively, and $\lambda_T$, $\lambda_B$ and $\lambda_U$ are the weights of these potential functions, which determine the relative importance of query concept types. The potential function $f_T(q_i, D)$ for unigrams is defined as:

$$f_T(q_i, D) = \frac{n(q_i, D) + \mu \frac{n(q_i, C)}{|C|}}{|D| + \mu} \quad (2)$$

where $n(q_i, D)$ and $n(q_i, C)$ are the counts of unigram $q_i$ in $D$ and collection $C$, $|D|$ and $|C|$ are the numbers of words in document and collection, and $\mu$ is the Dirichlet smoothing prior. $f_B(q_i q_{i+1}, D)$ and $f_U(q_i q_{i+1}, D)$ are obtained in a similar way by counting co-occurrences of $q_i$ and $q_{i+1}$ in $D$ and $C$ in sequential order or within a window of a given size.

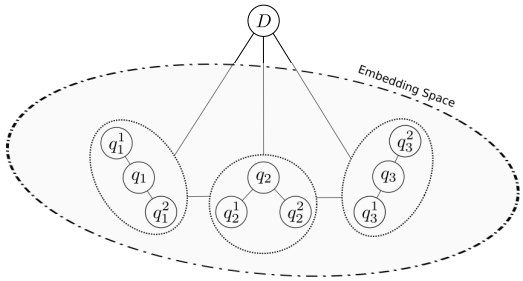


**Figure 1: Graphical representation of SWDM. $q_1$, $q_2$ and $q_3$ are the query terms and $D$ is a collection document. The words in dashed circles are the nearest neighbors of the query terms in the embedding space (only two most semantically similar words to each query term are shown for illustration).**

WSDM provides a more flexible parametrization of relevance than SDM by calculating the importance weight of each individual query concept rather than a concept type. The importance weight of each unigram and bigram is calculated as a linear combination of $k_u$ unigram feature functions $g_j^u(q_i)$ and $k_b$ bigram feature functions $g_j^b(q_i, q_{i+1})$ as follows:

$$P(D|Q) \overset{rank}{=} \sum_{q_i \in Q} \sum_{j=1}^{k_u} w_j^u g_j^u(q_i) f_T(q_i, D) + \sum_{q_i, q_{i+1} \in Q} \sum_{j=1}^{k_b} w_j^b g_j^b(q_i, q_{i+1}) f_B(q_i q_{i+1}, D) + \sum_{q_i, q_{i+1} \in Q} \sum_{j=1}^{k_b} w_j^b g_j^b(q_i, q_{i+1}) f_U(q_i q_{i+1}, D) \quad (3)$$

### 3.1 Semantic Weighted Dependence Model

In SWDM, the relevance score of $D$ to $Q$ also takes into account the words that are semantically similar to query terms in the embedding space. We define $q_i^j$ as the $j^{\text{th}}$ most similar term to $q_i$ in the embedding space, according to cosine similarity. We also define $\mathcal{E}_{q_i} = \{q_i^0, q_i^1, q_i^2, \cdots\}$ as a set of words, whose cosine similarity with $q_i$ in the embedding space exceeds a threshold $\tau_s$ (including $q_i^0 = q_i$ itself). Unlike SDM and WSDM, the potential functions

**Table 1: Performance of the proposed method with and without unigrams from the top retrieved documents and the baselines. 1 and 2 indicate statistically significant improvements of SWDM over WSDM and EQE1, respectively, while 3, 4 and 5 indicate statistically significant improvements of SWDM$^+$ over EQE1+RM1, PQE, and SWDM+RM1, respectively, according to the Fisher's randomization test ($p < 0.05$). Percentage improvements of SWDM over WSDM and EQE1 as well as percentage improvements of SWDM$^+$ over PQE and SWM+RM1 are shown in parenthesis.**

| Method | ROBUST04 | | GOV2 | | ClueWeb09B | |
|---|---|---|---|---|---|---|
| | MAP | P@10 | MAP | P@10 | MAP | P@10 |
| SDM | 0.2583 | 0.4278 | 0.3156 | 0.5457 | 0.0783 | 0.2777 |
| WSDM | 0.2689 | 0.4563 | 0.3232 | 0.5533 | 0.0762 | 0.2797 |
| EQE1 | 0.2597 | 0.4336 | 0.3172 | 0.5466 | 0.0742 | 0.2778 |
| SWDM | $0.2802^{1,2}$ (+4.20%/+7.89%) | $0.4676^{1,2}$ (+2.48%/+7.84%) | $0.3319^{1,2}$ (+2.69%/+4.63%) | $0.5598^{1,2}$ (+1.17%/+2.41%) | $0.0827^{1,2}$ (+8.53%/+11.46%) | $0.2812^{1,2}$ (+0.54%/+1.22%) |
| LCE | 0.2886 | 0.4697 | 0.3408 | 0.5667 | 0.0738 | 0.2693 |
| PQE | 0.2921 | 0.4726 | 0.3526 | 0.5858 | 0.0749 | 0.2751 |
| EQE1+RM1 | 0.2872 | 0.4672 | 0.3315 | 0.5459 | 0.0731 | 0.2695 |
| SWDM+RM1 | 0.2991 | 0.4828 | 0.3557 | 0.5872 | 0.0756 | 0.2716 |
| SWDM$^+$ | $0.3034^{3,4,5}$ (+3.87%/+1.44%) | $0.4909^{3,4,5}$ (+3.87%/+1.68%) | $0.3686^{3,4}$ (+4.54%/+3.63%) | $0.5997^{3,4}$ (+2.37%/+2.13%) | 0.0787 (+5.07%/+4.10%) | 0.2778 (+0.98%/+2.28%) |

$f_T(q_i, D)$, $f_B(q_i q_{i+1}, D)$ and $f_U(q_i q_{i+1}, D)$ in SWDM are calculated using all the terms in $\mathcal{E}$ and not just the query terms:

$$P(D|Q) \overset{rank}{=} \sum_{q_i^m \in \mathcal{E}_{q_i}} \sum_{j=1}^{k_u} w_j^u g_j^u(q_i^m) f_T(q_i^m, D) + \sum_{q_i^m, q_{i+1}^m \in \mathcal{E}_{q_i}, \mathcal{E}_{q_{i+1}}} \sum_{j=1}^{k_b} w_j^b g_j^b(q_i^m, q_{i+1}^m) f_B(q_i^m q_{i+1}^m, D) + \sum_{q_i^m, q_{i+1}^m \in \mathcal{E}_{q_i}, \mathcal{E}_{q_{i+1}}} \sum_{j=1}^{k_b} w_j^b g_j^b(q_i^m, q_{i+1}^m) f_U(q_i^m q_{i+1}^m, D) \quad (4)$$

Therefore, besides pair-wise dependencies between adjacent query terms, SWDM also captures pair-wise dependencies between query terms and the words semantically similar to them in the embedding space. For an example query in Figure 1, besides the query unigram $q_1$, retrieval score of $D$, according to SWDM, also includes the weighted matching scores for unigrams $q_1^1$ and $q_1^2$ that are semantically similar to $q_1$. Similarly, besides the query bigram $q_1 q_2$, SWDM also includes the weighted matching scores for: (1) the bigrams $q_1 q_2^1$, $q_1 q_2^2$, $q_1^1 q_2$, and $q_1^2 q_2$, which have only one of their constituent terms not from the original query (2) the bigrams $q_1^1 q_2^1$, $q_1^1 q_2^2$, $q_1^2 q_2^1$, and $q_1^2 q_2^2$, in which both constituent terms are not from the original query. If $\mathcal{E}_{q_i} = \{q_i^0\} = \{q_i\}$ (i.e., in the case when no semantically similar unigrams are in the neighborhood of original query terms), SWDM only takes into account the unigrams and bigrams in the original query (i.e. is identical to WSDM).

The importance weight of each query concept is computed based on several features. For an expansion unigram $q_i^j$, we use its cosine similarity with $q_i$, frequency in the collection and top retrieved documents, document frequency in the collection and top documents as features. For an expansion bigram $q_i^j q_{i+1}^j$, we use an average cosine similarity of the terms $q_i^j$ and $q_{i+1}^j$, sequential and window based co-occurrence frequency in the collection and top documents as well as sequential and window based document frequency in the collection and top documents.

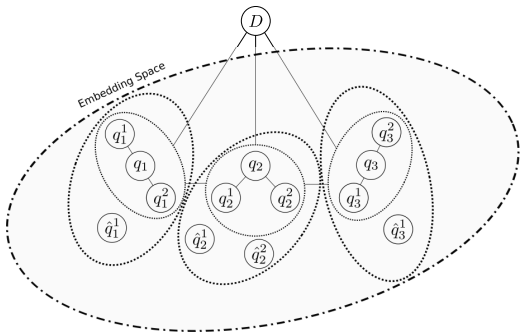


**Figure 2: Graphical representation of SWDM$^+$, a variant of SWDM, which besides the neighbors of the query terms in the embedding space ($q_1^1, \ldots, q_3^2$), also incorporates the unigrams from the top retrieved documents ranked by RM1 [11] ($\hat{q}_1^1, \ldots, q_3^1$).**

## 4 EXPERIMENTS

Experimental results reported below were obtained using word2vec word embeddings with 300 dimensions that were pre-trained on the Google News corpus[3], Indri-5.10 IR toolkit[4] and GenSim library[5] for all word embedding computations.

Retrieval accuracy of SWDM was evaluated with respect to Mean Average Precision (MAP) and Precision@10 (P@10) on ROBUST04, GOV2 and ClueWeb09B (excluding the documents for which Waterloo Fusion spam score was greater than 70) collections and compared with the state-of-the-art MRF retrieval models (SDM [12] and WSDM [5]), PRF methods for MRF retrieval models (LCE [13] and

[3] https://drive.google.com/file/d/0B7XkCwpI5KDYNlNUTTlSS21pQmM/
[4] https://www.lemurproject.org/indri/
[5] https://radimrehurek.com/gensim/

PQE [6]) and embedding based query expansion method for bag-of-words retrieval models (EEQ1 [19]). Collection and document frequencies were used as features to calculate the weights of query concepts in WSDM and PQE.

We also experimented with SWDM$^+$, a variant of SWDM, which, besides the neighbors of query terms in the embedding space, also incorporates $\hat{\mathcal{E}}_Q = \{\hat{q}^1, \hat{q}^2, \ldots, \hat{q}^k\}$, top $k$ unigrams from the top retrieved documents according to the relevance model (RM1) [11] scores, as illustrated in Figure 2. SWDM$^+$ uses the same set of features for weighting query concepts as SWDM. The similarity features to determine the importance of query concepts involving unigrams from top retrieved documents are calculated with respect to the closest query term in the embedding space. SWDM+RM1 and EEQ1+RM1 are a linear interpolation of RM1 with SWDM and EEQ1, respectively. Unigrams from the top retrieved documents in SWDM+RM1, EEQ1+RM1 and SWDM$^+$ are ranked using RM1 and the top $k$ are selected to be incorporated into the query.

The parameters of all models have been optimized using three-fold cross-validation and coordinate ascent to maximize MAP. The range of continuous and discrete model parameters has been examined with the step sizes of 0.02 and 1, respectively.

### 4.1 Results

Table 1 provides a summary of retrieval accuracy for SWDM, its variants and the baselines[6]. As follows from the first half of Table 1, SWDM outperforms SDM, WSDM and EQE1 in terms of both MAP and P@10, which indicates the utility of incorporating semantically related terms into WSDM. It also follows from Table 1 that the retrieval accuracy of EQE1 is close to that of SDM, which indicates that utilizing word embeddings for query expansion in conjunction with bag-of-words retrieval model results in similar improvements of retrieval accuracy as accounting for sequential dependencies between query terms. Our results also indicate that SWDM has better retrieval accuracy than EQE1, since besides incorporating similar words from the embedding space into a query, it also takes into account the dependencies between the expansion terms as well as between the expansion terms and the query terms.

The results in the second half of Table 1 indicate that incorporating unigrams from the top retrieved documents translates into a significant increase in retrieval accuracy of SWDM on ROBUST04 and GOV2 collections. In particular, SWDM$^+$ outperforms both LCE and PQE, state-of-the-art PRF methods for MRF retrieval models, which include a separate potential function for expansion terms, but do not take into account neither dependencies between the expansion terms nor between the expansion terms and the original query terms, and EQE1+RM1, which is designed for bag-of-words retrieval models. SWDM$^+$, however, has inferior performance to both SWDM and SWDM+RM1 on ClueWeb09B, which is due to relatively low accuracy of all retrieval models on this collection and, as a result, noisy unigrams from the top retrieved documents that are used for query expansion. This result suggests that the relative influence of query term neighbors and the expansion terms from the top retrieved documents on retrieval accuracy depends on a collection and the quality of the initial retrieval results. SWDM$^+$ also demonstrated a significantly statistical improvement in retrieval accuracy over SWDM+RM1 on ROBUST04 and GOV2 collections, indicating that the features based on similarity of expansion and the original query terms in the embedding space have a positive effect on retrieval accuracy.

[6]code and runs are available at http://github.com/teanalab/SWDM

## 5 CONCLUSION

In this paper, we proposed Semantic Weighted Dependence Model, which allows to address the vocabulary gap in Weighted Sequential Dependence Model, by leveraging distributed word representations (i.e. word embeddings) in two different ways. On one hand, word embeddings are used for calculating distributional similarity to find the terms that are semantically similar to query terms for query expansion. On the other hand, they are used as features to calculate the importance of query concepts. We also proposed an extension of SWDM, which besides semantically similar terms, also incorporates the terms from the top retrieved documents.

## REFERENCES


[1] Saeid Balaneshin-kordan and Alexander Kotov. 2016. Optimization method for weighting explicit and latent concepts in clinical decision support queries. In *Proceedings of ACM ICTIR*. 241–250.

[2] Saeid Balaneshin-kordan and Alexander Kotov. 2016. Sequential query expansion using concept graph. In *Proceedings of ACM CIKM*. 155–164.

[3] Saeid Balaneshin-kordan and Alexander Kotov. 2016. A study of document expansion using translation models and dimensionality reduction methods. In *Proceedings of ACM ICTIR*. 233–236.

[4] Saeid Balaneshinkordan and Alexander Kotov. 2016. An Empirical comparison of term association and knowledge graphs for query expansion. In *Proceedings of ECIR*. 761–767.

[5] Michael Bendersky, Donald Metzler, and W Bruce Croft. 2010. Learning concept importance using a weighted dependence model. In *Proceedings of ACM WSDM*. 31–40.

[6] Michael Bendersky, Donald Metzler, and W Bruce Croft. 2011. Parameterized concept weighting in verbose queries. In *Proceedings of ACM SIGIR*. 605–614.

[7] Jiafeng Guo, Yixing Fan, Qingyao Ai, and W Bruce Croft. 2016. A Deep Relevance Matching Model for Ad-hoc Retrieval. In *Proceedings of ACM CIKM*. 55–64.

[8] Jiafeng Guo, Yixing Fan, Qingyao Ai, and W Bruce Croft. 2016. Semantic Matching by Non-Linear Word Transportation for Information Retrieval. In *Proceedings of ACM CIKM*. 701–710.

[9] Alexander Kotov and ChengXiang Zhai. 2011. Interactive sense feedback for difficult queries. In *Proceedings of ACM CIKM*. 163–172.

[10] Alexander Kotov and ChengXiang Zhai. 2012. Tapping into knowledge base for concept feedback: leveraging conceptnet to improve search results for difficult queries. In *Proceedings of ACM WSDM*. 403–412.

[11] Victor Lavrenko and W Bruce Croft. 2001. Relevance based language models. In *Proceedings of ACM SIGIR*. 120–127.

[12] Donald Metzler and W Bruce Croft. 2005. A Markov random field model for term dependencies. In *Proceedings of ACM SIGIR*. 472–479.

[13] Donald Metzler and W Bruce Croft. 2007. Latent concept expansion using markov random fields. In *Proceedings of ACM SIGIR*. 311–318.

[14] Jay M Ponte and W Bruce Croft. 1998. A language modeling approach to information retrieval. In *Proceedings of ACM SIGIR*. 275–281.

[15] Stephen E Robertson, Steve Walker, Susan Jones, Micheline M Hancock-Beaulieu, Mike Gatford, and others. 1995. Okapi at TREC-3. *NIST* 109 (1995).

[16] Aliaksei Severyn and Alessandro Moschitti. 2015. Learning to rank short text pairs with convolutional deep neural networks. In *Proceedings of ACM SIGIR*. 373–382.

[17] Ivan Vulić and Marie-Francine Moens. 2015. Monolingual and cross-lingual information retrieval models based on (bilingual) word embeddings. In *Proceedings of ACM SIGIR*. 363–372.

[18] Di Wang and Eric Nyberg. 2015. A long short-term memory model for answer sentence selection in question answering. *Proceedings of ACL* (2015), 707–712.

[19] Hamed Zamani and W Bruce Croft. 2016. Embedding-based Query Language Models. In *Proceedings of ACM ICTIR*. 147–156.

[20] Guoqing Zheng and Jamie Callan. 2015. Learning to reweight terms with distributed representations. In *Proceedings of ACM SIGIR*. 575–584.

[21] Guido Zuccon, Bevan Koopman, Peter Bruza, and Leif Azzopardi. 2015. Integrating and evaluating neural word embeddings in information retrieval. In *Proceedings of ADCS*. 12.