ESSENCE VECTOR-BASED QUERY MODELING FOR SPOKEN DOCUMENT RETRIEVAL

Kuan-Yu Chen, Shih-Hung Liu[#], Berlin Chen[*], Hsin-Min Wang[#]

National Taiwan University of Science and Technology, Taiwan
[*]National Taiwan Normal University, Taiwan
[#]Academia Sinica, Taiwan

ABSTRACT

Spoken document retrieval (SDR) has become a prominently required application since unprecedented volumes of multimedia data along with speech have become available in our daily life. As far as we are aware, there has been relatively less work in launching unsupervised paragraph embedding methods and investigating the effectiveness of these methods on the SDR task. This paper first presents a novel paragraph embedding method, named the essence vector (EV) model, which aims at inferring a representation for a given paragraph by encapsulating the most representative information from the paragraph and excluding the general background information at the same time. On top of the EV model, we develop three query language modeling mechanisms to improve the retrieval performance. A series of empirical SDR experiments conducted on two benchmark collections demonstrate the good efficacy of the proposed framework, compared to several existing strong baseline systems.

Index Terms— Spoken document, retrieval, essence vector, query modeling

1. INTRODUCTION

Over the past two decades, spoken document retrieval (SDR) has become an interesting research subject in the speech processing community because tremendous volumes of multimedia data associated with speech tracks have been made available to the public [1-3]. A significant amount of research effort has been devoted to developing robust indexing techniques to extract probable spoken terms or phrases embedded in a spoken document that could match the query words or phrases literally [4, 5]. More recently, SDR research has also revolved around the notion of relevance of a spoken document in response to a query. It is generally agreed that a document is relevant to a query if it can address the stated information need of the query, but not because it happens to contain all the words in the query [5, 6]. Accordingly, an important research theme of related tasks is to represent queries and documents in a low-dimensional semantic space. By doing so, the relevance degree between a query and a document can be quantified by taking the inferred concept/semantic information into consideration [7-11].

Language modeling (LM) for information retrieval, as well as SDR, has received great attention due to its inherent neat formulation and clear probabilistic meaning, as well as excellent empirical performance [6, 12, 13]. In practice, one of the representative mechanisms is to leverage the Kullback–Leibler divergence measure [14] to determine the relevance degree between a query and a document. Such a mechanism (denoted by "KLM" hereafter) assumes that the words in a query are random draws from a language distribution that describes the information need of a user, and the words in a relevant spoken document are random draws from the same distribution as well. However, a query is usually too short to give an accurate query language model estimate. Therefore, a large body of follow-up work has been dedicated to reformulating the original query language model through a pseudo-relevance feedback process [13, 15-17].

More recently, deep learning has been introduced to infer a low-dimensional semantic space for representing paragraphs, with several empirical studies and experiments showing that such an approach can achieve impressive success on many natural language processing-related tasks [10, 18-20]. However, we observe that most classic paragraph embedding methods infer the representation of a paragraph by considering all of the words within the paragraph. Those stop or function words that occur frequently in the paragraph may mislead the embedding learning process to produce an uninformative paragraph representation. On a separate front, there is still a dearth of work on studying the effectiveness of the paragraph embedding methods in the SDR task. As an attempt to bridge such a research gap, the major contributions of this paper are at least three-fold. First, a novel unsupervised paragraph embedding framework, which aims at not only distilling the most representative information from a paragraph but also excluding the general background information, is introduced. Second, stemming from such a framework, we propose three effective query language models. Finally, a series of empirical evaluations and comparisons are conducted on two benchmark SDR corpora.

2. RELATED WORK

The research trend on learning the representation of a paragraph can trace back to 2010s. Classic methods include the distributed memory (DM) model, the distributed bag-of-words (DBOW) model [21-23], and the autoencoder-based methods [20, 24-26]. The DM model is inspired and hybridized from the traditional feed-forward neural network language model (NNLM) [27] and the recently proposed word embedding methods [28, 29]. Formally, NNLM is designed to predict the probability of a next word, given its $n-1$ immediately preceding words. The input of NNLM is a high-dimensional vector, which is constructed by concatenating (or taking an average over) the word representations of all words within the context, and the output can be viewed as that of a multi-class classifier. By doing so, the n-gram probability can be calculated through a softmax function at the output layer. Based on NNLM, the idea underlying the DM model is that a given paragraph also contributes to the prediction of the next word, given its previous (but not necessarily immediately adjacent) words in the paragraph. Since the learned paragraph representation acts as a memory unit that remembers what is missing from the current context, the model is named the distributed memory model. Different from the DM model, a simplified variant is to leverage only the paragraph representation to predict all of the words within the paragraph. Since the simplified model ignores the contextual words at

the input layer, the model is named the distributed bag-of-words model. In addition to the DM and DBOW models, which infer the associated paragraph representation by predicting all of the words occurring in a paragraph, there are several studies pursuing such a representation by employing the encoder-decoder mechanisms. The basic idea underpinning this line of mechanism is to infer a low-dimensional dense vector representation which can be used to reconstruct the original paragraph or contextual text. This way, the learned paragraph representation is regarded as an informative and compact embedding. Representatives include the semantic hashing [26] and the skip-thought vector [20], to name just a few.

3. THE PROPOSED METHODOLOGY

3.1. The Essence Vector Model

Classic paragraph embedding methods usually infer the representation of a paragraph by considering all of the words within the paragraph. However, the frequent words or modifiers may overshadow the indicative words, thereby making the learned representation deviate from the main theme of the semantic content expressed in the paragraph. Consequently, the associated representation capacity is limited. To overcome this deficiency, we hence investigate a novel unsupervised paragraph embedding method, named the essence vector (EV) model, which aims at not only distilling the most representative information from a paragraph but also discount the impact of the general background information (probably predominated by the stop or function words), so as to deduce a more informative and discriminative low-dimensional vector representation for the paragraph [30, 31].

To crystallize the idea, we begin with an assumption that each paragraph can be assembled by two components: paragraph-specific information and general background information. The assumption also holds in a low-dimensional representation space. Accordingly, given a set of training paragraphs $\{D_1, \cdots, D_t, \cdots, D_T\}$, in order to modulate the effect of different lengths of paragraphs, each paragraph is first represented by a bag-of-words high-dimensional probabilistic vector $P_{D_t} \in \mathbb{R}^{|V|}$, where each element corresponds to the frequency count of a word/term of the vocabulary V in D_t, and the vector is normalized to unit-sum. Then, a paragraph encoder $f(\cdot)$ is applied to extract the most specific information out from the paragraph and encapsulate it into a low-dimensional vector representation v_{D_t}. At the same time, the general background is also represented by a high-dimensional probabilistic vector with normalized word/term frequency counts, $P_{BG} \in \mathbb{R}^{|V|}$, and a background encoder $g(\cdot)$ is used to compress the general background information into a low-dimensional vector representation v_{BG} as well. Since each learned paragraph representation v_{D_t} only contains the most informative/discriminative part of D_t, we assume that the weighted combination of v_{D_t} and v_{BG} can be mapped back to P_{D_t} by a decoder $h(\cdot)$:

$$h\left(\alpha_{D_t} \cdot v_{D_t} + \left(1 - \alpha_{D_t}\right) \cdot v_{BG}\right) = P'_{D_t}, \tag{1}$$

where the combination weight can be determined by a trainable network or a simple linear/non-linear function $q(\cdot,\cdot)$. Further, to ensure the quality of the learned background representation v_{BG}, it should also be able to be mapped back to P_{BG} by the decoder $h(\cdot)$. In a nutshell, the objective function of the EV model is to minimize the total KL-divergence measure:

$$\min_{\theta_f,\theta_g,\theta_h} \sum_{t=1}^{T} \left(P_{D_t} \log \frac{P_{D_t}}{P'_{D_t}} + P_{BG} \log \frac{P_{BG}}{P'_{BG}} \right). \tag{2}$$

To sum up, the essence vector (EV) model consists of three modules: a paragraph encoder $f(\cdot)$ that infers the desired low-dimensional vector representation by considering only the paragraph-specific information; a background encoder $g(\cdot)$ that maps the general background information into a low-dimensional vector representation; and a decoder $h(\cdot)$ that reconstructs the original paragraph by combining the paragraph representation and the background representation.

3.2. Essence Vector-based Query Models

In the context of SDR, by considering a query (or a document) as a paragraph, the EV model can be employed to infer the low-dimensional representation of the query (or the document), which contains only the most representative information and excludes the background information. Then, the relevance degree between a query and a document can be determined by the cosine similarity measure between their vector representations.

Opposite to the above simple strategy, a convenient property inherits in the EV model is that it can be readily integrated into the KLM method widely used in IR. Fundamentally, the EV model is used to distill the indicative information from a given paragraph D_t, so as to deduce a vector representation v_{D_t} for the paragraph. Consequently, by feeding the vector representation to the shared decoder $h(\cdot)$, we can reconstruct a bag-of-words high-dimensional probabilistic vector for the paragraph,

$$\hat{P}'_{D_t} \equiv h\left(v_{D_t}\right). \tag{3}$$

In this way, $\hat{P}'_{D_t}$ is considered a paragraph specific language model, which contains only the paragraph specific information v_{D_t} leaving out the general background information. By doing so, the relevance degree between a query and a document can be determined by computing the KL-divergence between the corresponding two language models (i.e., by the KLM method).

However, like many IR tasks, the SDR task also suffers from the short query problem. That is, a query is usually composed of only a few words, and thus the statistics of the query would be sparse and uninformative. Since the learned representation is meant to represent the information need of a user, we adopt a pseudo-relevance feedback process to reformulate the original user query. Formally, in the first round of retrieval, the original query is input into a SDR system to retrieve a number of top-ranked documents $\mathbf{R} = \{D_1^R, \cdots, D_r^R, \cdots, D_{|\mathbf{R}|}^R\}$ (denoted as the feedback documents hereafter) for relevance feedback purposes. Subsequently, on top of these feedback documents, a refined query language model is constructed, and a second round of retrieval is conducted with this new query language model by the KLM method. It is usually anticipated that the SDR system can thus probe more relevant documents in the second round retrieval. In order to effectively utilize the selected set of feedback documents, this study proposes three modeling strategies for estimating a more accurate query language model based on the EV model.

3.2.1. Sample Pooling

A straightforward strategy to enrich the statistics of a user query is to gather all of the statistics from the feedback documents. As such, rich statistics can be obtained and used to render a new bag-of-words high-dimensional probabilistic vector. In practice, we pool every feedback document vector, $P_{D_r^R} \in \mathbb{R}^{|V|}$, weighted by its relevance score to the original query, which distinguishes highly relevant documents from less relevant ones, to yield a new representation, $P_{\hat{Q}}$, for a given query Q:

$$P_{\hat{Q}} = \beta \cdot P_Q + (1 - \beta) \cdot \left(\sum_{r=1}^{|\mathbf{R}|} s(Q, D_r^R) \cdot P_{D_r^R}\right), \tag{4}$$

where $s(Q, D_r^R)$ is the normalized similarity score for each feedback document D_r^R, and β is a weighting factor to modulate the information between the original query and the feedback documents. Finally, the new query representation, $v_{\hat{Q}}$, can be derived by feeding $P_{\hat{Q}}$ into encoder $f(\cdot)$, and then the query specific language model $\hat{P}'_{\hat{Q}}$ can be obtained by feeding $v_{\hat{Q}}$ into $h(\cdot)$. As such, each query Q has its own EV-based language model $\hat{P}'_{\hat{Q}}$. This method is referred to as the "sample pooling" method.

3.2.2. Vector Pooling

Due to the fact that the EV model aims at projecting a given paragraph into a low-dimensional semantic space, one reasonable manipulation is to create the new representation in the vector space directly. To formulate the idea, we can first interpret each feedback document D_r^R by its own inferred representation $v_{D_r^R}$ given by $f(\cdot)$. Then, the refined query representation can be obtained by pooling together all $v_{D_r^R}$ weighted by their normalized similarity scores $s(Q, D_r^R)$:

$$v_{\hat{Q}} = \gamma \cdot v_Q + (1-\gamma) \cdot (\sum_{r=1}^{|\mathbf{R}|} s(Q, D_r^R) \cdot v_{D_r^R}), \quad (5)$$

where γ is a weighting factor to strike a balance between the information distilled from the original query and the feedback documents. By feeding $v_{\hat{Q}}$ into $h(\cdot)$, the query-specific language model $\hat{P}'_{\hat{Q}}$ can be obtained. We term this pooling function as the "vector pooling" method.

3.2.3. Model Pooling

In addition to the above two pooling methods, another notion is that each query Q is assumed to be associated with an unknown relevance class R_Q, and words that are relevant to the semantic content expressed in Q are samples drawn from the relevance class R_Q. Thus, our goal turns to estimate the probability that each distinct word w occurs in the relevance class. However, in reality, there is no prior knowledge about R_Q, thus we may employ the feedback documents to approximate the relevance class R_Q. Further, the query can be introduced to quantify the approximate degree between the relevance class and each feedback document:

$$P(w|R_Q) \propto \sum_{r=1}^{|\mathbf{R}|} P(w, Q|D_r^R) \quad (6)$$

Finally, the joint probability of w and Q can be estimated by assuming that query words q and w are independent of one another:

$$P(w, Q|D_r^R) = P(w|D_r^R) \prod_{q \in Q} P(q|D_r^R), \quad (7)$$

where the component model $P(\cdot\,|D_r^R)$ is obtained by passing $v_{D_r^R}$ of each feedback document D_r^R through $h(\cdot)$. Since this strategy is functioned at the model level, we name this mechanism as the "model pooling" method.

4. EXPERIMENTS

4.1. Experimental Setup

The Mandarin Chinese collections of the TDT corpora (i.e., TDT-2 and TDT-3) are used for the retrospective retrieval task in this study[1][2]. The titles of Chinese news stories from Xinhua News Agency are used as our test queries and the Mandarin news stories from Voice of America news broadcasts as the spoken documents. For the TDT-2 corpus, on average, each query contains 7 words, and each document contains 173 words. For TDT-3, each query contains 9 words, and each document is 253 words. All news stories are exhaustively tagged with event-based topic labels which serve as the relevance judgments for performance evaluation. The Dragon large-vocabulary continuous speech recognizer provided Chinese word transcripts for the Mandarin audio collections. The average word error rate (WER) obtained for the spoken documents in TDT-2 and TDT-3 is about 35% and 36%, respectively. The non-interpolated mean average precision (MAP) following the TREC evaluation [6] is used to quantify the retrieval results. All networks built in the EV model are implemented with fully connected multilayer neural networks. The activation function used in the EV model is the hyperbolic tangent, except that the output layer in the decoder $h(\cdot)$ adopts the softmax. The cosine distance is used to calculate the attention coefficients, while the Adam [32] algorithm is employed to solve the optimization problem of all the model parameters.

Table 1. *Retrieval results (in MAP) of different paragraph embedding methods with the cosine similarity measure.*

	TDT-2		TDT-3	
	TD	SD	TD	SD
VSM	0.339	0.275	0.442	0.378
DM	0.344	0.302	0.442	0.382
DBOW	0.362	0.345	0.472	**0.428**
EV	**0.382**	**0.364**	**0.474**	0.409

Table 2. *Retrieval results (in MAP) of different language model-based IR methods.*

	TDT-2		TDT-3	
	TD	SD	TD	SD
KLM	0.372	0.323	0.438	0.395
LDA	0.401	0.341	0.458	0.418
RM	0.421	0.369	0.469	0.431
SMM	0.415	0.361	0.461	0.407
EV (Sample Pooling)	0.499	**0.452**	0.514	**0.481**
EV (Vector Pooling)	**0.522**	0.449	0.518	0.471
EV (Model Pooling)	0.516	0.407	**0.553**	0.469

4.2. Experimental Results

To begin with, we investigate the utilities of the vector space model (VSM), two classic paragraph embedding methods (i.e., DM and DBOW) [21], and the proposed EV model for SDR. In this set of experiments, for all systems, each query and document is represented by a vector, and the relevance degree is computed by the cosine similarity measure. For the EV model, the representation of each query and document is obtained by passing it through the paragraph encoder (i.e., $f(\cdot)$). In this study, we take a step forward to make a comparison between SDR and traditional text retrieval. Consequently, the retrieval results, assuming manual transcripts for the spoken documents to be retrieved (denoted by TD) are known, are also shown for reference, compared to the results when only the erroneous transcripts by speech recognition are available (denoted by SD). Experimental results are shown in Table 1. The best result within each column (corresponding to a specific evaluation condition) is type-set boldface. Inspection of these results reveals three noteworthy points. First, both of the celebrated paragraph embedding methods (i.e., DM and DBOW) outperform VSM, and DBOW consistently outperforms DM by a large margin, when applied to either text documents (i.e., the TD case) or spoken documents (i.e., the SD case). Second, the proposed EV model

[1] https://catalog.ldc.upenn.edu/LDC2001S93

[2] https://catalog.ldc.upenn.edu/LDC2001S95

Table 3. *Retrieval results (in MAP) of EV-based query modeling methods with respect to the number of feedback documents.*

	TDT-2						TDT-3					
	TD			SD			TD			SD		
|R|	Sample Pooling	Vector Pooling	Model Pooling	Sample Pooling	Vector Pooling	Model Pooling	Sample Pooling	Vector Pooling	Model Pooling	Sample Pooling	Vector Pooling	Model Pooling
1	0.430	0.471	0.469	0.394	0.382	0.374	0.507	**0.518**	0.537	0.449	**0.471**	**0.469**
3	0.444	0.514	0.460	0.398	0.403	0.386	**0.514**	0.504	0.537	**0.481**	0.465	0.463
5	0.450	0.501	0.486	0.416	0.424	0.400	0.514	0.512	**0.553**	0.456	0.439	0.458
10	**0.499**	**0.522**	**0.516**	**0.452**	**0.449**	**0.407**	0.507	0.512	0.544	0.438	0.434	0.438

demonstrates superior results over VSM, where each word is represented by multiplying its term frequency with the inverse document frequency (TF-IDF). Although IDF is used to constrain the frequent words and reveal the discriminative statistics in a paragraph, the suppressed/promoted degree of contribution for a word is constant across all paragraphs. In contrast, in the EV model, the combination degree between the paragraph specific information and the general background information is determined by a learned function (i.e., α_D; cf. Section 3.1). The EV model appears to be more flexible than VSM, thereby yielding better results. Third, the EV model outperforms both DM and DBOW in most cases, which indicates that the EV model can really yield more informative and discriminative representations for paragraphs, compared to the two classic paragraph embedding methods. Fourth, the performance gap between the retrieval on the manual transcripts (i.e., the TD case) and that on the recognition transcripts (i.e., the SD case) is about 10% in terms of MAP, which also shows that the recognition errors inevitably mislead the representation learning for a paragraph and thus will degrade the retrieval performance. In a nutshell, the results not only demonstrate the promise of the representation learning techniques for SDR, but also demonstrate the remarkable potential of the proposed EV model in both TD and SD cases.

Next, we compare several state-of-the-art LM-based IR models with the proposed EV-based query language models for SDR. The results are summarized in Table 2. KLM is the baseline system where the query and document language models are derived by the maximum likelihood estimator. LDA denotes latent Dirichlet allocation, in which each document language model is estimated by leveraging a probabilistic topic model [33]. In addition, two well-practiced query reformulation methods, namely the relevance model (RM) [34, 35] and the simple mixture model (SMM) [30, 35], are also compared here. It is worthy to note that RM, SMM, and all of the proposed EV-based query language modeling methods only reformulate the original query language model, while the document language model is derived by the maximum likelihood estimator as in the KLM system. The feedback documents with the similarity scores $s(Q, D_r^R)$ are selected by referring to the KLM results. Several observations can be drawn from Table 2. First, language model-based methods in general outperform vector space-based methods (*cf.* Tables 1). The results show that language model-based methods are a school of efficient and effective mechanism for SDR. Second, LDA outperforms KLM, while RM and SMM outperform LDA. The results indicate that deriving a more accurate query language model appears to be more effective than building an enhanced document model. The reason might be that a document usually contains relatively sufficient statistics to estimate a reliable language model, as compared to a short query. Third, RM consistently outperforms SMM in all cases. The results align well with those of previous studies. Fourth, the proposed EV-based query modeling methods yield comparable performance, and they all outperform LDA, RM and SMM by a large margin. Finally, a particularly noteworthy observation is that the retrieval results achieved by the proposed EV-based framework in the SD cases are even better than those obtained by the baseline systems (i.e., KLM, LDA, RM, and SMM) in the TD cases. The results demonstrate that the EV-based framework not only learns to distill the representative information from a paragraph, but also manages to encapsulate *homogeneous* information in the paragraph, thereby reducing the *noisy* information caused by the recognition errors made on feedback document. Moreover, the performance of the EV-based framework may be further improved if the EV model employs advanced neural network techniques, such as maxout, dropout, and recurrent/convolutional neural networks. We will study them in our future work.

In the third set of experiments, we look into the impact of the number of feedback documents on the EV-based query language modeling methods. As revealed by the results illustrated in Table 3, leveraging a large number of feedback documents (e.g., 10) seems to benefit their performance for the TDT-2 corpus, while using a small number of feedback documents (e.g., 1 and 3) seems to be adequate for the TDT-3 corpus. This can be attributed to the fact that the feedback documents in the TDT-3 task are seemingly more relevant to the original queries than those in the TDT-2 task (cf. KLM in Table 2, TDT-3 has better MAP performance than TDT-2). Nevertheless, the way to systemically determine the optimal number of feedback documents for each query reformulation method remains an open issue and needs further investigation. Table 3 also signals that the model pooling method seems to be the best choice for the TD case, while the sample pooling method seems to be most robust to the recognition errors. To sum up, the proposed EV model can offer concise vector representations for a simple cosine similarity measure, and the extended query modeling methods can further enhance the retrieval performance when paired with the KLM.

5. CONCLUSIONS

In this paper, we have presented the essence vector (EV) model, which can be leveraged to learn a semantic representation for a given paragraph in an unsupervised manner, and three EV-based query language modeling methods for SDR. All of the proposed methods have been evaluated on two SDR benchmark corpora. Experimental results demonstrate their remarkable superiority than other strong baselines compared in the paper, thereby indicating the potential of the new paragraph embedding framework. For future work, we will explore the incorporation of extra cues, such as acoustic statistics and sub-word information, into the proposed framework for the SDR task. We also plan to evaluate the framework on other large-scale IR corpora and NLP-related tasks.

6. ACKNOWLEDGMENT

This work was supported in part by the Ministry of Science and Technology of Taiwan under Grants: MOST 105-2221-E-001-012-MY3 and MOST 106-2218-E-011-019-MY3.

7. REFERENCES

[1] Lin-shan Lee and Berlin Chen, "Spoken document understanding and organization," *IEEE Signal Processing Magazine*, 22(5):42-60, 2005.

[2] Ciprian Chelba, Timothy J. Hazen, and Murat Saraclar, "Retrieval and browsing of spoken content," *IEEE Signal Processing Magazine*, 25(3):39-49, 2008.

[3] Mari Ostendorf, "Speech technology and information access," *IEEE Signal Processing Magazine*, 25(3):150-152, 2008.

[4] Sparck K. Jones, Stephen Walker, and Stephen E. Robertson, "A probabilistic model of information retrieval: development and comparative experiments (parts 1 and 2)," *Information Processing and Management*, 36(6):779-840, 2000.

[5] Christopher D. Manning, Prabhakar Raghavan and Hinrich Schtze, *Introduction to Information Retrieval*, New York: Cambridge University Press, 2008.

[6] Bruce Croft and John Lafferty (eds.), *Language modeling for information retrieval*, Kluwer International Series on Information Retrieval, Volume 13, Kluwer Academic Publishers, 2003.

[7] Scott Deerwester, Susan Dumais, George Furnas, Thomas Landauer, and Richard Harshman, "Indexing by latent semantic analysis," *Journal of the American Society of Information Science*, 41(6):391-407, 1990.

[8] Thomas Hofmann, "Unsupervised learning by probabilistic latent semantic analysis," *Machine Learning,* 42:177-196, 2001.

[9] David M. Blei, "Probabilistic topic models," *Commun. ACM*, 55(4):77-84, 2012.

[10] Hamid Palangi, Li Deng, Yelong Shen, Jianfeng Gao, Xiaodong He, Jianshu Chen, Xinying Song, and Rabab Ward, "Deep sentence embedding using the long short term memory network: analysis and application to information retrieval," *arXiv:1502.06922*, 2015.

[11] Debasis Ganguly, Dwaipayan Roy, Mandar Mitra, and Gareth J.F. Jones, "Word embedding based generalized language model for information retrieval," in *Proceedings of SIGIR*, pages 795-798, 2015.

[12] Yuanhua Lv and ChengXiang Zhai, "A comparative study of methods for estimating query language models with pseudo feedback," in *Proceedings of CIKM*, pages 1895-1898, 2009.

[13] Kyung-Soon Lee and Bruce Croft, "A deterministic resampling method using overlapping document clusters for pseudo-relevance feedback," *Inf. Process. Manage.* 49(4):792-806, 2013.

[14] Solomon Kullback and Richard A. Leibler, "On information and sufficiency," *The Annals of Mathematical Statistics*, 22(1):79-86, 1951.

[15] Tao Tao and ChengXiang Zhai, "Regularized estimation of mixture models for robust pseudo-relevance feedback," in *Proceedings of SIGIR*, pages 162-169, 2006.

[16] Claudio Carpineto and Giovanni Romano, "A survey of automatic query expansion in information retrieval," *ACM Computing Surveys*, 44:1-56, 2012.

[17] Stéphane Clinchant and Eric Gaussier, "A theoretical analysis of pseudo-relevance feedback models," in *Proceedings of ICTIR*, pages 1-6, 2013.

[18] Yoshua Bengio, Aaron Courville, and Pascal Vincent, "Representation learning: a review and new perspectives," *Pattern Analysis and Machine Intelligence*, 35(8):1798-1828, 2013.

[19] Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck, "Learning deep structured semantic models for web search using clickthrough data," in *Proceedings of CIKM*, pages 2333-2338, 2013.

[20] Ryan Kiros, Yukun Zhu, Ruslan Salakhutdinov, Richard S. Zemel, Antonio Torralba, Raquel Urtasun, and Sanja Fidler, "Skip-thought vectors," in *Proceedings of NIPS*, pages 3294-3302, 2015.

[21] Quoc Le and Tomas Mikolov, "Distributed representations of sentences and documents," in *Proceedings of ICML*, pages 1188-1196, 2014.

[22] Kuan-Yu Chen, Hung-Shin Lee, Hsin-Min Wang, and Berlin Chen, "I-vector based language modeling for spoken document retrieval," in *Proceedings of ICASSP*, pages 7083-7088, 2014.

[23] Andrew M. Dai, Christopher Olah, and Quoc Le, "Document embedding with paragraph vectors," *arXiv:1507.07998*, 2015.

[24] Ruslan Salakhutdinov and Geoffrey Hinton, "Using deep belief nets to learn covariance kernels for Gaussian processes," in *Proceedings NIPS*, pages 1249-1256, 2007.

[25] Marc' Aurelio Ranzato and Martin Szummer, "Semi-supervised learning of compact document representations with deep networks," in *Proceedings of ICML*, pages 792-799, 2008.

[26] Ruslan Salakhutdinov and Geoffrey Hinton, "Semantic hashing," *International Journal of Approximate Reasoning,* 50(7):969-978, 2009.

[27] Yoshua Bengio, Rejean Ducharme, Pascal Vincent, and Christian Jauvin, "A neural probabilistic language model," *Journal of Machine Learning Research* (3):1137-1155, 2003.

[28] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean, "Efficient estimation of word representations in vector space," in *Proceedings of ICLR*, 2013.

[29] Jeffrey Pennington, Richard Socher, and Christopher D. Manning, "GloVe: Global vector for word representation," in *Proceedings of EMNLP*, pages 1532-1543, 2014.

[30] Chengxiang Zhai and John Lafferty, "Model-based feedback in the language modeling approach to information retrieval," in *Proceedings of CIKM*, pages 403-410, 2001.

[31] Kuan-Yu Chen, Shih-Hung Liu, Berlin Chen, and Hsin-Min Wang, "Learning to distill: the essence vector modeling framework," in *Proceedings of Coling*, pages 358-368, 2016.

[32] Diederik Kingma and Jimmy Ba, "ADAM: A method for stochastic optimization," in *Proceedings of ICLR*, 2015.

[33] Xing Wei and Bruce Croft, "LDA-based document models for ad-hoc retrieval," in *Proceedings of SIGIR*, pages 178-185, 2006.

[34] Victor Lavrenko and Bruce Croft, "Relevance based language models," in *Proceedings of SIGIR*, pages 120-127, 2001.

[35] Kuan-Yu Chen, Shih-Hung Liu, Berlin Chen, Ea-Ee Jan, Hsin-Min Wang, Wen-Lian Hsu, and Hsin-Hsi Chen, "Leveraging effective query modeling techniques for speech recognition and summarization," in *Proceedings of EMNLP*, pages 1474-1480, 2014.