Analysis of the Paragraph Vector Model for Information Retrieval

Qingyao Ai[1], Liu Yang[1], Jiafeng Guo[2], W. Bruce Croft[1]
[1]College of Information and Computer Sciences,
University of Massachusetts Amherst, Amherst, MA, USA
{aiqy, lyang, croft}@cs.umass.edu
[2]CAS Key Lab of Network Data Science and Technology, Institute of Computing Technology,
Chinese Academy of Sciences, China
guojiafeng@ict.ac.cn

ABSTRACT

Previous studies have shown that semantically meaningful representations of words and text can be acquired through neural embedding models. In particular, paragraph vector (PV) models have shown impressive performance in some natural language processing tasks by estimating a document (topic) level language model. Integrating the PV models with traditional language model approaches to retrieval, however, produces unstable performance and limited improvements. In this paper, we formally discuss three intrinsic problems of the original PV model that restrict its performance in retrieval tasks. We also describe modifications to the model that make it more suitable for the IR task, and show their impact through experiments and case studies. The three issues we address are (1) the unregulated training process of PV is vulnerable to short document overfitting that produces length bias in the final retrieval model; (2) the corpus-based negative sampling of PV leads to a weighting scheme for words that overly suppresses the importance of frequent words; and (3) the lack of word-context information makes PV unable to capture word substitution relationships.

CCS Concepts

•**Information systems** → **Language models;** *Document representation;*

Keywords

Paragraph Vector; Language Model

1. INTRODUCTION

Most tasks in information retrieval (IR) benefit from representations that do not treat individual words and documents as unique symbols but reflect their semantic relationships. A common paradigm is to project both words and documents to a latent semantic space and perform matching or language estimation accordingly. This has led to a range of research that incorporates topic models into ad-hoc retrieval tasks. For example, the cluster-based retrieval model [15] and the LDA-based retrieval model [24] have been used to smooth the probability estimation in language modeling approaches with a cluster-based topic model and a Latent Dirichlet Allocation model, respectively. Both methods obtained consistent improvement over the original language models [19].

Recent advances in neural embedding models potentially provide new methods to acquire semantically meaningful representations for words and documents. In particular, Le et al. [13] propose a paragraph vector (PV) model that can jointly learn word and document embeddings through estimating a document level language model. In contrast to topic models, PV does not define a fixed number of topics as a priori. Documents and words are flexibly clustered through the learning of embedding vectors. Meanwhile, PV can be trained with stochastic gradient decent algorithm (SGD), which is simple yet efficient for large-scale learning problems. Previous studies showed that PV has superior performance on several linguistic tasks [5] and great potential for IR [13].

Since PV estimates a document language model, a natural idea is to incorporate it into the language model framework for IR tasks. However, according to our initial experiments, directly combining the original PV with language modeling approaches produces unstable performance and limited improvement. Recently, Ai et al. [1] proposed a retrieval model based on a modified version of PV-DBOW – the paragraph vector model with distributed bags of words assumption. Specifically, they introduced three modifications on the original PV-DBOW: document-frequency based negative sampling, L2 regularization and a joint learning objective. Although they reported positive results on standard ad-hoc retrieval tasks, they did not give detailed analysis on how their modifications affect the language estimation of PV and why they are beneficial for IR.

In this paper, we conduct both a theoretic and empirical analysis on PV-DBOW to define its limitation as a language model for IR. Specifically, we notice three problems when incorporating the original PV-DBOW into language modeling approaches. First, the unregulated learning objective makes PV-DBOW vulnerable to over-fitting. This version of the model tends to retrieve more short documents as

ICTIR '16, September 12-16, 2016, Newark, DE, USA

DOI: http://dx.doi.org/10.1145/2970398.2970409

training iterations increase. Second, the corpus-frequency based negative sampling strategy of PV-DBOW leads to a ICF-like weighting scheme for words in documents, which overly suppresses frequent words. Third, PV-DBOW does not capture word-context information, which makes it unable to model word substitution. By not capturing the substitution relations between words, PV-DBOW produces suboptimal vectors for words and documents which leads to inferior language estimation. In addition to the detailed analysis of these problems, we also provide clear explanations of how they are addressed by L2 regularization, document-frequency based negative sampling, and a joint learning objective. Results on TREC collections indicate that the proposed modifications improve both the effectiveness and robustness of PV-based retrieval models.

The rest of the paper is structured as follows. Section 2 describes the related work and Section 3 introduces the basic structure of PV-based retrieval models. Analysis of the problems and modifications for PV-based retrieval models is presented in Section 4. The proposed modifications are validated with experiments in Section 5. Finally, we conclude our work in Section 6.

2. RELATED WORK

In this section, we briefly review the previous studies in two research fields related to our work, including language smoothing with topic models for IR and neural embedding models.

2.1 Language Smoothing with Topic Models

Language model based retrieval models have been proven to be highly effective for ad hoc retrieval[19, 9]. These models rank documents according to the likelihood of observing a query given the document's language model. The simple language modeling approach estimates language models based on the bag-of-words assumption. This method, however, fails when the query words are not observed in documents. A common solution to this problem is applying smoothing techniques by incorporating a corpus language model for unobserved words. Example approaches include Jelinek-Mercer method, Absolute discounting, and Bayesian smoothing with Dirichlet priors [25].

One issue of language smoothing with the corpus language model is the lack of discrimination. Corpus-based smoothing techniques assume that all documents have similar background probability distributions for unseen words, which makes it difficult to differentiate semantic differences between documents. To overcome this problem, topic models were proposed to produce document specific language estimation by projecting both documents and queries to a same latent semantic space. For example, Deerwester et al. [6] proposed the Latent Semantic Indexing (LSI) technique to extract latent representations for words and documents through the SVD analysis of term frequency matrix. Hoffman [10] introduced the probability Latent Semantic Indexing (pLSI) that models words and documents as mixtures of topics. Blei et al. [3] further extend pLSI by drawing topic mixtures from a conjugate Dirichlet priori. Although these topic models do not work well in retrieval tasks on themselves [2], their combination with the original language models produces positive results. For example, Liu and Croft [15] showed that document clustering can significantly improve the effectiveness of language modeling approaches. Furthermore, Wei and Croft [24] introduced a LDA-based retrieval model that consistently outperforms cluster-based retrieval model and produces state-of-the-art performance for topic models in ad-hoc retrieval.

2.2 Neural Embedding Models

Neural embedding models have received considerable attention in the natural language processing (NLP) community[17, 16, 13, 22]. Mikelov et al. [16] proposed a skip-gram model for learning high-quality word embeddings from large amounts of unstructured text data. These representations capture word similarities at a semantic level and have good compositionality. Le et al. [13] introduced paragraph vector models that project both words and documents into a single semantic space and estimate word probabilities accordingly. Experiments show that paragraph vector models outperform LDA in many NLP tasks such as sentiment analysis and document clustering [5, 13].

Recently, researchers in IR community have applied neural embedding models for retrieval tasks. Vulić et al. [23] and Mitra et al. [18] represented both queries and documents with a composition of word vectors and performed ranking based on the cosine similarity between them. The compositional model with word embedding performs poorly by itself, but it can improve the overall performance of word-based models through rank fusions. Ganguly et al. [7] and Zuccon et al. [27] applied word embeddings in the translation language model framework. They defined translation probabilities based on the cosine similarity between word vectors. More recently, Ai et al. [1] introduced a new method to incorporate neural embedding representations for IR models. Instead of computing cosine similarity, they focused on the probabilistic framework of PV models and its application in language smoothing. They proposed three modifications to adapt PV for retrieval tasks and reported that the enhanced PV can significantly outperform the original PV and existing LDA-based retrieval models. In this paper, we provide a formal analysis and further modifications of this approach.

3. PARAGRAPH VECTOR MODEL FOR IR

In this section, we describe the details of how to apply the original PV model for information retrieval. In this paper, we focus on a specific type of PV model with distributed bag-of-words assumption (PV-DBOW) due to its direct connection with language models of documents.

3.1 PV-DBOW

The original PV-DBOW was proposed by Le et al. [13]. The concept of "paragraph" stands for texts with varied lengths, which can be sentences, paragraphs and, in our case, the whole documents. PV-DBOW assumes the independence between words in a document and uses the document to predict each observed word in it. In this way, PV-DBOW learns both document and word embeddings by estimating a document level language model. Specifically, each document d is first projected into a semantic space and then trained to predict its words w. With the bag-of-words assumption, the generative probability of word w in document d is obtained through a softmax function over vocabulary V_w:

$$P(w|d) = \frac{exp(\vec{w} \cdot \vec{d})}{\sum_{w' \in V_w} exp(\vec{w'} \cdot \vec{d})} \quad (1)$$

where $P(w|d)$ denotes the probability of word w given document d, $\vec{w}$ and $\vec{d}$ denote the vector representations for w and d. To reduce the cost of gradient computation for Equation (1) given a large vocabulary, Mikolov et al. [17] proposed a negative sampling strategy. Negative sampling randomly samples several words according to a predefined noise distribution and uses these words to approximate the denominator of Equation (1). With negative sampling, the global objective of PV-DBOW that sums over all possible word-document pairs is:

$$\ell = \sum_{w \in V_w} \sum_{d \in V_d} \#(w,d) \log(\sigma(\vec{w} \cdot \vec{d})) + \sum_{w \in V_w} \sum_{d \in V_d} \#(w,d)(k \cdot E_{w_N \sim P_V}[\log \sigma(-\vec{w_N} \cdot \vec{d})]) \quad (2)$$

where $\#(w,d)$ is the frequency of observed word-document pairs, V_d represents the corpus of documents, k is the number of negative samples, $\sigma(x)$ is the sigmoid function $\sigma(x) = \frac{1}{1+e^{-x}}$ and $E_{w_N \sim P_V}[\log \sigma(-\vec{w_N} \cdot \vec{d})]$ is the expected value of $\log \sigma(-\vec{w_N} \cdot \vec{d})$ given the noise distribution P_V.

The embedding process of PV-DBOW captures high level semantic information and conveys two major advantages over traditional topic models such as LDA. First, PV-DBOW does not have a fixed number of topics. Documents and words are automatically clustered through the training process without any prior assumptions. Second, PV-DBOW can be efficiently trained through SGD, which is more scalable to a large corpus than traditional probabilistic topic models. According to our experience, training PV-DBOW on a million documents is ten times faster than training LDA with Gibbs sampling on the same collection.

3.2 PV-based Retrieval Model

For each document, PV-DBOW builds a language model that directly estimates the probability of word given a certain document ($P(w|d)$). Therefore, a natural way to use PV-DBOW in the IR scenario is to combine its language estimation with traditional language modeling approaches. Inspired by the idea of the LDA-based retrieval model [24], we use PV-DBOW for language model smoothing in the query likelihood model (QL). Suppose that the word probability estimated with QL (with Dirichlet smoothing) and PV-DBOW are $P_{QL}(w|d)$ and $P_{PV}(w|d)$, the final word probability $P(w|d)$ is obtained through Jelinek-Mercer smoothing:

$$P(w|d) = (1-\lambda) P_{QL}(w|d) + \lambda P_{PV}(w|d) \quad (3)$$

where λ is the parameter that controls smoothing strength. In our experiments, we tried other smoothing methods such as Dirichlet smoothing, but we observed no significant difference between them in retrieval performance.

3.3 Stability of the Model

Our initial experiment show that the PV-based retrieval model indeed outperforms QL model, but its improvement is unstable throughout the training process. On Robust04, we trained PV-DBOW with 300 dimensions and evaluated QL and the PV-based retrieval model with a 5-fold cross validation on title queries (detailed settings are described in Section 5.2 and 5.3). The best mean average precision (MAP) of the PV-based retrieval model with the original PV-DBOW is 0.259, while that for QL model is 0.253. The difference between the PV-based retrieval model and QL is significant ($p < 0.05$), which demonstrates the effectiveness of language smoothing with PV-DBOW. However, we noticed that the performance of PV-based retrieval model is highly sensitive to the training iterations of PV-DBOW. As shown in Figure 1, the MAP of the PV-based retrieval model increases in the beginning, but starts to decrease after 20 iterations. The final performance in the 80 iterations is only slightly better than QL. In the worst cases, the performance improvement from PV-DBOW is inconsistent and marginal, which motivates us to further analyze the limitations of PV-DBOW in language estimation.

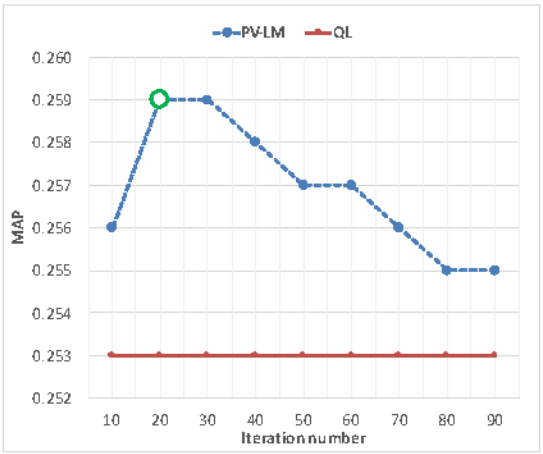

Figure 1: The MAP of QL and the PV-based retrieval model with the original PV-DBOW on Robust04 with title queries in respect of different training iteration. The point with an open circle is the result from Ai et al. [1].

4. PROBLEMS AND MODIFICATIONS

In this section, we conduct an analysis of the reasons for the unstable performance and marginal improvements of the original PV-based retrieval model. Based on this analysis, we talk about the corresponding modifications and show how these modifications affect the language estimation of the PV model.

4.1 Over-fitting on Short Documents

As shown in Section 3.3, one interesting phenomenon is that the performance of the PV-based retrieval model does not converge along with the training iterations. To analyze the possible reasons, we conducted experiments over the top retrieved results of the PV models. Figure 2 shows the distribution of documents with respect to document length in the top 50 documents retrieved by the PV-based retrieval model on Robust04 with title queries. We equally split the domain of document length (0 to 2500) into 50 bins and ignore documents longer than 2500 words (which accounts for less than 4% of the top 50 documents). To avoid confusion, all the models depicted in Figure 2 only use the probability produced by PV-DBOW in language estimation (namely $\lambda = 1$ in Equation (3)). As shown in Figure 2, the distribution of documents with respect to document length gradually moves to the left as training iterations increase for PV-DBOW. The median document length for PV-DBOW is 750-800 under 5 iterations, 600-650 under 20 iterations, and 550-600 under

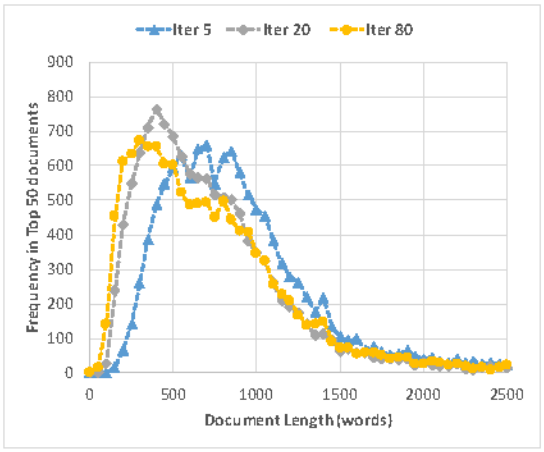

Figure 2: The distribution of documents in respect of document length for top 50 documents retrieved by PV-based retrieval model on Robust04 (title queries). Documents with more than 2500 words are ignored.

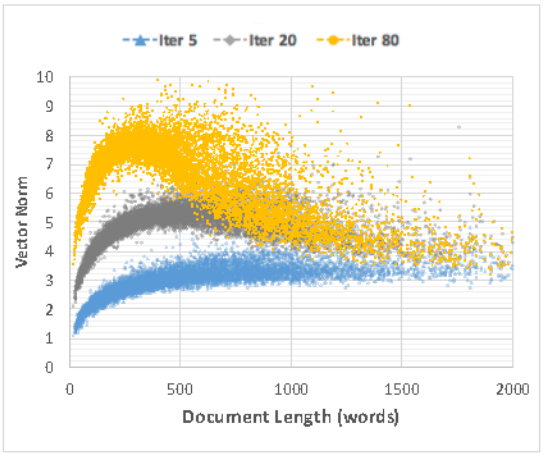

Figure 3: The distribution of vector norms in respect of document length for 10,000 documents randomly sampled from Robust04.

80 iterations. The results indicate that the training process of PV-DBOW introduces increasingly stronger bias toward short documents in the final retrieval model.

To understand the fundamental reason for this length bias, let us look back at the learning process of the PV-DBOW model. As shown in Equation (1), the prediction task of PV-DBOW requires the model to assign higher probability to words that occur in a document than others. In other words, the model will try to align the document vector to the word vectors that appear in the document. This alignment is much easier for short documents since on average the word vectors in short documents would be more concentrated than that in long documents. In practice, concentrated word vectors lead to concentrated gradient directions for document vectors. The partial derivative of the global objective with respect to a certain document d is computed as follows:

$$\frac{\partial \ell}{\partial d} = \sum_{w \in V_w} \#(w,d) \log(\sigma(-\vec{w} \cdot \vec{d}))\vec{w} - \sum_{w \in V_w} \#(w,d)(k \cdot E_{w_N \sim P_V}[\log \sigma(\vec{w_N} \cdot \vec{d})]\vec{w_N}) \quad (4)$$

Despite the part with w_N (which is randomly sampled according to a global noise distribution), we can see that the gradient of d is a weighted sum of its word vectors. Because short documents have less words, their gradients could easily converge to a direction that is not far from all the word vectors. This would result in more rapid increase of norms for short document vectors. Therefore, given an observed word, the probability produced by short documents will become higher and higher, leading to a potential over-fitting.

To verify this, we further plot the variation of the learned document vectors with respect to the document length under different learning iterations. Figure 3 shows the distribution of vector norms for 10,000 documents randomly sampled from Robust04. For documents with more than 1,000 words, vector norms in PV-DBOW with 5, 20 and 80 iterations show no significant difference. However, for documents with less than 1,000 words, the norms of document vectors increases rapidly as iteration number increases.

This analysis shows that the original PV-DBOW suffers from the over-fitting problem along with the training process, and this over-fitting problem is more severe for short documents. A direct method to solve the over fitting problems is to regularize the learning objective of PV-DBOW. Because the over-fitting problem is mainly caused by the unrestricted document vectors, we add an L2 regularizer over the document vectors. More formally, the local objective function for each (w, d) pair with regularization is now as follows:

$$\ell'(w,d) = \ell(w,d) - \frac{\gamma}{|d|}||\vec{d}||^2 \quad (5)$$

where $\ell(w,d)$ represents the local objective function for PV-DBOW, $||\vec{d}||$ denotes the norm of vector $\vec{d}$ and γ denotes a hyper-parameter that control the strength of regularization. Because each iteration of PV-DBOW goes through each word once, a length factor $\frac{1}{|d|}$ where $|d|$ denotes the number of words in document d (namely the length of d) is used to guarantee the same regularization term for all the documents in the training corpus.

The effect of L2 regularization on the language model of PV-DBOW is twofold. First, with L2 regularization, the vector norms for both short documents and long documents are roughly the same along with training iterations. Severe over-fitting on short documents no longer exists in long term training. Second, the restriction on vector norms makes the probability distribution in Equation (1) smoother, which potentially benefits the language smoothing of PV-based retrieval models.

4.2 Improper Noise Distribution

To analyze the reason for the limited performance improvement of the PV-based retrieval model, we first look at the learning objective of PV-DBOW. Inspired by the analysis of the skip-gram model in Levy et al. [14], we derive the local objective for a specific word-document pair from

Equation (2) as:

$$\ell(w,d) = \#(w,d)\log\sigma(\vec{w}\cdot\vec{d}) + k\#(d)P_V(w)\log\sigma(-\vec{w}\cdot\vec{d}) \quad (6)$$

where $\#(d)$ represents the length of d. Define $x = \vec{w}\cdot\vec{d}$, then the objective's partial derivative on x would be:

$$\frac{\partial\ell(w,d)}{\partial x} = \#(w,d)\cdot\sigma(-x) - k\cdot\#(d)\cdot P_V(w)\cdot\sigma(x) \quad (7)$$

Let the partial derivative equal to zero, then the only valid solution for Equation (7) is

$$\vec{w}\cdot\vec{d} = \log(\frac{\#(w,d)}{\#(d)}\cdot\frac{1}{P_V(w)}) - \log k \quad (8)$$

We can see that the original PV-DBOW model conducts implicit factorization over the term-document co-occurrence matrix. The noise distribution of negative sampling actually decides how we weight the terms in a document. The original negative sampling [17] adopts empirical word distribution in the whole corpus as the noise distribution P_V, which is defined as:

$$P_V(w_N) = \frac{\#w_N}{|C|} \quad (9)$$

where $\#(w_N)$ is the corpus frequency of w_N and $|C|$ is the size of the corpus. In Equation (8), $\frac{\#(w,d)}{\#(d)}$ is the normalized TF of w in d, and $\frac{1}{P_V(w)}$ (namely $\frac{|C|}{\#w}$) is the ICF value of w. Therefore, the original PV-DBOW with negative sampling is optimizing for a variation of TF-ICF weighting scheme.

However, TF-ICF is not a popular weighting scheme in IR. One direct reason is that ICF-based term weighting computes the discriminative ability of words only according to their frequency in the corpus and does not consider any form of document structure information. Empirically, a word with high corpus frequency could still be discriminative if it only appears in a small group of documents. This partially explains why PV-DBOW performs well on NLP tasks but not on IR tasks.

Based on these above ideas, one approach to address the problem of PV-DBOW is applying a document-frequency (DF) based negative sampling strategy. More formally, we replace P_V in the original negative sampling with a new noise distribution P_D as follows:

$$P_D(w_N) = \frac{\#D(w_N)}{|N|} \quad (10)$$

where $\#D(w_N)$ denotes the document frequency of w_N and $|N| = \sum_{w'\in V_w}\#D(w')$. After substituting P_V with P_D in Equation (8), we get the new optimal solution as

$$\vec{w}\cdot\vec{d} = \log(\frac{\#(w,d)}{\#(d)}\cdot\frac{|N|}{\#D(w)}) - \log(k) \quad (11)$$

Because $\frac{|N|}{\#D(w)}$ is a variant of the inverse document frequency (IDF) of w, PV-DBOW with DF-based negative sampling is factorizing a shifted matrix of TF-IDF, which is usually considered to be a better scheme for term weighting than TF-ICF [20].

We further plot both the corpus-frequency and document-frequency based distributions in Figure 4 (P_V and P_D respectively). Similar to Church and Gale [4], we observe considerable difference between these sampling distributions, especially on frequent words. As we can see from Figure 4, P_V grows in an exponential way and assigns much higher sample probability to frequent words compared to P_D, which may over-penalize frequent words in the learning of language model. For example, in Robust04 query 339 ("alzheimers drug treatment"), the probability estimated by PV-DBOW with corpus-frequency based negative sampling for "alzheimers" (0.042) is higher than "drug" (0.002) in document FT933-3956, even when "drug" appears two times more than "alzheimers". This suppression makes "drug" less important for the final ranking and consequentially hurts the performance of this query. With document-frequency based negative sampling, the term weighting is moderated and produces more reasonable language estimation (0.056 for "alzheimers" and 0.069 for "drugs" in FT933-3956).

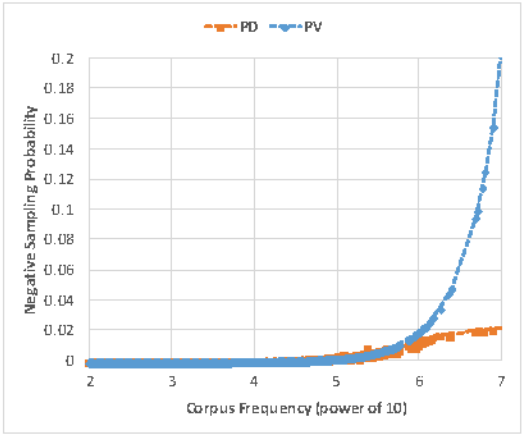

Figure 4: The distribution of the original negative sampling (P_V) and the document-frequency based negative sampling (P_D). The horizontal axis represents log value of word frequency (base 10).

In practice, negative sampling with a very skew distribution is suboptimal for the approximation of softmax function in the learning objective of PV-DBOW. This is the reason why Mikolov et al. [17] applied a unigram distribution raised to the power of 0.75. Similarly, we adopt a power version of document frequency that uses $\#D(w)^\eta (0 \le \eta \le 1)$ to replace $\#D(w)$ in Equation (10).

4.3 Insufficient Modeling for Word Substitution

From the analysis in the prior section, we find that the optimal solution of PV-DBOW's objective function (Equation (6)) is actually an implicit factorization over the term-document matrix. As shown in [22], models that leverage distributed information over the term-document matrix mainly capture words' syntagmatic relations but ignore paradigmatic relations. Syntagmatic relations relate words that co-occur in the same text region. For example, "NBA" is related to "basketball" because they often co-occur in same documents. Paradigmatic relations, namely substitution relations, relate words that often share similar context but may not co-occur in documents. For example, "subway" and "underground" are synonyms and often occur in similar contexts, but American people usually use "subway" while British people tend to use "underground". The original PV-DBOW aligns word vectors to document vectors so that words with high co-occurrence tend to have similar represen-

Table 1: The cosine similarities between *clothing*, *garment* and four relevant documents in Robust04 query 361 ("clothing sweatshops"). EPV-DR represents the PV-based retrieval model with document-frequency based negative sampling and L2 regularization. EPV-DRJ is EPV-DR with a joint objective.

	EPV-DR		EPV-DRJ	
	clothing	*garment*	*clothing*	*garment*
clothing	1.000	0.632	1.000	0.638
LA112689-0194 ($TF_{clothing} = 2, TF_{garment} = 26$)	0.044	0.134	0.107	0.169
LA112889-0108 ($TF_{clothing} = 0, TF_{garment} = 10$)	-0.003	0.100	0.126	0.155
LA021090-0137 ($TF_{clothing} = 7, TF_{garment} = 9$)	0.052	0.092	0.147	0.119
LA022890-0105 ($TF_{clothing} = 6, TF_{garment} = 6$)	0.066	0.079	0.107	0.107

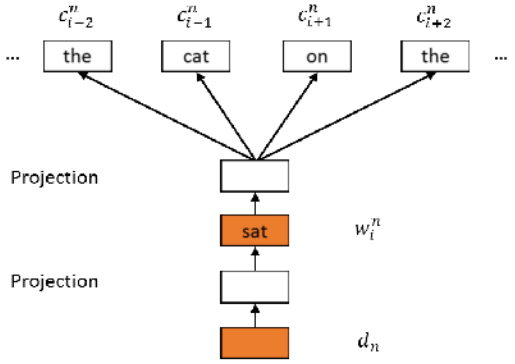

Figure 5: The structure of two-layer PV-DBOW. The document is trained to predict the observed word and then the observed word is trained to predict its context.

tations. However, it cannot model the semantic similarity between words that occur with similar context but not in the same document.

Word paradigmatic information, or word substitution relation is important for IR because it directly alleviates the problem of term mismatch. Term mismatch is common in IR tasks because a query term mismatches 40% to 50% of relevant documents on average [26]. A language model that cannot capture word substitution relation would be vulnerable to the mismatch problem and have limited smoothing ability. Here we take Robust04 query 361 ("clothing sweatshops") as an example. In this query, "garment" is frequent in relevant documents while "clothing" is not. Table 1 lists the cosine similarities between "clothing", "garment" and four relevant documents in the enhanced PV-based retrieval model with document-frequency based negative sampling and L2 regularization (EPV-DR). Intuitively, "clothing" should receive a similar probability to "garment" because they are synonyms. However, EPV-DR assigns much lower cosine similarities for "clothing" than "garment", which consequentially decreases the probability of "clothing" in these relevant documents and lowers their final ranks.

To model word substitution relations, we apply a joint learning objective for PV-DBOW as suggested in [5, 22]. As shown in Figure 5, the first layer of the model uses the document vector to predict the observed word. Then, the second layer of the model uses the observed word to predict its context. More formally, the local objective of the PV-DBOW with the joint objective function can be expressed as

$$\ell = \log(\sigma(\vec{w}_i \cdot \vec{d})) + k \cdot E_{w_N \sim P_D}[\log \sigma(-\vec{w_N} \cdot \vec{d})] + \sum_{\substack{j=i-L \\ j \neq i}}^{i+L} \log(\sigma(\vec{w}_i \cdot \vec{c_j})) + k \cdot E_{c_N \sim P_D}[\log \sigma(-\vec{w}_i \cdot \vec{c_N})] \quad (12)$$

where $\vec{c_j}$ is the context vector for word w_j, c_N denotes the sampled context and L represents the context window size.

From a learning perspective, adding the prediction objective between words and context actually regularizes the learning objective of PV-DBOW. This regularization usually results in better representations for words and documents according to previous studies [5, 22]. In Table 1, after incorporating EPV-DR with the joint objective (EPV-DRJ), the cos similarities between "clothing" and those four relevant documents increase considerably. Even LA112889-0108 (the document in which "clothing" never appears) now has similar cosine similarities for "clothing" and "garment". Therefore, the language estimation of EPV-DRJ based retrieval model gives higher probabilities for "clothing" in those documents and increases the final retrieval performance.

Table 2: Statistics of experimental data sets.

Collection	#Docs	#Words	Size	TREC Topics
Robust-04	528K	253M	1.9G	351-450, 601-700
Gov2	25,205K	24,007M	426G	701-850

5. EXPERIMENTS

In this section, we conduct empirical experiments to verify the effectiveness of different modifications on PV-DBOW for IR.

5.1 Data Set and Baselines

Two TREC collections (Robust04 and GOV2) have been used to evaluate the retrieval performance of PV-based retrieval models and proposed modifications. The statistics of Robust04 and GOV2 are provided in Table 2. We use the Galago search engine[1] to index the corpus and stemmed terms with the Krovetz stemmer [12]. Stop words in queries are removed in advance as suggested in [11]. To better understand the effectiveness of paragraph vector models in information retrieval, we include results from two baselines, i.e. the query likelihood model [19] and the LDA-based retrieval model [24].

Query likelihood (QL) [19] is a basic language modeling

[1] http://www.lemurproject.org/galago.php

approach for information retrieval. It constructs document models with bag-of-words representation and ranks documents according to the log likelihood of query words given the document models. Standard query likelihood model with Dirichlet smoothing [25] can be formulated as Equation (13):

$$P_{QL}(Q|D) = \sum_{w \in Q} tf_{w,Q} log \frac{tf_{w,D} + \mu P(w|C)}{|D| + \mu} \quad (13)$$

where $tf_{w,Q}$ is the number of times that w occurs in the query, $tf_{w,D}$ is the number of times that w occurs in the document, $|D|$ is the length of the document, μ is a parameter for Dirichlet smoothing and $P(w|C)$ is a background language model that is computed as the number of w in the whole corpus divided by the corpus size. To simplify the parameter tuning for both baselines and PV-based retrieval models, we do not tune μ in our experiments and use the average value of the 5-fold validation on Robust04 and GOV2 from Huston and Croft [11]. Specifically, for Robust04 collection, we set $\mu = 934$ for title queries and $\mu = 2166$ for description queries. For GOV2 collection, we set $\mu = 1481$ for title queries and $\mu = 2107$ for description queries.

LDA-based retrieval model (LDA-LM) [3]: LDA is a popular topic model based on a formal generative model of documents. It draws the document-topic distribution $\hat{\theta}$ and topic-word distribution $\hat{\phi}$ from two conjugate Dirichlet priors and models the posterior estimation of word w in document d as:

$$P_{Lda}(w|d) = \sum_{z=1}^{K} P(w|z, \hat{\phi}) P(z|d, \hat{\theta}) \quad (14)$$

where K is the number of topics in LDA model. Proposed by Wei and Croft [24], LDA-based retrieval models combines the original document model from QL with LDA model as:

$$P(w|d) = (1 - \lambda) P_{QL}(w|d) + \lambda P_{Lda}(w|d) \quad (15)$$

where $P_{QL}(w|d)$ is the maximum likelihood estimation of word w in document d with the query likelihood model and $P_{lda}(w|d)$ is the posterior estimation of w given d in the LDA model. In experiments, we use Gibbs sampling to estimate the parameters of LDA and empirically set topic number as $K = 800$. Following previous study [8], the symmetric Dirichlet priors in LDA are set as $\alpha = \frac{50}{K}$ and $\beta = 0.01$.

5.2 Evaluation Framework

We employ four standard retrieval metrics for evaluation: mean average precision (MAP), normlized discounted cumulative gain at 20 (nDCG@20) and precision at 20 (P@20). We list Ai et al. [1]'s results on Robust04 and our own experiments on GOV2 to show the overall retrieval performance of PV-based retrieval model.

Due to the limited number of annotated queries in our experiment collections, we conduct 5-fold cross-validation. We follow the same settings as Huston and Croft[11] and split the query topics for each collections randomly into 5 folds. We tune λ (the combination weight for the LDA-based retrieval model and PV-based retrieval models) with 4 of the 5 folds and test on the remaining 1 fold. The reported numbers are the average value over all test folds. As suggested by Smucker et al.[21], statistical significance is computed with Fisher randomization test with threshold 0.05.

For efficient computation, we adopt a re-ranking strategy. The initial retrieval is performed with query likelihood model to obtain 2,000 candidate documents. Then re-ranking is performed with different models. The final evaluation is carried out on the top 1,000 results.

Table 3: Results from Ai et al. [1] on Robust04 collection measured by MAP. $*$, $+$ means significant difference over QL, LDA-LM respectively at 0.05 significance level measured by Fisher randomization test.

	Robust04 collection	
Method	Titles	Descriptions
QL	0.253	0.246
LDA-LM	0.259^{*}	0.251^{*}
PV-LM	0.259^{*}	0.247
EPV-R-LM	0.259^{*}	0.247
EPV-DR-LM	0.262^{*}	0.252^{*}
EPV-DRJ-LM	$\mathbf{0.267}^{*+}$	$\mathbf{0.253}^{*}$

We trained LDA and paragraph vector models with documents in Robust04 and GOV2 separately. However, handling large scale dataset like GOV2 is computational expensive for LDA. For fair comparison, we randomly sampled 500k documents (including the candidates retrieved by QL) from GOV2 and trained LDA and paragraph vector models on the sampled subset.

5.3 Settings for Paragraph Vector Models

We tested four types of PV-based retrieval models:

- PV-LM: the PV-based retrieval model with PV-DBOW proposed by Le et al. [13].
- EPV-R-LM: the PV-LM model with L2 regularization.
- EPV-DR-LM: the EPV-R-LM model with document-frequency based negative sampling.
- EPV-DRJ-LM: the EPV-DR-LM model with a joint learning objective.

The tuning of all hyper-parameters in PV-DBOW requires considerable effort and is not the core of this paper, so we set most parameters same with the default settings from skip-gram word embedding model proposed in [17][2] except for iteration number. The iteration number is tuned offline with PV-LM from 10 to 80 (10 per step) on Robust04 titles. We observed the best performance under 20 iterations and fix this number for all PV-based retrieval models.

Modification-specific hyper-parameters are tuned separately for EPV-R-LM, EPV-DR-LM and EPV-DRJ-LM. For models with document-frequency based negative sampling, we tuned η from 0.0 to 1.0 (0.1 per step). The best performance for EPV-DR-LM and EPV-DRJ is 0.4 and 0.1. For models with L2 regularization, we tested γ from 0.1, 1, 10 and 100. The best performance is consistently obtained with 10 in EPV-R-LM, EPV-DR-LM and EPV-DRJ-LM.

5.4 Results and Discussion

Ai et al. [1] showed that the proposed modifications for PV-DBOW improve the performance of PV-based retrieval model on Robust04. However, they only reported the best retrieval scores of each model and did not illustrate how

[2]https://code.google.com/p/word2vec/

Table 4: **Comparison of different models over GOV2 collection. $*$, $+$ means significant difference over QL, LDA-LM respectively at 0.05 significance level measured by Fisher randomization test. The best performance is highlighted in boldface.**

	GOV2 collection					
	Titles			Descriptions		
Method	MAP	nDCG@20	P@20	MAP	nDCG@20	P@20
QL	0.295^{+}	0.409	0.510^{+}	0.249^{+}	0.371	0.470
LDA-LM	0.290	0.406	0.505	0.245	**0.376**	0.468
PV-LM	0.294	0.409	0.510^{+}	0.246	0.364	0.463
EPV-R-LM	0.295^{+}	0.410	0.511^{+}	0.250^{+}	0.368	0.467
EPV-DR-LM	0.296^{+}	0.412	0.512	0.250^{+}	0.371	0.470
EPV-DRJ-LM	$\mathbf{0.297}^{+}$	$\mathbf{0.415}^{*+}$	$\mathbf{0.519}^{*+}$	$\mathbf{0.252}^{*+}$	0.371	**0.472**

those models behave in different parameter settings. We extend their work with analysis on PV-based retrieval models with different training iterations and vector dimensions. Our experiments show that the proposed modifications improve both the effectiveness and robustness of PV-based retrieval models.

5.4.1 *Overall Performance*

We refer the results on Robust04 from Ai et al. [1] in Table 3 and further extend the evaluation of baselines and PV-based retrieval models on GOV2 in Table 4. As observed by previous studies [24, 15, 1], topic level estimation is beneficial for language modeling approach. Both LDA-LM and PV-LM outperform QL on Robust04 titles and descriptions. The relative improvements in respect of MAP for LDA-LM are 2.4% on titles and 2.0% on descriptions; for PV-LM are 2.4% on titles and 0.4% on descriptions. The performance of LDA-LM and PV-LM show no significant difference. After adding L2 regularization, document-frequency based negative sampling and a joint objective, the performance of PV-LM increases and finally outperforms all baselines in both Robust04 and GOV2. On Robust04, the relative improvements of MAP for EPV-R-LM, EPV-DR-LM and EPV-DRJ-LM over PV-LM are 0.0%, 1.2% and 3.1% on titles, 0.0%, 2.0% and 2.4% on descriptions; on GOV2, the relative improvements of MAP for EPV-R-LM, EPV-DR-LM and EPV-DRJ-LM over PV-LM are 0.3%, 0.7% and 1.0% on titles, 1.6%, 1.6% and 2.4% on descriptions.

We notice that topic level smoothing tends to be more effective on short queries than long queries. Both LDA-LM and PV-based retrieval models achieve better improvement over QL in title queries than in description queries. For example, the best PV-based retrieval model, EPV-DRJ-LM, outperforms QL with 5.5% on Robust04 titles but 2.5% on Robust04 descriptions in respect of MAP. An explanation for this phenomenon is that vocabulary mismatch is more severe in short queries. With less words in short queries, the missing of one word could hurt the maximum likelihood estimation of QL. In contrast, long queries like descriptions usually have sufficient terms to express their query intents and are more robust to mismatch problems. The introduction of semantic matching to long queries could bring less benefits but more noise. We will give more examples in Section 5.4.4.

In our experiments, GOV2 receives less benefits from semantic smoothing (comparing to Robust04). The incorporation of LDA even damages the performance of language modeling approach in most metrics. One potential reason

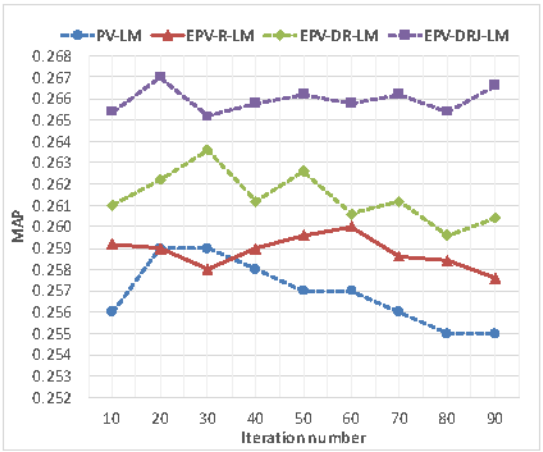

Figure 6: MAP variation of PV-based retrieval models with respect to iteration number. The horizontal axis represents the number of training iterations, and the vertical axis represents MAP on Robust04 title queries.

is that GOV2 consists of web pages, which have a complex and noisy topic distribution comparing to news articles in Robust04. Because our experiments restrict the number of topic in LDA to 800 (due to efficiency), the topics learned by LDA may be too vague and coarse for language estimation. In comparison, although the dimension of the vectors is 300, the number of topics in paragraph vector models is not limited. Because documents are automatically clustered without prior assumptions about topic distribution, PV-DBOW could capture finer semantic relations in a noisy environment. In our experiments, the EPV-DRJ-LM outperforms both QL and LDA-LM in most metrics.

5.4.2 *Iteration Number*

Our analysis shows that the number of training iterations in PV-DBOW have a considerable effect on the language estimation of PV-based retrieval models. To study the effect of training iterations, we depict the MAP value of PV-based retrieval models under different iteration numbers on Robust04 titles in Figure 6.

As shown in Figure 6, the over-fitting problem of PV-LM without L2 regularization is evident as iteration number increases. The best performance of PV-LM (0.259) is observed at 20 iterations, but it drops to 0.255 at 90 iterations. In con-

trast, the results of PV-based retrieval models with L2 regularization (EPV-R-LM, EPV-DR-LM and EPV-DRJ-LM) are steady across different iteration numbers. The MAP of EPV-R-LM slightly wave around 0.259 and consistently outperforms PV-LM after 30 iterations.

Although the L2 regularization can effectively solve the over fitting problem of PV-based retrieval models, it does not significantly improve retrieval performance. By incorporating document-frequency based negative sampling strategy and the join objective, we observed improvement in the MAP scores on Robust04. These results indicate that those modifications together can significantly improve the robustness and effectiveness of PV-based retrieval models.

5.4.3 *Vector Dimensionality*

Previous studies find that higher dimensional vector representation can improve the performance of neural embedding models in NLP tasks [16]. To understand the effect of vector dimensionality, we test PV-based retrieval models with different vector sizes on Robust04 titles and show the results in Figure 7.

In Figure 7, the vector size in PV-DBOW shows a minor correlation with the performance of PV-based retrieval models. Although the MAP value of EPV-DRJ-LM increases slowly from 0.263 to 0.268 when vector dimensionality changes from 50 to 500, the performance of PV-LM fluctuates between 0.256 and 0.259. The improvement caused by increasing vector dimensionality is not consistent in different PV-based retrieval models. Zuccon et al. [27] find that vector dimensionality in word embedding has a minor effect on model performance in ad-hoc retrieval. Similarly, we notice that the setting of dimensionality for PV-based retrieval models is not as important as it is for LDA-LM [24] in language estimation. A potential explanation is that the dimensionality of document vectors is not explicitly linked with the topic number in paragraph vector models. Even with low-dimensional vectors, paragraph vector models can still model a complex topic structure. In our experiments, the EPV-DRJ-LM with 50 dimensions still outperforms the LDA-LM with 800 topics on Robust04 (MAP 0.263 v.s. 0.259).

5.4.4 *Case Studies*

To further illustrate how paragraph vector models work for information retrieval, we conduct case studies to show the advantages and disadvantages of PV-based retrieval models.

The advantages of PV-based retrieval models mostly come from its semantic matching process. We use Robust04 title query 317 ("unsolicited faxes") as an example. In Robust04, only three documents have "unsolicited" and "faxes" simultaneously and two of them contain each word exactly once. QL failed in this case (MAP 0.186) because it cannot reasonably differentiate the relevance of documents that do not have "unsolicited" or "faxes". By projecting documents into semantic concepts, paragraph vector models and LDA provide finer information for the query words and the mismatched documents. As a result, the MAP for EPV-DRJ-LM and LDA-LM in query 317 outperform QL by 75.3% (0.186 to 0.326) and 19.4% (0.186 to 0.222). The results show that both the PV-based and LDA-based retrieval models can improve retrieval performance by involving semantic matching information in language modeling approaches, while PV models can provide even better estimation than LDA model.

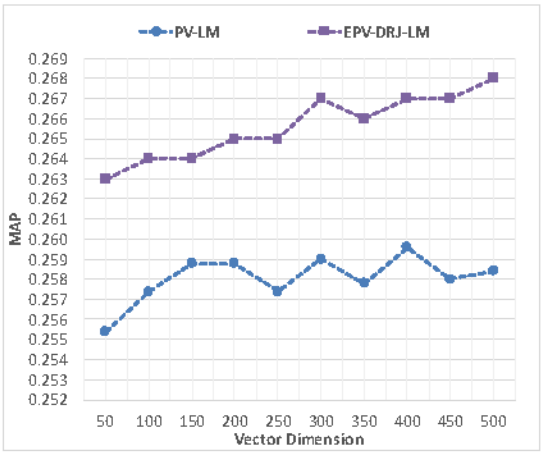

Figure 7: MAP variation of PV-based retrieval models with respect to vector dimensions. The horizontal axis represents vector dimensions, and the vertical axis represents MAP on Robust04 title queries.

However, the semantic matching in PV-based retrieval models may sometimes not work well on long queries. One representative example in our experiments is Robust04 query 614: *flavr savr tomato* (the title), *find information about the first genetically modified food product to go on the market flavr savr also flavor saver tomato developed by calgene* (the description with stopwords removed). With the query title, EPV-DRJ-LM performs better than QL (MAP 0.522 v.s. 0.174) because most of the documents in Robust04 do not contain the exact matching of these query words (only four documents contain "flavr" or "savr"). However, the situation changes when replacing the query title with the query description. One reason is that the query description expands the query title with high quality words that can significantly boost the performance of exact matching model (such as "genetical", "food" and "calgene"). In our experiments, the MAP value of QL is increased by 336.8% (0.174 to 0.76), but the gain for EPV-DRJ-LM is only 44.1% (from 0.522 to 0.752 in MAP). Generally, long queries describe query intents with sufficient information. In this case, semantic matching may bring less benefits but more noise to retrieval models.

6. CONCLUSION

In this paper, we study PV-DBOW with both theoretic and empirical analysis to understand its limitation as a language model for IR. We discuss three problems that restrict the effectiveness of PV-DBOW in IR scenario: over-fitting on short documents, improper negative sampling strategy and lack of word substitution modeling. To address these problems, three modifications for the original PV-DBOW have been proposed. We analyze how these modifications affect the language estimation in PV-DBOW and how they improve the performance of PV-based retrieval models. Experiments and case studies on standard TREC collections are presented to better illustrate and backup our analysis.

Although the discussions of this paper mainly focuses on PV-DBOW for IR, some results are also instructive for future work on other neural embedding models. First, the noise distribution of negative sampling can significantly af-

fect the performance of PV-based retrieval models. With formal inductions, we show that different noise distributions lead PV-DBOW to optimize a different weighting scheme. In this way, one may easily adapt neural embedding models to incorporate different information for different tasks. Second, the norms of embedding vectors contain important information for IR. Previous work mainly focuses on the cosine similarities between embedding vectors, but our analysis show that the norms of embedding vectors also influence the language estimation of PV models. Vector norms in neural embedding models are related to both word frequency and document structures, which could be potentially useful for future studies.

7. ACKNOWLEDGMENTS

This work was supported in part by the Center for Intelligent Information Retrieval, in part by NSF IIS-1160894, and in part by NSF grant IIS-1419693. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsor.

8. REFERENCES

[1] Q. Ai, L. Yang, J. Guo, and W. B. Croft. Improving language estimation with the paragraph vector model for ad-hoc retrieval. In *Proceedings of the 39th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2016.

[2] A. Atreya and C. Elkan. Latent semantic indexing (lsi) fails for trec collections. *ACM SIGKDD Explorations Newsletter*, 12(2):5–10, 2011.

[3] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, Mar. 2003.

[4] K. W. Church and W. A. Gale. Poisson mixtures. *Natural Language Engineering*, 1(02):163–190, 1995.

[5] A. M. Dai, C. Olah, Q. V. Le, and G. S. Corrado. Document embedding with paragraph vectors. In *NIPS Deep Learning Workshop*, 2014.

[6] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. Harshman. Indexing by latent semantic analysis. *JAsIs*, 41(6):391–407, 1990.

[7] D. Ganguly, D. Roy, M. Mitra, and G. J. Jones. Word embedding based generalized language model for information retrieval. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 795–798. ACM, 2015.

[8] T. L. Griffiths and M. Steyvers. Finding scientific topics. *PNAS*, 101(suppl. 1):5228–5235, 2004.

[9] D. Hiemstra and W. Kraaij. Twenty-one at trec-7: Ad-hoc and cross-language track. 1999.

[10] T. Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 50–57. ACM, 1999.

[11] S. Huston and W. B. Croft. A comparison of retrieval models using term dependencies. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pages 111–120. ACM, 2014.

[12] R. Krovetz. Viewing morphology as an inference process. In *Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 191–202. ACM, 1993.

[13] Q. Le and T. Mikolov. Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 1188–1196, 2014.

[14] O. Levy and Y. Goldberg. Neural word embedding as implicit matrix factorization. In *Advances in Neural Information Processing Systems*, pages 2177–2185, 2014.

[15] X. Liu and W. B. Croft. Cluster-based retrieval using language models. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 186–193. ACM, 2004.

[16] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[17] T. Mikolov, I. Sutskever, K. Chen, G. S. CJorrado, and M. I. Dean, Jeffdan. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[18] E. Nalisnick, B. Mitra, N. Craswell, and R. Caruana. Improving document ranking with dual word embeddings. In *Proceedings of the 25th International Conference Companion on World Wide Web*, pages 83–84. International World Wide Web Conferences Steering Committee, 2016.

[19] J. M. Ponte and W. B. Croft. A language modeling approach to information retrieval. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 275–281. ACM, 1998.

[20] S. Robertson. Understanding inverse document frequency: on theoretical arguments for idf. *Journal of documentation*, 60(5):503–520, 2004.

[21] M. D. Smucker, J. Allan, and B. Carterette. A comparison of statistical significance tests for information retrieval evaluation. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 623–632. ACM, 2007.

[22] F. Sun, J. Guo, Y. Lan, J. Xu, and X. Cheng. Learning word representations by jointly modeling syntagmatic and paradigmatic relations. In *Proceedings of the 53rd Annual Annual Meeting of the Association for Computational Linguistics*, 2015.

[23] I. Vulić and M.-F. Moens. Monolingual and cross-lingual information retrieval models based on (bilingual) word embeddings. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 363–372. ACM, 2015.

[24] X. Wei and W. B. Croft. Lda-based document models for ad-hoc retrieval. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '06, pages 178–185, New York, NY, USA, 2006. ACM.

[25] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 334–342. ACM, 2001.

[26] L. Zhao and J. Callan. Term necessity prediction. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 259–268. ACM, 2010.

[27] G. Zuccon, B. Koopman, P. Bruza, and L. Azzopardi. Integrating and evaluating neural word embeddings in information retrieval. In *Proceedings of the 20th Australasian Document Computing Symposium*, pages Article–No. ACM, 2015.