# NEURAL RELEVANCE-AWARE QUERY MODELING FOR SPOKEN DOCUMENT RETRIEVAL

*Tien-Hong Lo*[1], *Ying-Wen Chen*[1], *Kuan-Yu Chen*[2], *Hsin-Min Wang*[3], *and Berlin Chen*[1]

[1] National Taiwan Normal University, Taiwan
[2] National Taiwan University of Science and Technology, Taiwan
[3] Academia Sinica, Taiwan

E-mail: {60547047s, cliffchen, berlin}@ntnu.edu.tw, {kychen, whm}@iis.sinica.edu.tw

## ABSTRACT

Spoken document retrieval (SDR) is becoming a much-needed application due to that unprecedented volumes of audio-visual media have been made available in our daily life. As far as we are aware, most of the wide variety of SDR methods mainly focus on exploring robust indexing and effective retrieval methods to quantify the relevance degree between a pair of query and document. However, similar to information retrieval (IR), a fundamental challenge facing SDR is that a query is usually too short to convey a user's information need, such that a retrieval system cannot always achieve prospective efficacy when with the existing retrieval methods. In order to further boost retrieval performance, several studies turn their attention to reformulating the original query by leveraging an online pseudo-relevance feedback (PRF) process, which often comes at the price of taking significant time. Motivated by these observations, this paper presents a novel extension of the general line of SDR research and its contribution is at least two-fold. First, building on neural network-based techniques, we put forward a neural relevance-aware query modeling (NRM) framework, which is designed to not only infer a discriminative query language model automatically for a given query, but also get around the time-consuming PRF process. Second, the utility of the methods instantiated from our proposed framework and several widely-used retrieval methods are extensively analyzed and compared on a standard SDR task, which suggests the superiority of our methods.

***Index Terms*****—** Spoken document retrieval, relevance feedback, query language models, neural networks

## 1. INTRODUCTION

Owing to vast amounts of multimedia associated with speech content continuously filling in our daily life, spoken document retrieval (SDR) has become an attractive research field over the past two decades [1-4]. A significant body of research has been devoted to developing effective retrieval methods with successful applications to various SDR tasks, such as the vector space model [5], the Okapi BM25 model [6] and the topic models [7], among others. Especially, in the recent past an emerging paradigm has been to employ a statistical language modeling (LM) approach for information retrieval (IR) as well as SDR. This paradigm has also garnered much attention due to its simplicity and clear probabilistic meaning, as well as state-of-the-art performance [8-10]. For the idea to work, a given text or spoken document can be framed as a generative model composed of a mixture of multinomial (*n*-gram) distributions for observing a query, while the query is regarded as an observation, expressed by a sequence of words. Accordingly, documents can be ranked according to their likelihoods of generating the query, namely the query-likelihood measure (QLM) [11]. Another popular formulation is the Kullback-Leibler divergence measure (KLM) [12], where both the query and the documents are represented by a unigram language model, respectively. The relevance degree between a query and a document is recast as the divergence distance between the two respective unigram models.

More recently, deep neural network-based retrieval methods have enjoyed considerable popularity in a number of IR and SDR tasks. Most of the studies on such methods concentrate exclusively on designing methods that can be employed to calculate the relevance degree between a query and a document, building on a diverse range of neural network architectures and training criteria [13, 14]. One thing to note is that inputs to these methods are primarily based surface statistics such as word proximity or co-occurrence counts, while the associated training objectives usually aim to distinguish between relevant and irrelevant documents in a pairwise manner. Among them, the deep structure semantic model (DSSM) [15, 16] and the locality preserving essence vector model (LPEV) [17] are two representatives. In addition, there are been efforts intended to infer dense vector representations for both queries and documents in an unsupervised manner. As such, the similarity degree between a pair of query and document can be readily quantified by using the existing ranking mechanisms based on the learned representations. Celebrated methods developed in this vein include the word embedding-based methods (e.g., the skip-gram model [18] and the continuous bag-of-words model [19]), and the paragraph embedding methods (e.g., the distributed memory model [20], the distributed bag-of-words model [20, 21], the skip-thought vector [22] and the essence vector model [23]), just to name a few.

However, one critical issue facing IR and SDR is that the input text or spoken query is usually too short to carry the information need of a user. In order to mitigate the problem, a promising strategy is to reformulate the query representation with extra statistics so as to boost the retrieval performance [12, 24-26]. The query reformulation methods devised following the line of research can be grouped into two distinct classes. One is to leverage external resources, such as Wikipedia or WordNet, to expand and reorganize the original query. The other is to reformulate the original query by referring to a small set of feedback documents locally collected from an initial round of retrieval, i.e., the so-called pseudo-relevance feedback (PRF) process. Since the former requires more sophisticated natural language processing techniques, including semantic representation and inference, as well as natural language generation, most efforts have been concentrated on launching the query reformulation methods by using the top-ranked feedback documents locally obtained from PRF [11]. Popular LM-based methods that employ the PRF process include, but are not limited to, the relevance model (RM) [24], the simple mixture model (SMM) [12] and the significant words model [25, 27]. Although several studies have confirmed the effectiveness of these query reformulation methods on a broad range of IR and SDR tasks, the time-consuming problem still makes them unappealing for realistic applications.

Motivated by the above considerations, this paper strives to develop a novel query representation learning approach with neural network-based techniques, named the neural relevance-aware query modeling framework, which is designed to not only infer a more discriminative query language model for a given query automatically, but also avoid the time-consuming online PRF process. By doing so, the SDR system can thus perform more effectively and efficiently. The remainder of this paper is structured as follows. In Section 2, we briefly review the foundamentals of the classic LM-based query reformulation methods. Next, in Section 3, we shed light on the proposed neural relevance-aware query modeling framework. After that, the experimental settings and a series of retrieval experiments are presented in Sections 4, respectively. Finally, Section 5 concludes this paper and discusses avenues for future work.

## 2. RELATED WORK

Due to the fact that a query usually consists of only a few words, the true query language model $P(w|Q)$ of a query $Q$ for predicting an arbitrary word $w$ might not be accurately estimated by the simple maximum likelihood (ML) estimator. With the alleviation of this deficiency as motivation, there are several studies devoted to estimating a more accurate query representation, saying that it can be approached through a pseudo-relevance feedback (PRF) process. Such an integration have proven effective for query reformulation. Its success, however, depends largely on the assumption that the set of top-ranked feedback documents obtained from an initial round of retrieval $\mathbf{D}_F = \{D_1, \cdots, D_r, \cdots, D_{|\mathbf{D}_F|}\}$ are relevant and can be used to estimate a more accurate query language model. Representative methods include, among others, the relevance model (RM), the simple mixture model (SMM) and the significant words model (SWM).

### 2.1. Relevance Model (RM)

Under the notion of relevance modeling [24], each query $Q$ is assumed to be associated with an unknown relevance class $R_Q$, and documents that are relevant to the semantic content expressed in $Q$ are also samples drawn from the relevance class $R_Q$. Nevertheless, in reality, since there is no prior knowledge about $R_Q$, we may instead use the top-ranked feedback documents $\mathbf{D}_F$ obtained from PRF to approximate the relevance class $R_Q$. The corresponding relevance model (RM), on the grounds of a multinomial view of $R_Q$, can be estimated using the following equation [4, 24]:

$$P_{\mathrm{RM}}(w|Q) = \frac{\sum_{D_r \in \mathbf{D}_F} P(D_r) P(w|D_r) \prod_{w' \in Q} P(w'|D_r)}{\sum_{D'_r \in \mathbf{D}_F} P(D'_r) \prod_{w'' \in Q} P(w''|D'_r)}, \quad (1)$$

where the prior probability $P(D_r)$ of each document can be simply kept uniform, while the document language model $P(w|D_r)$ is estimated with the ML estimator on the basis of the occurrence count of $w$ in each respective document.

### 2.2. Simple Mixture Model (SMM)

Another perspective of estimating an enhanced query model with the feedback documents is the simple mixture model (SMM) [12], which assumes that words in $\mathbf{D}_F$ are drawn from a two-component mixture model: one is the query-specific topic model $P_{\mathrm{SMM}}(w|Q)$; the other is a general background language model $P_{\mathrm{BG}}(w)$. This way, the SMM model $P_{\mathrm{SMM}}(w|Q)$ can be estimated by maximizing the likelihood of all words in the feedback documents:

$$L = \prod_{D \in \boldsymbol{D}_F} \prod_{w \in V} [\alpha \cdot P_{\mathrm{SMM}}(w|Q) + (1-\alpha) \cdot P_{\mathrm{BG}}(w)]^{c(w,D)}, \quad (2)$$

where $\alpha$ is a pre-defined weighting parameter used to control the degree of reliance between $P_{\mathrm{SMM}}(w|Q)$ and $P_{\mathrm{BG}}(w)$. Such estimation will enable more specific words (i.e., words in $\mathbf{D}_F$ that are not well-explained by the general background language model) to receive more probability mass, thereby leading to a more discriminative query model $P_{\mathrm{SMM}}(w|Q)$. Simply put, it is anticipated that the SMM model can extract useful word usage cues from $\mathbf{D}_F$, which are not only probably relevant to the query $Q$, but also external to those already well captured by the general background language model.

### 2.3. Significant Words Model (SWM)

Inspired from the Luhn's theory [28] and SMM, the significant words model (SWM) [25, 27] explores to estimate an accurate query language model by parsimonizing the estimation toward not only the generally common words, but also the too specific words reoccurring concentratedly in only few feedback documents. More formally, SWM assumes words in each feedback document are samples drawn from a three-component mixture model: the general background language model $P_{\mathrm{BG}}(w)$, the specific language model $P_{\mathrm{S}}(w|Q)$ and the desired SWM model $P_{\mathrm{SW}}(w|Q)$. This way, the probability of a word occurring in a feedback document $D$ can be defined by

$$P(w|D) = \alpha \cdot P_{\mathrm{BG}}(w) + \beta \cdot P_{\mathrm{S}}(w|Q) + (1 - \alpha - \beta) \cdot P_{\mathrm{SW}}(w|Q), \quad (3)$$

where $\alpha$ and $\beta$ are tuneable parameters used to modulate the contributions between $P_{\mathrm{BG}}(w)$, $P_{\mathrm{S}}(w|Q)$ and $P_{\mathrm{SW}}(w|Q)$. In practice, the general background language model is employed to represent the frequent words in general, which can be estimated from a large collection of corpora a priori. The specific language model $P_{\mathrm{S}}(w|Q)$ is formulated to capture those words that occur repetitively in a very small portion of feedback documents for the query $Q$. Accordingly, the SWM model $P_{\mathrm{SW}}(w|Q)$ can be estimated by maximizing the likelihood over all the feedback documents with the expectation-maximization (EM) algorithm [29].

## 3. THE NEURAL RELEVANCE-AWARE QUERY LANGUAGE MODELING FRAMEWORK

Although the aforementioned well-practiced methods, which aim to reformulate the original query language model through a pseudo-relevance feedback process, have enjoyed much success in several IR and SDR tasks, they nevertheless pose an undesirable side effect on the efficiency of retrieval, namely the time-consuming problem [11]. This is because that the enhanced query language model is estimated based on a set of top-ranked feedback documents, which comes at the cost of performing an additional round of retrieval at query time. In order to mitigate such a deficiency, we put forward a neural relevance-aware query modeling (NRM) framework on top of neural network-based techniques, which not only can concentrate on reformulating the original query language model, but also dispense with the time-consuming PRF process.

### 3.1. Modeling Relevance for Query Language Model

To turn the idea into reality, we first revisit the fundamentals of query reformulation. The primary objective in many query reformulation methods is to model the notion of relevance. For example, RM explores a systematic way to approximate the relevance class, while SMM and SWLM leverage background information (and additional document-specific information) to deduce the homogeneous concept among the feedback documents. In this study, we set out to model the notion of relevance by neural networks.

More formally, given a set of training queries $\mathbf{Q} = \{Q_1, \cdots, Q_t, \cdots, Q_T\}$ and their corresponding query-document relevance information $\mathbf{R} = \{R_1, \cdots, R_t, \cdots, R_T\}$, in order to modulate the effect of different lengths of queries, each query is first represented by a high-dimensional bag-of-words vector $P_{Q_t} \in \mathbb{R}^{|V|}$, where each vector element corresponds to the frequency count of a specific word (term) occurring in $Q_t$ and $|V|$ denotes the vocabulary $V$. The vector $P_{Q_t}$ is further normalized to unit-sum. After that, a query encoder $f(\cdot)$ is applied to encapsulate the original query into a low-dimensional vector representation:

$$f(P_{Q_t}) = v_{Q_t}, \quad (4)$$

where $f(\cdot)$ is a feed-forward fully-connected neural network used in this paper. Since the ultimate goal of the proposed NRM framework is to derive an enhanced query language model, an intuitive idea is to infer a set of word embeddings paired with the learned query representation and in turn build a query language model based on them. As far as we are aware, most classic word embedding methods, such as the skip-gram model and the continuous bag-of-words model, deduce the word embeddings based only on the local proximity of words occurring in the training corpus. As a result, the learned word embeddings can preserve the contextual and structural information well; they also have shown empirical success in many NLP-related tasks like analog analysis [30] and sentiment prediction [31]. However, several studies have indicated that two words that have their word embeddings close to each other may convey opposite meanings (such as “good” and “bad”), just because they often appear in similar contexts [32, 33]. Thus, reformulating a query language model with these classic word embedding methods could mislead the original user’s information need, resulting in misty retrieval results. The above reasoning motivates us to learn a new set of word embeddings which is more suitable for modeling relevance in query reformulation. In order to crystallize the notion, we stack a feed-forward fully-connected layer $g(\cdot)$ on top of the query encoder, followed by an output layer that is equipped with a softmax function. The parameters of the feed-forward layer $g(\cdot)$ is a weight matrix $\mathbf{W} \in \mathbb{R}^{k \times |V|}$, where $k$ is the size of the learned query representation and $|V|$ denotes the size of the vocabulary. Consequently, the neural relevance-aware query language model can be expressed by

$$P_{\mathrm{NRM}}(w|Q_t) = g\left(f(P_{Q_t})\right) = \frac{\exp(v_{Q_t} \cdot v_w)}{\sum_{w' \in V} \exp(v_{Q_t} \cdot v_{w'})}, \quad (5)$$

where $v_w$ is the $w^{th}$ column in $\mathbf{W}$ and denotes the word embedding for word $w$. Finally, to capture the notion of relevance, the training objective is formulated to search a set of model parameters that maximize the likelihood of all the training instances:

$$L = \prod_{t=1}^{\mathrm{T}} \prod_{w \in V} P(w|R_t) log P_{\mathrm{NRM}}(w|Q_t), \quad (6)$$

where $P(w|R_t)$ is the desired relevance distribution for each training query $Q_t$ (we will discuss this distribution in more detail in the next subsection). To put everything together, the proposed neural relevance-aware query language modeling framework will encompass two components: a query encoder $f(\cdot)$ for inferring a low-dimensional query representation and a set of word embeddings $\mathbf{W}$ for representing the notion of relevance for a given query and an arbitrary word in the vocabulary.

### 3.2. Implementation Details

In the contexts of IR and SDR, a reasonable and straightforward definition of relevance refers to a user's information need. In this paper, we explore two ways to distill such information for training the associated model of the proposed NRM framework. In the first strategy, we simulate a scenario in which a set of training query exemplars and the corresponding query-document relevance information (e.g., the click-through information of a retrieval system that to some extent reflects users' relative preferences on document relevance) can be utilized. However, collecting suitable click-through information might be tedious and labor-consuming. In the second strategy, we therefore assume a scenario that query-document relevance information of the set of training query exemplars collected beforehand is not readily available. As such, a natural solution to this is to conduct a run of retrieval and then take the top-ranked documents in response to each training query exemplar as the pseudo-relevant documents. In turn, a set of training instances can be complied as well. Such a strategy in fact leverages the pseudo-relevance feedback process at training time. After we have obtained a set of training queries and their corresponding (pseudo) relevant documents, the desired relevance distribution (i.e., $P(w|R)$) of each training query can be approximated by using any of the existing language modeling methods for query reformulation (e.g. RM, SMM and SWM; *cf.* Section 2). The activation function used in the NRM-based model is a linear function, except that the output layer is equipped with the softmax function, while the Adam algorithm [34] is employed to solve the optimization problem. At test (query) time, a given query will first have its own low-dimensional embedding by feeding its original bag-of-words representation into the query encoder $f(\cdot)$, and subsequently the associated relevance-aware query language model can be readily obtained by taking this inferred query embedding as input to the feed-forward network $g(\cdot)$. After that, the KLM can be employed to determine the relevance degree between the neural relevance-aware query language model and each document language model for document ranking. It is worthwhile to note that the proposed NRM framework makes a novel step forward to replacing online query reformulation along with the pseudo-relevance feedback process by an offline query relevance modeling mechanism, which obviously introduce a substantial improvement in efficiency and practicality. To recap, the proposed neural relevance-aware query modeling framework can infer a discriminative language model automatically for any given query, and meanwhile exclude the time-consuming, online pseudo-relevance feedback process.

**Table 1.** Statistics of the TDT-2 collection (in characters).

| | | | | |
|---|---|---|---|---|
| # Spoken documents | 2,265 stories,<br>46.03 hours of audio | | | |
| # Distinct test queries | 16 Xinhua text stories<br>(Topics 20001~20096) | | | |
| | Min. | Max. | Med. | Mean |
| Doc. length | 23 | 4,841 | 153 | 287.1 |
| Length of test short query | 8 | 27 | 13 | 14.0 |
| Length of test long query | 183 | 2,623 | 329 | 532.9 |
| # Relevant documents per test query | 2 | 95 | 13 | 29.3 |

**Table 2.** Retrieval results of the 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 |
| KLM | 0.632 | 0.368 | 0.553 | 0.317 |
| LDA | 0.643 | 0.401 | 0.581 | 0.341 |
| RM | 0.702 | 0.421 | 0.612 | 0.369 |
| SMM | 0.686 | 0.485 | 0.570 | 0.414 |
| SWM | 0.717 | 0.556 | 0.669 | 0.491 |

## 4. EXPERIMENT SETUP & RESULTS

### 4.1. Experimental Setup

We used the Topic Detection and Tracking collection (TDT-2) for our experiments [35]. 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% [36]. 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." Table 1 shows some basic statistics of the TDT-2 collection. The retrieval performance is evaluated with the commonly-used non-interpolated mean average precision (MAP) metric [11, 37].

### 4.2. Experimental Results

To begin with, we compare the performance levels of several well-practiced retrieval methods for SDR, including the vector space-based methods and the language model-based methods. The corresponding results are summarized in Table 2. For the vector space-based methods, including the vector space model (VSM), the distributed memory model (DM) and the distributed bag-of-words model (DBOW), both queries and documents are represented by vectors, while the relevance degree is computed by the cosine similarity measure. In contrast to the vector space-based methods, KLM belongs to the language model-based approach, where the query and document language models are derived by the maximum likelihood estimator. LDA denotes latent Dirichlet allocation [38], for which each document language model is estimated based on a probabilistic topic modeling paradigm. In addition, three state-of-the-art LM-based query reformulation methods, namely the relevance model (RM), the simple mixture model (SMM) and the significant words model (SWM), are also compared. It is worthwhile to note that RM, SMM and SWM are respectively used to reformulate the original query language model through a online pseudo-relevance feedback process, while the document language models are derived by the ML estimator as in KLM. Several observations can be drawn here. First, both the celebrated paragraph embedding methods (i.e., DM and DBOW) outperform VSM, while DBOW consistently outdoes DM by a large margin, when being applied to either text documents (i.e., manual transcripts of documents) or spoken documents (i.e., speech recognition transcripts of documents). The results demonstrate the feasibility of using the neural network-based methods (especially for the unsupervised methods) for SDR. Second, KLM in general performs better than the vector space-based methods, including DM and DBOW. The results evidence that the LM-based methods introduce a promising family of efficient and effective mechanisms for SDR. Third, it is not surprising that LDA works better than KLM, whereas RM, SMM and SWM show superiority over LDA in most cases. The results confirm that deriving a more accurate query language model seems more effective than building more sophisticated document langue models. The reason might be that a document usually contains relatively sufficient statistics to estimate a reliable language model in relation to a short query.

Next, we evaluate the proposed NRM framework with two different scenarios (*cf.* Section 3.2). In the first scenarios, we assume the *click-through information* can be readily available, and thus a set of 819 training query exemplars with their corresponding query-document relevance information was compiled. The desired distribution (i.e., $P(w|R)$ ) was estimated by using either RM, SMM or SWM in this study. The results are shown in Table 3, where the best result within each column (corresponding to a specific evaluation condition) is type-set boldface. Moreover, the results of two recently proposed state-of-the-art neural network-based methods, namely the deep structure semantic model (DSSM) [15] and the locality preserving essence vector model (LPEV) [23], are also listed for comparison. DSSM exploits click-through data to train a discriminative deep neural network that can maximize the likelihood of clicked (relevant) documents given a query. Distinctively, LPEV aims to infer a vector representation for each input text (i.e., a query or a document), which not only contains the most representative information from the original input, but also preserves the semantic structure among training data set. At query time, both DSSM and LPEV use the cosine similarity measure to quantify the relevance degree between a query and a document with the learned embedding vectors.

**Table 3.** Retrieval results of the NRM-based models offline trained on click-through information.

| | Text Documents | | Spoken Documents | |
|---|---|---|---|---|
| | Long | Short | Long | Short |
| DSSM | 0.687 | 0.435 | **0.691** | 0.462 |
| LPEV | 0.684 | 0.418 | 0.556 | 0.390 |
| NRM (RM) | 0.702 | 0.562 | 0.620 | 0.504 |
| NRM (SMM) | 0.685 | **0.585** | 0.610 | 0.514 |
| NRM (SWM) | **0.730** | 0.563 | 0.686 | **0.547** |

**Table 4.** Retrieval results of the NRM-based models offline trained with pseudo-relevance feedback.

| | Text Documents | | Spoken Documents | |
|---|---|---|---|---|
| | Long | Short | Long | Short |
| DSSM | 0.407 | 0.248 | 0.353 | 0.235 |
| LPEV | 0.580 | 0.392 | 0.533 | 0.339 |
| NRM (RM) | **0.648** | 0.493 | 0.530 | 0.429 |
| NRM (SMM) | 0.636 | **0.494** | 0.567 | 0.426 |
| NRM (SWM) | **0.648** | 0.467 | **0.589** | **0.449** |

Several remarkable observations can be made from the results. First, DSSM consistently outperforms LPEV in all cases, which is because LPEV seeks only to learn a representation for a given query or document, whereas the training objective of DSSM aims at correctly determining the relevance degree between a query and document explicitly. Second, since SWM delivers better results than RM and SMM (*cf.* Table 2), the proposed NRM framework paired with SWM, as expected, can offer superiority performance than with RM and SMM in general. Second, comparing Tables 2 and 3, it is obvious that the proposed various NRM-based methods outperform both the vector space-based methods and the classic language model-based methods. Specifically, the various NRM-based methods can achieve better results than the existing query reformulation methods (i.e., RM, SMM and SWM), which may be attributed to the fact that the parameters of these NRM-based models are estimated beforehand from a set of click-through data (in contrast to RM, SMM and SWM that are online trained on pseudo-relevant documents obtained locally from PRF at query time). That is, an attractive characteristic of the proposed NRM framework is that it can be used to obtain a better query language model without additionally invoking an online, time-consuming pseudo-relevance feedback process at query time. Finally, we can also see that with supervised

**Table 5.** Retrieval results achieved by using a reformulated query language model, where the original NRM-based query language models are offline trained on click-through information or with pseudo-relevance feedback.

| | Click-through Information | | | | Pseudo-relevance Feedback | | | |
|---|---|---|---|---|---|---|---|---|
| | Text Documents | | Spoken Documents | | Text Documents | | Spoken Documents | |
| | Long | Short | Long | Short | Long | Short | Long | Short |
| NRM (RM) | 0.733 | 0.583 | 0.550 | 0.483 | **0.716** | **0.571** | 0.608 | **0.495** |
| NRM (SMM) | 0.735 | **0.600** | 0.603 | 0.529 | 0.666 | 0.539 | 0.590 | 0.406 |
| NRM (SWM) | **0.738** | **0.600** | **0.713** | **0.550** | 0.694 | 0.564 | **0.629** | 0.469 |

(click-through) training data, the three NRM-based models outperform LPEV by a large margin for all cases, while the combination of NRM with SWM, denoted by NRM (SWM), surpasses DSSM for most cases. The above results indeed confirm the effectiveness and capability of the proposed NRM framework for the SDR task studied here.

In the third set of experiments, we examine the second scenario that query-document relevance information of the training query exemplars is not readily available. A natural solution to this is to conduct a run of retrieval and take the top-ranked documents in response to each training query exemplar as its pseudo-relevant documents for estimating the query language models. In our experiments, the top 10 retrieved documents for each training query are treated as relevant ones. The target distribution (i.e., $P(w|R)$) for NRM is approximated by either RM, SMM or SWM again. The results are exhibited in Table 4 with some interesting findings. First, NRM paired with SWM can still achieve better results than with RM or SMM for the case of using spoken documents, while the superiority is less pronounced when using text documents instead. Second, with unsupervised training instances obtained from the pseudo-relevance feedback process, the various NRM-based models outdo DSSM and LPEV considerably, revealing the practical utility of our proposed NRM framework. Third, when compared to Table 3, it signals that the performance of the various NRM-based methods appears to heavily rely on the correctness of the relevance information employed for model training. Finally, when comparing Tables 3 and 4, LPEV seems more robust against the recognition errors than DSSM, since LPEV delivers superior results than DSSM in the second scenario.

In the last set of experiments, we turn to investigate the potential benefit of combining the various NRM-based methods with existing state-of-the-art LM-based query reformulation methods (i.e., RM, SMM or SWM in this study), albeit that the latter ones recourses to an extra online pseudo-relevance feedback process at query time. To do this, we first exploit the proposed various NRM-based methods to build the language model for an input query and in turn perform an initial round of retrieval with this query language model. The top-retrieved documents are treated as the pseudo-relevant documents for use in query reformulation. Then, a new query language model is estimated by using one of the existing query reformulation methods based on these top-retrieved documents, which in turn can be used to replace or work in combination with the original query language model for a second run of retrieval. The corresponding results are shown in Table 5; several noteworthy observations can be drawn here. First, in the first scenario where the NRM-based models were trained on click-through information, we find that NRM (SWM) still outperforms NRM (RM) and NRM (SMM) as before, whereas NRM (RM) instead achieves better results than NRM (SMM) and NRM (SWM) in the second scenario for most cases (i.e., the NRM-based models were trained with pseudo-relevance feedback). These results appear to indicate that though SWM is more sophisticated than RM and SMM (*cf.* Table 2), it is, however, too sensitive to the correctness on the relevance of the top-retrieved documents. Second, comparing Table 5 with Table 4, it signals that the reformulated query models based on an extra pseudo-relevance feedback process (*cf.* Table 5) can deliver further performance gains than the original ones (*cf.* Table 4). In summary, from the series of experiments discussed above, the proposed neural relevance-aware query modeling (NRM) framework seems to hold practical promise for SDR and IR related applications.

## 5. CONCLUSION AND OUTLOOK

In this paper, we have proposed a novel query modeling approach, named the neural relevance-aware query modeling framework, which can be employed to derive a discriminative query language model for SDR without the need of a tedious pseudo-relevance feedback process. We have also thoroughly evaluated the models stemming from this framework on a representative SDR task. Experimental results confirm the effectiveness of the proposed query language modeling framework in relation to the strong baselines compared in the paper, thereby indicating its potential for SDR and related applications. For future work, we will explore to couple the proposed framework with more sophisticated neural network-based techniques. We also plan to seek effective ways to integrate extra syntactic and prosodic information cues into the proposed framework. Furthermore, we will extend its applications to summarization [39, 40] and among others.

## 6. ACKNOWLEDGEMENTS

University Project" of National Taiwan Normal University (NTNU), sponsored by the Ministry of Education, Taiwan, and by the Ministry of Science and Technology, Taiwan, under Grants MOST 104-2911-I-003-301, MOST 104-2221-E-003-018-MY3 and MOST 105-2221-E-003-018-MY3.

## 6. REFERENCES

[1] C. Chelba, T. J. Hazen, and M. 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 *Proceedings of INTERSPEECH*, pp. 717–720, 2011.

[4] B. Chen, K.-Y. Chen, P.-N. Chen, and Y.-W. Chen, "Spoken document retrieval with unsupervised query modeling techniques," *IEEE Transactions on Audio, Speech, and Language Processing* 20(9), pp. 2602–2612, 2012.

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

[6] 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.

[7] D. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet Allocation" *Journal of Machine Learning Research*, 3(4–5), pp. 993–1022, 2013.

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

[9] F. Song and W. B. Croft, "A general language model for information retrieval," in *Proceedings of CIKM*, pp. 316–321, 1999.

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

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

[12] C. Zhai and J. Lafferty, "Model-based feedback in the language modeling approach to information retrieval," in *Proceedings of CIKM*, pp. 403–410, 2001.

[13] J.-F. Guo, Y. Fan, Q. Ai, and W. B. Croft, "A deep relevance matching model for ad-hoc retrieval," in *Proceedings of CIKM*, pp. 55–64, 2016.

[14] B. Mitra, F. Diaz, and N. Craswell, "Learning to match using local and distributed representations of text for web search," in *Proceedings of WWW*, pp. 1291–1299, 2017.

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

[16] Y. Shen, X. He, J. Gao, L. Deng, and G. Mesnil, "A latent semantic model with convolutional-pooling structure for information retrieval," in *Proceedings of CIKM*, pp. 101–110, 2014.

[17] K. Y. Chen, S. H. Liu, B. Chen, and H. M. Wang, "A locality-preserving essence vector modeling framework for spoken document retrieval," in *Proceedings of ICASSP*, pp. 5665–5669, 2017.

[18] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proceedings of NIPS*, pp. 3111–3119, 2013.

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

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

[21] 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 *Proceedings of ICASSP*, pp. 7083–7088, 2014.

[22] R. Kiros, Y. Zhu, R. Salakhutdinov, R. S. Zemel, A. Torralba, R. Urtasun, and S. Fidler, "Skip-thought vectors," arXiv:1506.06726, 2015.

[23] K. Y. Chen, S. H. Liu, B. Chen and H. M. Wang, "Learning to Distill: The Essence Vector Modeling Framework," in *Proceedings of COLING*, pp. 358–368, 2016.

[24] V. Lavrenko and W. Bruce Croft, "Relevance based language models," in *Proceedings of SIGIR*, pp. 120–127, 2001.

[25] M. Dehghani, H. Azarbonyad, J. Kamps, D. Hiemstra, and M. Marx, "Luhn revisited: significant words language models," in *Proceedings of CIKM*, pp. 1301–1310, 2016.

[26] 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*, 80, pp. 49–59, 2016.

[27] Y. W. Chen, K. Y. Chen, H. M. Wang, and B. Chen, "Exploring the use of significant words language modeling for spoken document retrieval," in *Proceedings of INTERSPEECH*, 2017.

[28] H. P. Luhn, "The automatic creation of literature abstracts," *IBM Journal of research and development*, 2(2), pp. 159–165, 1958.

[29] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the Royal Statistical Society*, 39(1), pp.1–38, 1977.

[30] J. Pennington, R. Socher, and C. D. Manning, "GloVe: Global vectors for word representation," in *Proceedings of EMNLP*, pp. 1532–1543, 2014.

[31] D. Y. Tang, F. Wei, B. Qin, M. Zhou, and T. Liu, "Building large-scale twitter-specific sentiment lexicon: a representation learning approach" in *Proceedings of COLING*, pp. 172–182, 2014.

[32] H. Zamani and W. B. Croft, "Relevance-based word embedding," arXiv:1705.03556, 2017.

[33] D. Y. Tang, F. Wei, N. Yang, M. Zhou, T. Liu, and B. Qin, "Learning sentiment-specific word embedding for twitter

sentiment classification," in *Proceedings of ACL*, pp. 1555–1565, 2014.

[34] D. Kingma and J. Ba, "ADAM: A method for stochastic optimization," in *Proceedings of ICLR*, 2015.

[35] LDC, *Project of Topic Detection and Tracking*, Linguistic Data Consortium, 2000.

[36] H. Meng, B. Chen, S. Khudanpur, G.-A. Levow, W.-K. Lo, D. Oard, P. Schone, K. Tang, H.-M. Wang, and J. Wang, "Mandarin–English information (MEI): investigating translingual speech retrieval," *Computer Speech and Language*, 18(2), pp. 163–179, 2004.

[37] R. Baeza-Yates and B. Ribeiro-Neto, *Modern information retrieval: the concepts and technology behind search*, ACM Press, 2011.

[38] X. Wei and W. Bruce Croft, "LDA-based document models for ad-hoc retrieval," in *Proceedings of SIGIR*, pp. 178–185, 2006.

[39] K. Y. Chen, S. H. Liu, B. Chen, H. M. Wang, E. E. Jan, W. L. Hsu, and H. H. Chen, "Extractive broadcast news summarization leveraging recurrent neural network language modeling techniques," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 23(8), pp. 1322–1334, August 2015.

[40] S. H. Liu, K. Y. Chen, Y. L. Hsieh, B. Chen, H. M. Wang, H. C. Yen, and W. L. Hsu, "A position-aware language modeling framework for extractive broadcast news speech summarization," *ACM Transactions on Asian and Low-Resource Language Information Processing*, 16(4), pp. 27:1–13, 2017.