A LOCALITY-PRESERVING ESSENCE VECTOR MODELING FRAMEWORK FOR SPOKEN DOCUMENT RETRIEVAL

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

Academia Sinica, Taiwan
*National Taiwan Normal University, Taiwan

ABSTRACT

Because unprecedented volumes of multimedia data associated with spoken documents have been made available to the public, spoken document retrieval (SDR) has become an important research area in the past decades. Recently, representation learning has emerged as an active research topic in many machine learning applications owing largely to its excellent performance. In the context of natural language processing, the pioneering work can date back to the word embedding methods. However, learning of paragraph (or sentence and document) representations is more reasonable and suitable for some tasks, such as information retrieval and document summarization. Nevertheless, as far as we are aware, there is relatively less work focusing on launching paragraph embedding methods into SDR. Motivated by these observations, this paper proposes a novel paragraph embedding method, named the locality-preserving essence vector (LPEV) model. LPEV is designed with consideration to two aspects. First, the model aims at not only distilling the most representative information from a paragraph but also getting rid of the general background information. Second, inspired by the local invariance perspective, which is a celebrated principle used in manifold learning techniques, LPEV also manages to preserve semantic locality in the learned low-dimensional embedding space for producing more informative and discriminative vector representations of paragraphs. On top of the proposed framework, a series of empirical SDR experiments conducted on the TDT-2 (Topic Detection and Tracking) collection demonstrate the good efficacy of our SDR methods as compared to existing strong baselines.

Index Terms— Representation, spoken document retrieval, locality, distill

1. INTRODUCTION

Over the past two decades, spoken document retrieval (SDR) [1, 2] has become an interesting research subject in the speech processing community due to large volumes of multimedia data associated with spoken documents made available to the public. A significant amount of research effort has been devoted towards developing robust indexing (or representation) techniques [3-6] so as to extract probable spoken terms or phrases inherent in a spoken document that could match the query words or phrases literally. 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 [7]. In the past, the vector space model (VSM) [7, 8], the Okapi BM25 model [7, 9], and the unigram language model [10, 11] are well-representative ones for many information retrieval (IR) and SDR applications. Their efficient and effective abilities have been validated by many researchers and practitioners for a wide variety of IR-related tasks. Yet, the later effort for further extending these methods to capture context dependence based on *n*-grams of various orders or some grammar structures mostly lead to mild gains or even spoiled results [10, 11]. The reasons for this phenomenon are twofold. First, this is due to the fact that these methods might suffer from the problem of word usage diversity, which sometimes degrades the retrieval performance severely as a given query and its relevant documents may use quite different sets of words (e.g. synonyms). Second, lots of polysemy words have different meanings in different contexts. As such, merely matching words occurring in the original query and a document may not capture the semantic intent of the query.

On a separate front, representation learning has gained significant interest of research and experimentation in many machine learning applications because of its amazing performance. When it comes to the field of natural language processing (NLP), word embedding methods can be viewed as pioneering studies [12-14]. A common thread of leveraging word embedding methods to NLP-related tasks is to represent a given paragraph (or sentence and document) by simply taking an average over the word embeddings corresponding to the words occurring in the paragraph. By doing so, this thread of methods has enjoyed substantial success in many tasks [15-17]. Although the empirical effectiveness of word embedding methods has been proven recently, the composite representation for a paragraph (or sentence and document) is a bit queer. Theoretically, paragraph-based representation learning is expected to be more suitable for such tasks as information retrieval, sentiment analysis and document summarization [18-21], to name but a few. However, to the best of our knowledge, paragraph embedding has been largely under-explored on these tasks.

Classic paragraph embedding methods infer the representation of a given paragraph by considering all of the words occurring in the paragraph. Consequently, those stop or function words that occur frequently in the paragraph may mislead the embedding learning process to produce a misty paragraph representation. In other words, the frequent words or modifiers may overshadow the indicative words, thereby drifting the main theme of the semantic content in the paragraph. As a result, the learned representation for the paragraph might be undesired. Moreover, it is obvious that classic paragraph embedding methods only take surface statistics (term, word, or character) into account. By doing so, the deduced representation might suffer from the problem of word usage diversity and could not accurately embed the semantic relationship among paragraphs. In order to address these shortcomings, we propose a novel locality-preserving essence vector (LPEV) model, which aims at not only distilling the most representative information from a paragraph but also preserving semantic locality to produce a more informative and discriminative low-dimensional vector representation for a given paragraph.

2. RELATED WORK

In contrast to the large body of work on developing various word embedding methods, there are relatively few studies concentrating on learning paragraph representations [18-21]. Representative methods include the distributed memory model [18] and the distributed bag-of-words model [18, 19].

2.1. The Distributed Memory Model

The distributed memory (DM) model is inspired and hybridized from the traditional feed-forward neural network language model (NNLM) [12] and the recently proposed word embedding methods [13]. Formally, given a sequence of words, $\{w^1, w^2, \cdots, w^L\}$, the objective function of feed-forward NNLM is to maximize the total log-likelihood,

$$\sum_{l=1}^{L} \log P(w^l | w^{l-n+1}, \cdots, w^{l-1}). \quad (1)$$

Obviously, NNLM is designed to predict the probability of the future word, given its $n-1$ previous 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 (i.e., $w^{l-n+1}, \cdots, w^{l-1}$), 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 the NNLM, the idea underlying the DM model is that a given paragraph also contributes to the prediction of the next word, given its previous words in the paragraph [18]. To make the idea work, the training objective function is defined by

$$\sum_{t=1}^{\mathrm{T}} \sum_{l=1}^{L_t} \log P(w^l | w^{l-n+1}, \cdots, w^{l-1}, D_t), \quad (2)$$

where T denotes the number of paragraphs in the training corpus, D_t denotes the t-th paragraph, and L_t is the length of D_t. Since the paragraph representation (i.e., D_t) acts as a memory unit that remembers what is missing from the current context, the model is named the distributed memory model.

2.2. The Distributed Bag-of-Words Model

Opposite to the DM model, a simplified version is to only leverage the paragraph representation to predict all of the words occurring in the paragraph [18, 19]. The training objective function can then be defined by maximizing the predictive probabilities all over the words occurring in the paragraph:

$$\sum_{t=1}^{\mathrm{T}} \sum_{l=1}^{L_t} \log P(w^l | D_t). \quad (3)$$

Since the simplified model ignores the contextual words at the input layer, the model is named the distributed bag-of-words (DBOW) model. In addition to being conceptually simple, the DBOW model only needs to store the softmax weights, whereas the DM model stores both softmax weights and word vectors [18].

3. THE LOCALITY-PRESERVING ESSENCE VECTOR MODELING FRAMEWORK

Classic paragraph embedding methods infer the representation for a given paragraph by considering all of the words occurring in the paragraph. However, we all agree upon that the number of content words in a paragraph is usually less than that of stop or function words. Namely, those stop or function words may misguide the representation learning process to produce an ambiguous paragraph representation. Consequently, the associated performance gains will be limited. Another flaw of these methods is that they only take surface statistics (such as term, word, or character) into account when inferring the representation for a given paragraph. Consequently, the deduced representations might suffer from the problem of word usage diversity and could not accurately embed the semantic relationship among paragraphs so as to degrade the associated performance. In order to remedy the aforementioned flaws, we hence strive to develop a novel paragraph embedding method with two orthogonal objectives: 1) it aims at not only distilling the most representative information from a given paragraph but also getting rid of the general background information (probably caused by stop or function words); 2) it also targets at preserving semantic locality in the learned low-dimensional embedding space, so as to deduce an informative and discriminative low-dimensional vector representation for a given paragraph. More formally, the proposed method is divided into two major mechanisms: an essence vector (EV) model and a locality preserving (LP) model.

3.1. The Essence Vector Model

In order to realize the first idea, we begin with an assumption that each paragraph can be assembled by two components: the paragraph specific information and the general background information [22]. The assumption also holds in the low-dimensional representation space. Accordingly, the essence vector (EV) model consists of three modules: a paragraph encoder $f(\cdot)$, which can automatically infer the desired low-dimensional vector representation by considering only the paragraph-specific information; a background encoder $g(\cdot)$, which is used to map the general background information into a low-dimensional representation; and a decoder $h(\cdot)$ that can reconstruct the original paragraph by combining the paragraph representation and the background representation.

Formally, given a set of training paragraphs $\{D_1, \cdots, D_t, \cdots, D_{\mathrm{T}}\}$, 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 in the vocabulary V, and the vector is normalized to unit-sum. Then, a paragraph encoder is applied to extract the most specific information from the paragraph and encapsulate it into a low-dimensional vector representation:

$$f(P_{D_t}) = v_{D_t}. \quad (4)$$

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 is used to compress the general background information into a low-dimensional vector representation:

$$g(P_{BG}) = v_{BG}. \quad (5)$$

Both $f(\cdot)$ and $g(\cdot)$ are fully connected multilayer neural networks with different model parameters θ_f and θ_g, respectively. It is worthy to note that the model structures of $f(\cdot)$ and $g(\cdot)$ can be either the same or different. Since each learned paragraph representation v_{D_t} only contains the most informative/discriminative part of P_{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 $h(\cdot)$:

$$h(\alpha_{D_t} \cdot v_{D_t} + (1 - \alpha_{D_t}) \cdot v_{BG}) = P'_{D_t}, \quad (6)$$

where $h(\cdot)$ is also a fully connected multilayer neural network with parameter θ_h, and the interpolation weight can be determined by an attention function $q(\cdot,\cdot)$:

$$\alpha_{D_t} = q(v_{D_t}, v_{BG}). \quad (7)$$

The attention function can be realized by a trainable network or a simple linear/non-linear function. Further, to ensure the quality of

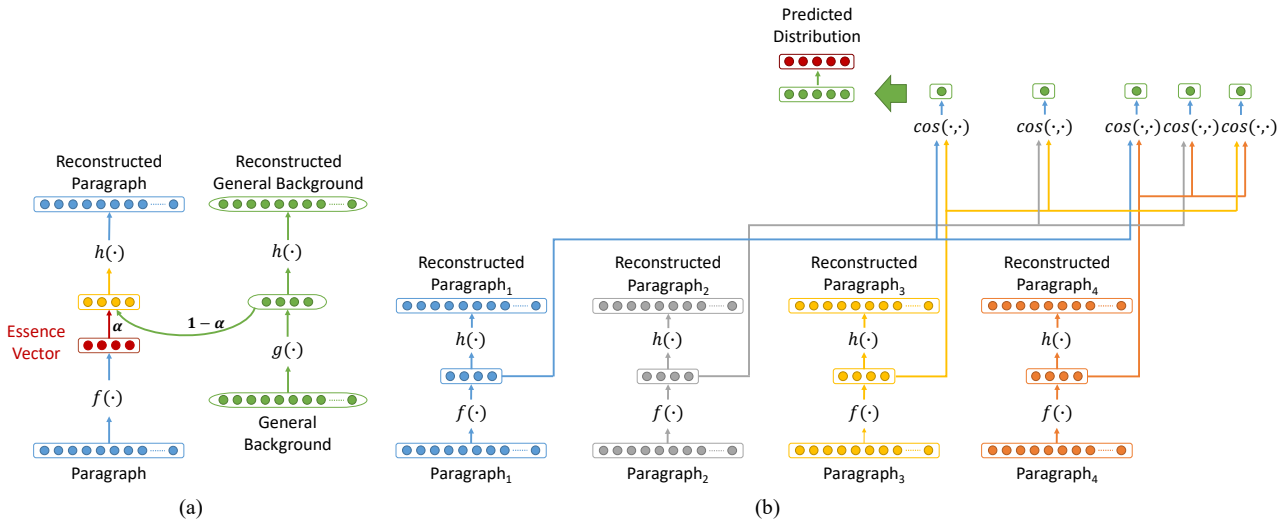

Figure1. *Running examples for (a) the essence vector model and (b) the locality preserving model.*

the learned background representation v_{BG}, it should also be able to be mapped back to P_{BG} by $h(\cdot)$:

$$h(v_{BG}) = P'_{BG}. \tag{8}$$

In a nutshell, the objective function of the EV model is to minimize the total KL-divergence measure:

$$\min_{\theta_f,\theta_g,\theta_h} \sum_{t=1}^{T} \left(P_{D_t} \log \frac{P_{D_t}}{P'_{D_t}} + P_{BG} \log \frac{P_{BG}}{P'_{BG}} \right). \tag{9}$$

Fig. 1(a) illustrates the architecture of the EV model employed in the proposed paragraph embedding method.

3.2. The Locality Preserving Model

For the second modeling concept, similar to the EV model, we begin with a paragraph encoder $f(\cdot)$ and a decoder $h(\cdot)$ that can reconstruct the original paragraph by referring to the deduced paragraph representation. Again, both $f(\cdot)$ and $h(\cdot)$ are fully connected multilayer neural networks with different model parameters θ_f and θ_h, respectively. v_{D_i} is the learned paragraph embedding for paragraph D_i. Next, from the local invariance perspective [23], which is a well-established principle adopted in manifold learning techniques [23-25], our idea is that the "nearby" paragraphs are likely to have similar embeddings in the learned low-dimensional space. More formally, at each training step, we first randomly select a set of nearby paragraphs, denoted by $\mathbf{D}^+$, and a set of paragraphs that occur far apart as negative examples, denoted by $\mathbf{D}^-$. Subsequently, for each pair of nearby paragraphs $D_i \in \mathbf{D}^+$ and $D_j \in \mathbf{D}^+$, a likelihood function can be defined to indicate how likely the two paragraphs are adjacent to each other:

$$P'(D_i, D_j) = \frac{e^{cos(v_{D_i}, v_{D_j})}}{\sum_{D_{i'} \in \mathbf{D}^+} \sum_{\substack{D_{j'} \in (\mathbf{D}^+ \cup \mathbf{D}^-) \\ s.t.\ D_{i'} \neq D_{j'}}} e^{cos(v_{D_{i'}}, v_{D_{j'}})}}. \tag{10}$$

It is worthy to note that we only ensure that any pairs of paragraphs in $\mathbf{D}^+$ are nearby to each other and that a paragraph in $\mathbf{D}^-$ should be located far away from a paragraph in $\mathbf{D}^+$, but we cannot guarantee the relationship (i.e., being neighbors or not) between each pair of paragraphs in $\mathbf{D}^-$. Therefore, we exclude the pairs of paragraphs in $\mathbf{D}^-$ in the denominator. Another notable issue is that we do not consider the order in constructing the pair of paragraphs, i.e., (D_i, D_j)=(D_j, D_i), thus they will be counted only once. Moreover, "nearby" can be determined by a variety of characteristics pertaining to the associated tasks. In the context of NLP, the semantic distance is a reasonable and intuitive manner. Detailed implementations and experimentations will be described in Section 4. To recap, given a set of training instances $\{(\mathbf{D}_1^+, \mathbf{D}_1^-), \cdots, (\mathbf{D}_t^+, \mathbf{D}_t^-), \cdots, (\mathbf{D}_T^+, \mathbf{D}_T^-)\}$, the objective function of the locality preserving (LP) model is to minimize the total KL-divergence measure:

$$\min_{\theta_f,\theta_h} \sum_{t=1}^{T} \left(\left(\sum_{D_i \in \mathbf{D}_t^+} \sum_{\substack{D_j \in (\mathbf{D}_t^+ \cup \mathbf{D}_t^-) \\ s.t.\ D_i \neq D_j}} P(D_i, D_j) \log \frac{P(D_i, D_j)}{P'(D_i, D_j)} \right) + \left(\sum_{D_i \in (\mathbf{D}_t^+ \cup \mathbf{D}_t^-)} P_{D_i} \log \frac{P_{D_i}}{P'_{D_i}} \right) \right), \tag{11}$$

where $P(D_i, D_j)$ is the desired distribution:

$$P(D_i, D_j) = \begin{cases} \frac{1}{C_2^{|\mathbf{D}^+|}} & ,if\ D_i \in \mathbf{D}^+, D_j \in \mathbf{D}^+, and\ D_i \neq D_j \\ 0 & ,otherwise \end{cases}, \tag{12}$$

where $|\mathbf{D}^+|$ denotes the number of paragraphs in $\mathbf{D}^+$. In implementation, the cosine similarities between paragraphs can be calculated first, and then the similarity scores are concatenated to a vector. After that, a softmax layer is stacked upon the vector (i.e., the concatenated vector is treated as an input) to obtain the final predicted distribution. Fig. 1(b) illustrates an example, where paragraphs 1 and 2 belong to $\mathbf{D}^-$, and paragraphs 3 and 4 belong to $\mathbf{D}^+$. In this example, the desired distribution is (0,0,0,0,1).

3.3. The Locality-Preserving Essence Vector Model

The essence vector model and the locality preserving model can be combined as a locality-preserving essence vector (LPEV) model. In this way, the resulting paragraph representation not only contains the most representative information from the paragraph (done by EV) but also preserves important semantic locality information (done by LP). To put everything together, given a set of training instances $\{(\mathbf{D}_1^+, \mathbf{D}_1^-), \cdots, (\mathbf{D}_t^+, \mathbf{D}_t^-), \cdots, (\mathbf{D}_T^+, \mathbf{D}_T^-)\}$, LPEV has three sets of parameters θ_f, θ_g and θ_h, and the objective function becomes:

$$\min_{\theta_f,\theta_g,\theta_h} \sum_{t=1}^{T} \left(\left(\sum_{D_i \in \mathbf{D}_t^+} \sum_{\substack{D_j \in (\mathbf{D}_t^+ \cup \mathbf{D}_t^-) \\ s.t.\ D_i \neq D_j}} P(D_i, D_j) \log \frac{P(D_i, D_j)}{P'(D_i, D_j)} \right) + \sum_{D_i \in (\mathbf{D}_t^+ \cup \mathbf{D}_t^-)} \left(P_{D_i} \log \frac{P_{D_i}}{P'_{D_i}} + P_{BG} \log \frac{P_{BG}}{P'_{BG}} \right) \right). \tag{13}$$

The activation function used in the proposed models (including EV,

LP, and LPEV) is the hyperbolic tangent, except that the output layers in the decoder $h(\cdot)$ and LP are the softmax function [26], and the Adam [27] is employed to solve the optimization problem.

4. EXPERIMENTAL SETUP & RESULTS

4.1. Experimental Setup

We used the Topic Detection and Tracking collection (TDT-2) [28] for our spoken document retrieval experiments. The Mandarin news stories from Voice of America news broadcasts were used as the spoken documents. All news stories were exhaustively tagged with event-based topic labels, which served as the relevance judgments for performance evaluation. The average word error rate (WER) obtained for the spoken documents is about 35% [29]. The Chinese news stories from Xinhua News Agency were used as our test queries. More specifically, in the following experiments, we will either use a whole news story as a "long query," or merely extract the title field from a news story as a "short query." The retrieval performance is evaluated with the commonly-used non-interpolated mean average precision (MAP) [30] metric.

4.2. Experimental Results

To begin with, we investigate the utilities of the vector space model (VSM) and two classic paragraph embedding methods (i.e., DM and DBOW) for SDR. The results are shown in Table 1, where "Text Documents" denotes the results obtained based on the manual transcripts of spoken documents and "Spoken Documents" denotes the results using the speech recognition transcripts that may contain recognition errors. Inspection of Table 1 reveals two noteworthy points. First, the performance gap between the retrieval using the manual transcripts and the recognition transcripts is about 0.05 in terms of MAP, such degradation is apparently less pronounced as compared to the high WER of spoken documents. Second, both of the two celebrated paragraph embedding methods outperform VSM in most cases, and DBOW consistently outperforms DM by a large margin when using either text documents or spoken documents. The results also evidence the success of employing representation learning techniques for SDR.

Next, we evaluate the proposed framework. In different NLP-related applications, "nearby paragraphs" may have different definitions. For (spoken) document retrieval, a reasonable and straightforward definition of "nearby paragraphs" refers to semantically related paragraphs. In this paper, we explore two ways to collect such information for training the LP and LPEV models.

In the first way, we perform experiments that simulate a scenario in which a set of training query exemplars and the corresponding query-document relevance information (i.e., the click-through information that to some extent reflects users' relative preferences of document relevance) can be utilized. 819 training query exemplars with the corresponding query-document relevance information are compiled. Based on that, a set of training instances is generated by 1) randomly selecting a training query, 2) picking two relevant (clicked) documents to the query to be $\mathbf{D}^+$, and 3) choosing an irrelevant document to the query to form $\mathbf{D}^-$. Thus, the desired distribution is (0,0,1) (cf. Section 3.2 and Fig. 1(b)). The results are listed in Table 2. It is obvious that LP outperforms EV in all cases, and the hybrid model, LPEV, achieves the best results in most cases as expected. One possible reason is that EV only focuses on distilling the most representative information from a paragraph and getting rid of the general background information, but does not leverage the click-through information to benefit the performance gains. LPEV outperforms EV and LP because it inherits advantages from both models. Comparing Tables 2 and 1, we can see all the new paragraph embedding methods outperform the baseline methods.

In the second way, we evaluate LP and LPEV under the condition that query-document relevance information of the training query exemplars is not readily available. A natural solution is to conduct a run of retrieval and take the top-ranked documents in response to each training query exemplar as the pseudo-relevant documents of the query for training the models. Such strategy is known as the pseudo-relevance feedback process [31]. In our experiments, the top 3 retrieved documents for each training query are treated as relevant documents. Thereupon, a set of training instances is generated in the same way as mentioned in the previous set of experiment. The results are listed in Table 3. There are some interesting observations. First, LPEV can still outperform EV and LP in the text document case as expected, while the superiority is not as obvious in the case of using spoken documents. Second, when compared to Table 2, the results signal that their associated performance heavily relies on the information used in model training. Third, although all of the models compared in Table 3 outperform VSM and DM, they only achieve comparable results with DBOW (cf. Table 1). One possible reason might be that the pseudo-relevant documents are obtained by using the unigram language model [11] method, which might not offer sufficient/suitable semantic locality information.

Table 1. *Retrieval results achieved by baseline systems for both short and long queries.*

	Text Documents		Spoken Documents	
	Long	Short	Long	Short
VSM	0.548	0.338	0.484	0.273
DM	0.558	0.344	0.484	0.302
DBOW	0.579	0.362	0.540	0.345

Table 2. *Retrieval results achieved by the proposed EV, LP, and LPEV models for both short and long queries with click-through information.*

	Text Documents		Spoken Documents	
	Long	Short	Long	Short
EV	0.571	0.382	0.518	0.364
LP	0.620	0.410	**0.567**	0.381
LPEV	**0.684**	**0.418**	0.556	**0.390**

Table 3. *Retrieval results achieved by the proposed EV, LP, and LPEV models for both short and long queries with semantic information obtained by pseudo-relevance feedback process.*

	Text Documents		Spoken Documents	
	Long	Short	Long	Short
EV	0.571	0.382	0.518	**0.364**
LP	0.573	0.383	0.507	0.339
LPEV	**0.580**	**0.392**	**0.533**	0.339

5. CONCLUSIONS

In this paper, we have proposed a novel paragraph embedding framework, which is embodied with the essence vector (EV) model, the locality preserving (LP) model, and the locality-preserving essence vector (LPEV) model. We also made a step forward to evaluate these models on a representative SDR task. Experimental results demonstrate that the proposed framework is the most robust in relation to the strong baselines compared in the paper, thereby indicating the potential of the new paragraph embedding framework. For future work, we will first focus on pairing the proposed framework with more other state-of-the-art retrieval methods. Moreover, we will explore other effective ways to integrate extra cues, such as syntactic information, into the proposed framework. Furthermore, we also plan to extend the applications to language modeling and among others.

6. ACKNOWLEDGEMENT

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

7. REFERENCES

[1] C. Chelba, T. J. Hazen, and Murat Saraclar, "Retrieval and browsing of spoken content," *IEEE Signal Processing Magazine*, 25(3), pp. 39–49, 2008.

[2] L. S. Lee and B. Chen, "Spoken document understanding and organization," *IEEE Signal Processing Magazine*, 22(5), pp. 42–60, 2005.

[3] C. L. Huang, B. Ma, H. Li, and C. H. Wu, "Speech indexing using semantic context inference," in *Proc. of INTERSPEECH*, pp. 717–720, 2011.

[4] W. Naptali, M. Tsuchiya, and S. Nakagawa, "Context dependent class language model based on word co-occurrence matrix in LSA framework for speech recognition," in *Proc. of ACS*, pp. 275–280, 2008.

[5] W. Naptali, M. Tsuchiya, and S. Nakagawa, "Word co-occurrence matrix and context dependent class in LSA based language model for speech recognition," *International Journal of Computers*, pp. 85–95, 2009.

[6] S. Deerwester, S. Dumais, G. Furnas, T. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *Journal of the American Society of Information Science*, 41(6), pp. 391–407, 1990.

[7] C. D. Manning, P. Raghavan and H. Schtze, *Introduction to Information Retrieval*, New York: Cambridge University Press, 2008.

[8] G. Salton , A. Wong , and C. S. Yang, "A vector space model for automatic indexing," *Communications of the ACM*, 18(11), pp. 613–620, Nov. 1975

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

[10] J. M. Ponte and W. B. Croft, "A language modeling approach to information retrieval," in *Proc. of SIGIR*, pp. 275–281, 1998.

[11] W. B. Croft and J. Lafferty (eds.), "Language modeling for information retrieval," Kluwer International Series on Information Retrieval, Volume 13, Kluwer Academic Publishers, 2003.

[12] Y. Bengio, R. Ducharme, P. Vincent, and C. Jauvin, "A neural probabilistic language model," *Journal of Machine Learning Research* (3), pp. 1137–1155, 2003.

[13] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *Proc. of ICLR*, pp. 1–12, 2013.

[14] J. Pennington, R. Socher, and C. Manning, "GloVe: Global vector for word representation," in *Proc. of EMNLP*, pp. 1532–1543, 2014.

[15] D. Tang, F. Wei, N. Yang, M. Zhou, T. Liu, and B. Qin, "Learning sentiment-specific word embedding for twitter sentiment classification," in *Proc. of ACL*, pp. 1555–1565, 2014.

[16] R. Collobert and J. Weston, "A unified architecture for natural language processing: deep neural networks with multitask learning," in *Proc. of ICML*, pp. 160–167, 2008

[17] K.-Y. Chen, S.-H. Liu, H.-M. Wang, B. Chen, and H.-H. Chen, "Leveraging word embeddings for spoken document summarization," in *Proc. of INTERSPEECH*, 2015.

[18] Q. Le and T. Mikolov, "Distributed representations of sentences and documents," in *Proc. of ICML*, pp. 1188–1196, 2014.

[19] K.-Y. Chen, H.-S. Lee, H.-M. Wang, B. Chen, and H. H. Chen, "I-vector based language modeling for spoken document retrieval," in *Proc. of ICASSP*, pp. 7083–7088, 2014

[20] P.-S. Huang, X. He, J. Gao, and L. Deng, "Learning deep structured semantic models for web search using clickthrough data," in *Proc. of CIKM*, pp. 2333–2338, 2013.

[21] H. Palangi, L. Deng, Y. Shen, J. Gao, X. He, J. Chen, X. Song, and R. Ward, "Deep sentence embedding using the long short term memory network: analysis and application to information retrieval," in *Proc. of arXiv*, 2015.

[22] K.-Y. Chen, S.-H. Liu, B. Chen, and H.-M. Wang, "Learning to distill: the essence vector modeling framework," in *Proc. of Coling*, pp. 358–368, 2016.

[23] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering," *Advances in Neural Information Processing Systems 14*, pp. 585–591, MIT Press, 2001.

[24] S. Roweis and L. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.

[25] J. Tenenbaum, V. Silva, and J. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.

[26] I. Goodfellow, Y. Bengio, and A. Courille, *Deep Learning*, Cambridge, MA: MIT Press, 2016.

[27] D. P. Kingma and J. L. Ba, "ADAM: A method for stochastic optimization," in *Proc. of ICLR*, 2015.

[28] LDC, "Project topic detection and tracking," *Linguistic Data Consortium*, 2000.

[29] H. Meng, S. Khudanpur, G. Levow, D. Oard, and H.-M. Wang, "Mandarin–English information (MEI): investigating translingual speech retrieval," *Computer Speech and Language*, 18(2), pp. 163–179, April 2004.

[30] R. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieval: The Concepts and Technology behind Search*, ACM Press, 2011.

[31] K.-Y. Chen, S.-H. Liu, B. Chen, H.-M. Wang, and H.-H. Chen, "Exploring the use of unsupervised query modeling techniques for speech recognition and summarization," *Speech Communication*, pp. 49–59, 2016.