

Query Expansion Using Word Embeddings

Saar Kuzi
saarkuzi@campus.technion.ac.il

Anna Shtok
annabel@tx.technion.ac.il

Oren Kurland
kurland@ie.technion.ac.il

Technion — Israel Institute of Technology

ABSTRACT

We present a suite of query expansion methods that are based on *word embeddings*. Using Word2Vec’s CBOW embedding approach, applied over the entire corpus on which search is performed, we select terms that are semantically related to the query. Our methods either use the terms to expand the original query or integrate them with the effective pseudo-feedback-based relevance model. In the former case, retrieval performance is significantly better than that of using only the query, and in the latter case the performance is significantly better than that of the relevance model.

1. INTRODUCTION

Query expansion approaches are intended to ameliorate the vocabulary mismatch between the short query and relevant documents [4]. We present query expansion methods that rely on global, query-independent, analysis of the corpus using *word embeddings*. Specifically, we use Word2Vec’s Continuous Bag-of-Words (CBOW) approach [11] that represents terms in a vector space based on their co-occurrence in windows of text. Similarities between the term vectors were shown to correspond to semantic similarities [11].

We use the Word2Vec term embeddings to select terms that are similar to the query as a whole or to its constituent terms. The selected terms are used to expand the query in a unigram language model framework. The resultant retrieval performance is significantly better than using the query alone, and a recently proposed translation model based on Word2Vec [21]. In addition, we present an approach to integrating the terms selected using word embeddings with an effective pseudo-feedback-based method: the relevance model [10]. The resultant performance significantly transcends that of the relevance model.

2. QUERY EXPANSION MODELS

Our goal is to expand a query q using semantically related terms. To that end, we train offline the Word2Vec Continuous Bag-of-Words (CBOW) model [11] using the entire doc-

ument corpus D on which search is performed. The model’s goal is to estimate the probability that a term will appear in a position in a text based on terms that appear in a window around this position. As a result, each term in the corpus is represented by a vector embedded in a vector space. Similarities between these vectors were shown to correspond to semantic similarities between the terms [11]. Accordingly, we select terms similar to the query in this vector space.

Our query expansion methods operate in a unigram language model framework. The methods use term scoring functions presented in Section 2.1. Section 2.2 presents a method of using the terms with the highest scores for query expansion. In Section 2.3 we present an approach for integrating the terms selected by our methods with a pseudo-feedback-based relevance model [10].

2.1 Term scoring methods

2.1.1 The centroid method

Adding Word2Vec vectors representing terms that constitute an expression often yields a vector that semantically corresponds to the expression [11]. We leverage this observation to score terms for expansion. Herein, \vec{t} denotes the L_2 -normalized Word2Vec vector representing term t .

We represent q by $\vec{q}_{Cent} \stackrel{def}{=} \sum_{q_i \in q} \vec{q}_i$, where q_i is a query term; this is the (query-length) scaled centroid of the query terms’ vectors. The score of term t in the collection is:

$$S_{Cent}(t; q) \stackrel{def}{=} \exp(\cos(\vec{t}, \vec{q}_{Cent})). \quad (1)$$

2.1.2 Fusion-based methods

The Cent approach scores terms by their semantic similarity with the query as a whole. As an alternative, we create for each query term, q_i , a list L_{q_i} of its n most similar terms t in the corpus according to $\cos(\vec{q}_i, \vec{t})$ (cf., [3, 13]); n is a free parameter. These n similarities are softmax-normalized to yield probabilities: $p(t|q_i) \stackrel{def}{=} \frac{\exp(\cos(\vec{q}_i, \vec{t}))}{\sum_{t' \in L_{q_i}} \exp(\cos(\vec{q}_i, \vec{t}'))}$ for

$t \in L_{q_i}$; $p(t|q_i) \stackrel{def}{=} 0$ for $t \notin L_{q_i}$. We fuse the resulting term lists using CombSUM, CombMNZ and CombMAX [7]:

$$S_{CombSUM}(t; q) \stackrel{def}{=} \sum_{q_i \in q} p(t|q_i); \quad (2)$$

$$S_{CombMNZ}(t; q) \stackrel{def}{=} |\{q_i \in q : p(t|q_i) \neq 0\}| \sum_{q_i \in q} p(t|q_i); \quad (3)$$

$$S_{CombMAX}(t; q) \stackrel{def}{=} \max_{q_i \in q} p(t|q_i). \quad (4)$$

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

CIKM’16, October 24–28, 2016, Indianapolis, IN, USA

© 2016 ACM. ISBN 978-1-4503-4073-1/16/10...\$15.00

DOI: <http://dx.doi.org/10.1145/2983323.2983876>

2.2 Expanding the query

The ν terms assigned the highest score by a method \mathcal{M} (\in {Cent, CombSUM, CombMNZ, CombMAX}) are used for query expansion; ν is a free parameter. We sum-normalize these terms' scores, $S_{\mathcal{M}}(t; q)$, to obtain a probability distribution, $p(\cdot|\mathcal{M})$, that constitutes a unigram language model defined over the vocabulary. (Terms not among the ν with the highest scores are assigned zero probability.) We integrate this language model with a language model induced from q . Formally, the maximum likelihood estimate (MLE) of term t with respect to q is: $p_{MLE}(t|q) \stackrel{def}{=} \frac{tf(t \in q)}{|q|}$, where $tf(t \in q)$ is the count of t in q and $|q|$ is the query length. The resulting model, $\mathbf{Q}\text{-}\mathcal{M}$, uses the interpolation parameter λ :

$$p(t|\mathcal{M}, q) \stackrel{def}{=} (1 - \lambda)p(t|\mathcal{M}) + \lambda p_{MLE}(t|q). \quad (5)$$

2.3 Integration with the relevance model

Our premise is that pseudo-feedback-based query expansion approaches are of complementary nature to our global expansion methods that use Word2Vec for corpus analysis. Hence, we now turn to integrate an effective pseudo feedback approach, the relevance model [10], with our methods.

The probability assigned to term t by RM1 is [10]:

$$p(t|RM1) \stackrel{def}{=} \sum_{d \in \mathcal{D}_{init}} p(t|d)p(d|q); \quad (6)$$

\mathcal{D}_{init} is an initial document list described below; $p(t|d)$ is the probability assigned to t by d 's Dirichlet smoothed language model [19]; $p(d|q)$ is d 's normalized query likelihood.

Clipping RM1 by setting to zero the probability of all but the c terms that yield the highest $p(t|RM1)$, and sum-normalizing their probabilities, yields $p_{clip}(t|RM1)$ [1]. Similarly, we use the c terms assigned the highest $S_{\mathcal{M}}(t; q)$, where $\mathcal{M} \in$ {Cent, CombSUM, CombMNZ, CombMAX}, to induce the language model $p(\cdot|\mathcal{M})$ as described above. The language models are integrated using a parameter α :

$$p(t|RM, \mathcal{M}) \stackrel{def}{=} \alpha p(t|\mathcal{M}) + (1 - \alpha)p_{clip}(t|RM1). \quad (7)$$

We clip this language model to use the ν terms (ν is a free parameter) assigned the highest $p(t|RM, \mathcal{M})$, yielding $p_{clip}(\cdot|RM, \mathcal{M})$ which is integrated with q :

$$p(t|RM, \mathcal{M}, q) \stackrel{def}{=} (1 - \lambda)p_{clip}(t|RM, \mathcal{M}) + \lambda p_{MLE}(t|q). \quad (8)$$

These methods are denoted $\mathbf{RM}\text{-}\mathcal{M}$ where $\mathcal{M} \in$ {Cent, CombSUM, CombMNZ, CombMAX}. If α is set to 0 in Eq. 7, i.e., Word2Vec-based expansion is not used, then Eq. 8 is the relevance model RM3 [1].

2.4 Document ranking

Documents are ranked by the cross entropy between a query model and their Dirichlet smoothed language models [19]. We use a few query language models. Using $p_{MLE}(\cdot|q)$ results in an initial (standard language-model-based) ranking (denoted **init**) from which the result list \mathcal{D}_{init} , used for relevance model construction, is derived. We also use the (query) language models from Eq. 5 and 8, $\mathbf{Q}\text{-}\mathcal{M}$ and $\mathbf{RM}\text{-}\mathcal{M}$, respectively. Relevance model RM3 [1], which is a special case of Equation 8 as noted above, is used for reference.

3. EXPERIMENTAL SETUP

The TREC datasets specified in Table 1 were used for experiments. Titles of TREC topics serve for queries. Krovetz stemming and stopword removal (using the INQUERY list)

Table 1: TREC datasets used for experiments.

Collection	TREC disks	# of Docs	Topics
WSJ	Disks 1&2	173,252	51-200
AP	Disks 1&3	242,918	51-200
ROBUST	Disks 4,5-CR	528,155	301-450, 601-700
WT10G	WT10g	1,692,096	451-550
GOV2	GOV2	25,205,179	701-850

were applied to documents and queries. The Indri toolkit (www.lemurproject.org) was used for experiments.

MAP (@1000), precision of the top five documents (p@5), and the reliability of improvement (RI) [14] are used as evaluation measures. RI is: $\frac{|Q_+| - |Q_-|}{|Q|}$; Q is the set of queries; Q_+ and Q_- are the sets queries for which the average precision (AP) is higher and lower, respectively, than that of the initial ranking. RI is a measure of the robustness of a query expansion method. Statistically significant differences of performance are determined using the two-tailed paired t-test at a 95% confidence level.

We train the CBOW model of Word2Vec¹ using all documents in each corpus, except for GOV2. For GOV2, due to vocabulary size scale issues, we used the Word2Vec models trained over WT10G². Documents were processed as described above except that terms longer than 50 characters were excluded due to Word2Vec limitations. Word2Vec incorporates several free parameters: the dimension of the term vectors was set to values in {100, 500}, the number of negative examples is in {5, 10}, and the window size is in {8, 16, 64}; sub-sampling of frequent terms was not performed. All other parameters were set to default values.

The Dirichlet smoothing parameter for document language models is set to 1000 unless otherwise specified. The query weight λ in *all* expansion models is in {0, 0.2, ..., 1}. The number of terms, ν , used by *all* query expansion models, is in {10, 25}. The number of terms similar to a single query term, n (see Section 2.1.2), is selected from {50, 100}. For the methods described in Section 2.2, the values of the aforementioned free parameters (including those of Word2Vec) are set using leave-one-out cross validation performed over queries, where MAP serves as the optimization criterion.

The free parameters of the $\mathbf{RM}\text{-}\mathcal{M}$ models (Section 2.3) are set as follows. First, we set the parameters of RM1 per value of c (\in {50, 100}) terms. The other parameters of RM1 are set using leave-one-out cross validation where the maximization criterion is the resultant RM3's MAP: (1) the Dirichlet smoothing parameter used for document language models for constructing RM1 is in {0, 1000}; (2) the number of documents in \mathcal{D}_{init} from which RM1 is constructed is in {10, 25, 50}; (3) the original query weight λ (\in {0, 0.2, ..., 1}). The values of all remaining parameters are set using leave-one-out cross validation; here, the MAP of the resultant query expansion method is optimized. The parameters are those of the CBOW model, α (\in {0, 0.2, ..., 1}), λ (\in {0, 0.2, ..., 1}), n and c ; the latter two are coupled to the same values in the $\mathbf{RM}\text{-}\mathbf{Comb}$ methods.

For RM3, which serves as a baseline, the ranges of the number of expansion terms, the query weight λ , the Dirich-

¹www.code.google.com/p/word2vec

²Applying Word2Vec models trained using one corpus on another corpus was shown to be effective for query expansion [17] and translation models [21].

Table 2: Expanding the original query. The best and second best results in each column are boldfaced. The best result in a column is underlined. Statistically significant differences with the initial ranking (init), RM3, NTLM, Q-Cent, Q-CombSUM and Q-CombMNZ are marked with 'i', 'r', 't', 'c', 's' and 'm', respectively.

	WSJ			AP			ROBUST			WT10G			GOV2		
	MAP	p@5	RI	MAP	p@5	RI	MAP	p@5	RI	MAP	p@5	RI	MAP	p@5	RI
init	.272	.489	—	.216	.413	—	.249	.476	—	.198	.355	—	.299	.588	—
RM3	<u>.324ⁱ</u>	.520 ^t	.355	<u>.274ⁱ</u>	.435	.443	<u>.282ⁱ</u>	.483	.293	<u>.234ⁱ</u>	.390	.082	<u>.331ⁱ</u>	.596	.392
NTLM	.277 ^r	.504	-.073	.226 ^{i,r}	.442	.094	.241 ^r	.481	-.241	.201 ^r	.344	-.031	.280 ^{i,r}	.574	-.311
Q-Cent	.294 ^{i,r,t}	.519 ^t	.187	.243^{i,r,t}	.432	.289	.256^{i,r,t}	.488	.241	.201 ^r	.348	.052	.300 ^{r,t}	.622^{i,t}	-.041
Q-CombSUM	.295^{i,r,t}	.520 ^t	.227	.235 ^{i,r}	.447 ^t	.221	.253 ^{r,t}	.487	.137	.205^r	.359	.134	.301 ^{r,t}	.615 ^{i,t}	-.014
Q-CombMNZ	.291 ^{i,r,t}	.524ⁱ	.273	.234 ^{i,r}	.451ⁱ	.383	.252 ^{i,r,t}	.476	.205	.203 ^r	.353	.155	.297 ^{s,t}	.616 ^{i,t}	-.095
Q-CombMAX	.294 ^{i,r,t}	.521ⁱ	.313	.237 ^{i,r,t}	.448ⁱ	.282	.253 ^{i,r,t}	.491ⁱ	.237	.203 ^r	.351	.124	.305^{i,r,t}	.631^{i,r,t}	.074

Table 3: Integration with the relevance model. The best result in a column is boldfaced. Statistically significant differences with the initial ranking (init), RM3, RM-Cent, RM-CombSUM and RM-CombMNZ are marked with 'i', 'r', 'c', 's' and 'm', respectively.

	WSJ			AP			ROBUST			WT10G			GOV2		
	MAP	p@5	RI	MAP	p@5	RI	MAP	p@5	RI	MAP	p@5	RI	MAP	p@5	RI
init	.272	.489	—	.216	.413	—	.249	.476	—	.198	.355	—	.299	.588	—
RM3	.324 ^t	.520 ^t	.355	.274 ^t	.435	.443	.282 ^t	.483	.293	.234 ^t	.390	.082	.331 ^t	.596	.432
RM-Cent	.332^{i,r}	.548^{i,r}	.387	.282^{i,r}	.464^{i,r}	.416	.291^{i,r}	.495	.261	.240^{i,r}	.396	.155	.335 ^{i,r}	.623^r	.392
RM-CombSUM	.325 ^t	.533 ^t	.320	.279 ^{i,r}	.447	.483	.291^{i,r}	.492	.245	.236 ^t	.402ⁱ	.113	.335 ^{i,r}	.618 ^r	.405
RM-CombMNZ	.318 ^{i,r,s}	.523 ^t	.247	.278 ^{i,r}	.432 _c	.456	.289 ^{i,r}	.483	.249	.231 ^t	.396 ^t	.093	.335 ^{i,r}	.614 ^r	.432
RM-CombMAX	.330 ^{i,r,s,m}	.541 ^{i,r}	.373	.278 ^t	.439 _c	.483	.290 ^{i,r}	.496_m	.245	.237 ^t	.402ⁱ	.113	.336^{i,r}	.622 ^r	.297

let smoothing parameter used to construct the model, and the number of documents in $\mathcal{D}_{\text{init}}$ from which RM3 is constructed, are as specified above for the RM- \mathcal{M} models. Leave-one-out cross validation was applied to set parameter values.

The NTLM translation model serves as an additional baseline [21]. The 10 terms most similar to each query term, based on Word2Vec, were considered as translation terms. Translation probabilities were computed using sum-normalized cosine similarity between a query term and a translation term. The top 1000 documents in the initial ranking were re-ranked using NTLM as we found that ranking all documents in the corpus that contain at least one query term often resulted in inferior performance in four out of five corpora. Following the source code of [21], very frequent or very rare query terms were assigned a 1.0 translation probability³. The free parameters of NTLM, namely those of Word2Vec, were set using leave-one-out cross validation.

4. EXPERIMENTAL RESULTS

4.1 Expanding the query

Table 2 presents the performance of the Q- \mathcal{M} methods from Section 2.2; these expand the original query with semantically related terms. All these methods outperform, in terms of MAP and p@5, both the initial retrieval and NTLM in the vast majority of cases; the majority of MAP improvements are statistically significant; the Q- \mathcal{M} methods are also more robust (RI) than NTLM. In all corpora except for WT10G, the Q- \mathcal{M} methods often outperform RM3 in terms of p@5, however rarely statistically significantly. Although RM3 is the best MAP performing expansion method in most

³The exact criteria for frequency-based term filtering is provided in the source code of [21]; applying the method without this filter or applying the filter on translation terms as well resulted in inferior retrieval effectiveness.

cases, the Q- \mathcal{M} methods yield effective performance and do not require initial retrieval as RM3.

We also see in Table 2 that no Q- \mathcal{M} method dominates the others. The MAP and p@5 of Q-Cent, which performs expansion based on the query as a whole, is almost always statistically indistinguishable from that of the Q-Comb models that use expansions based on individual query terms.

Table 2 shows that NTLM, which uses Word2Vec-based translation probabilities, is almost always outperformed by RM3. Therefore, below we focus on RM3 as a baseline.

4.2 Integration with the relevance model

Table 3 presents the performance of the RM- \mathcal{M} methods from Section 2.3 which integrate the Word2Vec-based expansion terms with the relevance model. RM-Cent, which integrates terms semantically similar to the query as a whole with the relevance model, performs best in most cases. RM-Cent always outperforms RM3 in terms of MAP and p@5; all MAP improvements are statistically significant. Among the models that utilize expansions based on individual query terms, RM-CombSUM and RM-CombMAX are the most effective: they always outperform RM3 for MAP and p@5. Finally, the robustness (RI) of each of our methods is higher than that of RM3 over at least two out of the five corpora.

4.3 Further analysis

We use query #546, "recycle cans", from WT10G to shed some light on the advantages of embedding-based query expansion. The 10 terms assigned the highest probability by RM1, Q-Cent and RM-Cent are: {**recycle**, **can**, paint, **plastic**, paper, accept, steel, contain, material, **bottle**}, {**can**, **recycle**, styrofoam, scrap, postconsumer, compost, **plastic**, trash, cardboard, **bottle**} and {**recycle**, **can**, **plastic**, **bottle**, paint, styrofoam, scrap, postconsumer, compost, trash}, respectively. (Boldfaced terms appear in all three lists.) Free parameters were fixed to effective values,

and no integration with the query was performed so as to focus on the effectiveness of the expansion terms.

The average precision (AP) of RM1, Q-Cent and RM-Cent for the query is 0.084, 0.096 and 0.107, respectively, showing again the merits of integrating Word2Vec-based expansion terms with the relevance model. The three models share 4 out of 10 terms. Among the RM1 terms we find general verbs (“accept”, “contain”) which could cause query drift as is common in pseudo-feedback-based methods. In contrast, all the terms in the Q-Cent list are topical (“scrap”, “styrofoam” and “bottle” are all recycled materials/items). This difference translates to a higher AP of Q-Cent than RM1. Naturally, word-embeddings-based expansion can cause query drift, as is the case for other global expansion methods. Indeed, it was shown that local training of word embeddings using retrieved documents can improve expansion effectiveness [6].

We also found that the top terms of the Word2Vec-based expansion and the relevance model often differ. This further attests to the complementary nature of these expansion approaches leveraged by the RM- \mathcal{M} methods.

5. RELATED WORK

Word embeddings were used to re-weight the query terms [20] and to compare a query directly with a document [16, 12]. Document embeddings [5, 2] and word-embeddings-based translation models [8, 21] were also used for retrieval. We focused on query expansion and showed that our methods outperform an embedding-based translation model [21].

Embedding-based query expansion was also used in the medical domain [9], but did not outperform retrieval using only the query. Supervised concept-based embedding of n -grams was used for document retrieval [15]. Our methods utilize unsupervised unigram embeddings.

Recently, query expansion approaches conceptually similar to our Q-CombSUM method were proposed [3, 6, 13]: expansion terms are scored using the (weighted) sum of their embedding-based similarities with query terms. We used additional approaches to integrate these similarities, and combined them with pseudo-feedback-based expansion.

Work on embedding a query language model for query expansion [18] provides formal grounds to using a centroid-based representation of the embedding vectors of the query terms as in our Q-Cent and RM-Cent methods. Our RM-Cent method which integrates the expanded query with a relevance model is different than that which uses an embedding of a relevance model to expand the query [18].

In work to be presented closely in time to ours [17], expansion terms are scored, in spirit, by the arithmetic (similarly to our Q-CombSUM method) and geometric mean of their embedding-based similarities to the query terms; our Q-Cent method was also used; our Q-CombMNZ and Q-CombMAX methods (and their RM- \mathcal{M} counterparts) were not presented. The merits of integrating embedding-based query expansion with a relevance model were demonstrated as in our work; and, the merits of using embeddings for relevance model construction were also shown [17].

6. CONCLUSIONS

We demonstrated the merits of query expansion methods that utilize word embeddings induced in a query-independent manner from the entire document corpus. Specifically, our methods utilize the embeddings to expand the query with

terms that are semantically related to the query as a whole or to its terms. In addition, we showed the effectiveness of an approach for integrating our query expansion methods with a pseudo-feedback-based query expansion approach; namely, the relevance model.

Acknowledgments. We thank the reviewers for their comments. This work was supported in part by the Israel Science Foundation (grant no. 433/12), the Technion-Microsoft Electronic Commerce Research Center, and by the Irwin and Joan Jacobs Fellowship for graduate students.

7. REFERENCES

- [1] N. Abdul-Jaleel, J. Allan, W. B. Croft, F. Diaz, L. Larkey, X. Li, M. D. Smucker, and C. Wade. UMSS at TREC 2004 — novelty and hard. In *Proc. of TREC-13*, 2004.
- [2] Q. Ai, L. Yang, J. Guo, and W. B. Croft. Improving language estimation with the paragraph vector model for ad-hoc retrieval. In *Proc. of SIGIR*, pages 869–872. ACM, 2016.
- [3] M. Almasri, C. Berrut, and J. Chevallet. A comparison of deep learning based query expansion with pseudo-relevance feedback and mutual information. In *Proc. of ECIR*, pages 709–715, 2016.
- [4] C. Carpineto and G. Romano. A survey of automatic query expansion in information retrieval. *ACM Computing Surveys*, 44(1):1, 2012.
- [5] S. Clinchant and F. Perronnin. Aggregating continuous word embeddings for information retrieval. In *Proc. of the Workshop on Continuous Vector Space Models and their Compositionality*, pages 100–109. ACL, 2013.
- [6] F. Diaz, B. Mitra, and N. Craswell. Query expansion with locally-trained word embeddings. *CoRR*, 2016.
- [7] E. A. Fox and J. A. Shaw. Combination of multiple searches. In *Proc. of TREC-2*, 1994.
- [8] D. Ganguly, D. Roy, M. Mitra, and G. J. Jones. Word embedding based generalized language model for information retrieval. In *Proc. of SIGIR*, pages 795–798, 2015.
- [9] T. Goodwin and S. M. Harabagiu. UTD at TREC 2014: Query expansion for clinical decision support. In *Proc. of TREC*, 2014.
- [10] V. Lavrenko and W. B. Croft. Relevance-based language models. In *Proc. of SIGIR*, pages 120–127, 2001.
- [11] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781, 2013.
- [12] E. Nalisnick, B. Mitra, N. Craswell, and R. Caruana. Improving document ranking with dual word embeddings. In *Proc. of WWW*, pages 83–84, 2016.
- [13] D. Roy, D. Paul, M. Mitra, and U. Garain. Using word embeddings for automatic query expansion. In *Proc. of the ACM SIGIR 2016 Workshop on Neural Information Retrieval (Neu-IR 2016)*, 2016.
- [14] T. Sakai, T. Manabe, and M. Koyama. Flexible pseudo-relevance feedback via selective sampling. *ACM Transactions on Asian Language Information Processing (TALIP)*, 4(2):111–135, 2005.
- [15] A. Sordani, Y. Bengio, and J.-Y. Nie. Learning concept embeddings for query expansion by quantum entropy minimization. In *Proc. of AAAI*, pages 1586–1592, 2014.
- [16] I. Vulić and M.-F. Moens. Monolingual and cross-lingual information retrieval models based on (bilingual) word embeddings. In *Proc. of SIGIR*, pages 363–372, 2015.
- [17] H. Zamani and W. B. Croft. Embedding-based query language models. In *Proc. of ICTIR*, 2016.
- [18] H. Zamani and W. B. Croft. Estimating embedding vectors for queries. In *Proc. of ICTIR*, 2016.
- [19] C. Zhai and J. D. Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proc. of SIGIR*, pages 334–342, 2001.
- [20] G. Zheng and J. Callan. Learning to reweight terms with distributed representations. In *Proc. of SIGIR*, pages 575–584, 2015.
- [21] G. Zuccon, B. Koopman, P. Bruza, and L. Azzopardi. Integrating and evaluating neural word embeddings in information retrieval. In *Proc. of ADCS*, pages 12:1–12:8, 2015.