Embedding Prior Task-specific Knowledge into Language Models for Context-aware Document Ranking

Shuting Wang
Gaoling School of Artificial Intelligence, Renmin University of China, Beijing, China
wangshuting@ruc.edu.cn

Yutao Zhu
Gaoling School of Artificial Intelligence, Renmin University of China, Beijing, China
yutaozhu94@gmail.com

Zhicheng Dou
Gaoling School of Artificial Intelligence, Renmin University of China, Beijing, China
dou@ruc.edu.cn

Abstract

Exploiting users' contextual behaviors in the current session has been proven favorable to the document ranking task. Recently, the context-aware document ranking task has benefited from pre-trained language models (PLMs) due to their superior ability in language modeling. Most PLM-based context-aware document ranking models implicitly learn task-specific knowledge by fine-tuning PLMs on historical search logs. However, since search log data is noisy and contains various user intents and search patterns, such a black-box way may prevent models from fully mastering effective context-aware search knowledge. To solve this problem, we propose LOCK, a PLM-based context-aware document ranking model that explicitly embeds task-specific prior knowledge into PLMs to guide the model optimization. From local to global, we identify three types of task-specific knowledge, including intra-turn signals, inter-turn signals, and global session signals. LOCK formulates such prior knowledge into prior attention biases for impacting the fine-tuning of PLMs. This operation can guide the ranking model by task-specific prior knowledge, thereby improving model convergence and ranking ability. Additionally, we introduce a task-specific pre-training stage that involves masked language modeling and the soft reconstruction of the prior attention matrix, which helps the PLMs adapt to our task. Extensive experiments validate the effectiveness and convergence of our method.

CCS Concepts

• **Information systems → Retrieval models and ranking**.

Keywords

Context-aware ranking, Prior knowledge, Language models

ACM Reference Format:
Shuting Wang, Yutao Zhu, and Zhicheng Dou. 2025. Embedding Prior Task-specific Knowledge into Language Models for Context-aware Document Ranking. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.1 (KDD '25), August 3–7, 2025, Toronto, ON, Canada.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3690624.3709282

KDD '25, August 3–7, 2025, Toronto, ON, Canada
© 2025 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 979-8-4007-1245-6/25/08
https://doi.org/10.1145/3690624.3709282

Table 1: Visualized attention distributions of some key terms over a session (including the previous query q_l, its clicked document d_l, the current query q, and its clicked document d). Darker colors mean higher attention values.

	(a) Attention distribution of the term "*madden*"
q_l	best offensive plays for madden of
d_l	madden tips madden strategies madden football plays
q	strategies offensive plays for madden of
d	madden nfl of guides and strategy
	(b) Attention distribution of the term "*strategies*"
q_l	best offensive plays for madden of
d_l	madden tips madden strategies madden football plays
q	strategies offensive plays for madden of
d	madden nfl of guides and strategy

1 Introduction

Search intents are usually complex and may require multiple queries to fully explore the desired information. Such a series of search queries and the associated user behaviors (*e.g.*, clicked documents) are referred to as a *search session* [2]. The contextual information in a search session, *e.g.*, historical user behaviors, is valuable to modeling the user's current intent and returning satisfactory ranking results, which has been confirmed by many studies [2, 37, 42].

To exploit the contextual information in measuring document relevance, various heuristic rule-based methods [29] and neural network-based models [1, 8, 31] have been proposed. Recently, pre-trained language models (PLMs), which have superior abilities in language modeling, have further boosted the development of context-aware document ranking [6, 25, 42].

Existing PLM-based context-aware document ranking methods typically concatenate the word sequences of session data, including historical queries, their clicked documents, the current query, and a candidate document, as the input. Then, they are fine-tuned on a large amount of search log data, hoping that matching patterns can be automatically captured. It can be achieved when the data are extremely clean and the matching signals are clear. However, this is not the case for the context-aware document ranking task, where search sessions contain complex search intents and diverse search patterns. Solely relying on data makes it hard to obtain the knowledge necessary to complete the task. Thus, it is beneficial to provide the model with additional task-related guidance (knowledge).

Let us use an example to illustrate the problem and analyze which kinds of task-related knowledge are missed by the PLM-based methods. Table 1 visualizes some key terms' attention distributions over a session.[1] We can see: (1) The model **captures term-matching signals insufficiently.** Term matching between the query and document is a very effective signal in various information retrieval tasks [4, 28]. However, from the example, we notice that the fine-tuned model mainly focuses on neighbor terms rather than matched terms, *e.g.*, the term "madden" in the query q pays more attention to its neighbor term "of" rather than the matched terms of the clicked document d, as shown in Table 1 (a). (2) The model has an **inadequate modeling for the user intents' change.** Capturing search intents from the user behavior sequence is critical for document ranking, while the query evolution process contains important signals for search intent transition [44]. So, an ideal attention distribution should focus on the terms that can reflect the query evolution. However, we can see that most attention interactions occur inside the queries or documents. As Table 1 (b) shows, the current query's term "strategies", an added term compared to the previous query q_l, mainly focuses on the terms inside the query rather than the same term of the previously clicked document, d_l. Indeed, d_l may be the source of this added term, which deserves more attention when modeling user intent changes. All these observations suggest that the **solely data-driven fine-tuning of PLMs may block their adequate grasp of task-specific knowledge, hence limiting their downstream abilities.**

To tackle this problem, we propose to **explicitly embed task-specific knowledge into PLMs when fine-tuning it for context-aware document ranking.** We design a P**L**M-based c**O**ntextual do**C**ument ranking model with explicitly embedded task-specific prior **K**nowledge, which is called **LOCK**. The basic idea is to formulate effective prior knowledge of context-aware document ranking into attention biases and affect the self-attention process. Such attention biases can guide the model's learning direction to shrink the search space of model parameters and improve the convergence and performance of the model. Specifically, we consider three types of task-specific prior knowledge from local to global perspectives: (1) **Intra-turn signals.** In each search turn, containing a query and a relevant document, term matching is a strong signal for relevance modeling proven by many traditional and PLM-based ranking methods [4, 38]. We believe that the attention interactions between the identical terms within a q-d pair should also be enhanced. (2) **Inter-turn signals.** During a search session, user intents often evolve with search turns. It can be reflected by query reformulation. Therefore, we propose to enhance the attention interactions between tokens at different session positions based on three common types of query reformulation. It mimics the changing of user intents, which can provide a better context for token representation modeling. (3) **Global session signals.** The current query is the most important part of a search session, as it directly shows the user's current search intent. Thus, we believe the global representation of the session should focus more on the current query's terms to capture more valuable relevance signals. Based on the above analyses, we transform such prior knowledge into attention biases by enhancing important attention interactions between input tokens. The attention biases are then added to the self-attention matrices of PLMs to achieve knowledge-guided model optimization.

Additionally, we notice that directly fine-tuning the PLM may lead to its inadaption to prior attention biases. Therefore, we introduce a task-specific pre-training stage before fine-tuning the PLM on our ranking task. We retain the masked language modeling task with a task-specific masking strategy to adapt PLMs' ability at semantic modeling to session data. Furthermore, to guarantee that the PLM can digest the embedded prior knowledge, we build a soft reconstruction task, where a naive decoder is used to reconstruct the prior attention matrix from model outputs. Experiments on three search log datasets validate that the ranking ability of PLM can be greatly enhanced by our task-specific prior knowledge.

The main contributions of our study are three-fold:

(1) We analyze the context-aware document ranking task and conclude three kinds of task-specific prior knowledge that have not been well-studied by existing PLM-based methods.

(2) We formulate such task-specific knowledge into attention bias to guide the fine-tuning process of PLMs, leading to improved convergence and effectiveness on our task.

(3) We introduce the task-specific pre-training stage with the soft reconstruction task of the prior attention matrix to enhance the adaptability of PLMs to context-aware document ranking.

2 Related Work

2.1 Context-aware Document Ranking

The benefit of utilizing the contextual session information to facilitate the document ranking has been demonstrated by many existing works [1, 6, 29, 42]. The model structures of context-aware document ranking have evolved from statistical methods [5, 29, 34, 37] to neural-network-based methods [1, 2, 8, 10, 11, 20, 31, 33].

Recently, due to the superior ability of PLMs, *e.g.*, BERT [9], to model semantic information, many advanced studies also leveraged them to solve the context-aware document ranking task and achieve significant improvement [6, 25, 33, 42, 43]. Qu et al. [25] proposed to equip BERT with self-devised additional structures to encode session information, and COCA [42] designed several pre-training tasks via data augmentation to enhance the BERT's ability to model user behaviors over sessions. ASE [6] leveraged PLMs with an encoder-decoder structure and devised three generative tasks to denoise contextual information via multi-task learning.

Nevertheless, these methods implicitly learn downstream knowledge from complex session data, leading to the loss of some heuristic but effective knowledge. In this paper, we propose to explicitly embed task-specific prior knowledge into BERT to enhance its effectiveness on the context-aware document ranking task.

2.2 Prior Attention for PLM

As a key component of Transformer [32], the self-attention module has been widely studied in existing work [3, 12, 14, 23, 36, 39, 40]. Traditional attention distributions are generated based on input sequences. Recently, some studies [14, 24, 24, 39–41] proposed to introduce attention distributions from other sources, namely prior attention [18], to supplement the input-generated attention distribution and promote the model performance. For example, Yang et al. [39] enhanced the locality of attention distributions by using

[1] Attention distributions are derived from the mean of the multi-head attention maps of a random Transformer layer (other layers have similar results) of a BERT model trained on AOL search logs, a popular context-aware document ranking dataset. We omit non-semantic tokens, *e.g.*"[SEP]" to show semantic modeling more intuitively.

a Gaussian distribution over positions as the attention bias. Xia et al. [36] introduced a word similarity matrix as prior knowledge to guide the BERT to solve the semantic textual similarity task.

Inspired by these studies, in this paper, we propose to embed task-specific prior knowledge of the context-aware document ranking task in the form of attention bias to guide the optimization of BERT and improve its performance and reliability.

3 Method

The goal of context-aware document ranking is to evaluate document relevance based on session information. Existing PLM-based methods learn to perform context-aware document ranking by being solely fine-tuned on the search log data. However, we discover that the automatically learned attention distributions are poor at capturing some heuristic but effective knowledge for context-aware document ranking. Therefore, we propose to explicitly embed task-specific prior knowledge in the form of attention bias to guide the optimization and improve the effectiveness and convergence of our ranking model. Following existing studies [42], we take BERT to build and demonstrate our methods. Further experiments and analyses that expand our approach to other PLMs are presented in Appendix B.

3.1 Problem Definition

Following existing works [6, 25, 42, 44], we first provide key notations and formulate the context-aware document ranking task as below. User behaviors in each session are represented by a sequence containing M issued queries, $\mathcal{S} = \{q_1, \cdots, q_M\}$, where queries are ordered by their issued timestamps. Each query q_i associates with n candidate documents $\mathcal{D}_i = \{d_{i,1}, \cdots, d_{i,n}\}$. The original text string entered into the search engine serves as the representation of each query, q_i, and the text content represents each candidate document $d_{i,j}$. We represent the context of a certain query q_i by all its previous queries in the same session and their clicked documents, *i.e.*, $C_i = \{q_1, d_1, \cdots, q_{i-1}, d_{i-1}\}$.[2] Consequently, the context-aware document ranking task is defined as: given an issued query q_i and its search context C_i, the model should compute relevance scores for its candidate documents to rank the relevant (clicked) documents as high as possible.

3.2 Overview

The framework of LOCK is shown in Figure 1. The workflow is: First, given the input sequence generated from a search session, we build a prior adjacent matrix of input tokens via the process of prior knowledge embedding. Then, we adopt a projector to map the adjacent matrix into prior attention biases and add them to the BERT's self-attention matrices. Finally, an MLP layer produces the document ranking score from the sequence output. The training process has two stages. The first one is the task-specific pre-training stage, where we leverage the objectives of task-specific MLM and soft reconstruction of the prior adjacent matrix to adapt BERT to our data distribution. The second one is the fine-tuning stage, where a ranking loss is used to optimize the BERT for our ranking task.

3.3 Prior Knowledge Embedding

In this section, we introduce how to embed task-specific prior knowledge into attention bias and incorporate it into BERT. Specifically, given a query q_i, its search context C_i, and a candidate document $d_{i,j}$ (d_i for brevity), we first tokenize them by a tokenizer, $\mathrm{Tok}(\cdot)$, *e.g.*, for the query q_i, we tokenize it as:

$$T^{q_i} = \mathrm{Tok}(q_i) = [t_1^{q_i}, t_2^{q_i}, \ldots, t_{n_{q_i}}^{q_i}], \tag{1}$$

where n_{q_i} is the length of the query q_i. Then, following existing studies [42], we concatenate these token sequences to construct the input sequence of BERT as follows:

$$I = [\mathtt{CLS}]T^{q_1}[\mathtt{EOS}]T^{d_1}[\mathtt{EOS}]\cdots T^{q_i}[\mathtt{EOS}][\mathtt{SEP}]T^{d_i}[\mathtt{SEP}], \tag{2}$$

where the [EOS] token indicates the end of a query or document and the [CLS] is the global token to represent the entire session. The length of the input sequence is denoted as L_I.

By viewing the input tokens as graph nodes, we can build the prior adjacent matrix via the prior knowledge embedding process, which is introduced from Section 3.3.1 to Section 3.3.3. Then, we apply the projector to map the adjacent matrix into prior attention biases and add them to the self-attention matrices of the BERT model. We depict the mapping process in Section 3.3.4.

3.3.1 Modeling Intra-turn Signals. In each search turn, including a query and a clicked document, exact term matching is the most basic and effective signal to model query-document relevance. It has been widely used in either traditional [28, 38] or PLM-based [4] IR models. While MarkedBERT [4] identifies matching tokens by the special token, in this paper, we believe that adding attention bias is a more direct and effective way to capture such signals. An ideal attention matrix should assign higher weights to the position of exactly matched tokens between query and document.

Therefore, for each search turn $(q_j, d_j), j \in [1, i]$ in the input session, we introduce bidirectional edges between the identical tokens. Formally, $\forall\ t_m \in T^{q_j}, t_n \in T^{d_j}$, we have:

$$\mathbf{A}[\phi(t_m), \phi(t_n)] = \mathbf{A}[\phi(t_n), \phi(t_m)] = w_1, \quad \text{if } t_m = t_n. \tag{3}$$

$\phi(x)$ is the index function producing the position of the token x in the input sequence I. $\mathbf{A} \in \mathbb{R}^{L_I \times L_I}$ is the adjacent matrix, and $w_1 \in \mathbb{R}^+$ is a hyper-parameter to control edges' relative importance.

3.3.2 Modeling Inter-turn Signals. Modeling the evolution of user intent with search turns is essential to capture context-aware user intents. Therefore, encoding token interactions across search turns is crucial to producing reasonable attention matrices for PLM-based context-aware document ranking. Existing studies [16, 17, 26, 27, 30] have identified several query reformulation patterns from different perspectives to represent user intent evolution. Following [7], our study mainly focuses on three representative ones, *i.e.*, specification, generalization, and topic change.[3] According to our analysis of these query reformulation patterns, we build weighted token edges across different search turns to facilitate the modeling of user intent changes. Furthermore, we adopt a sliding window with size W to establish these edges for modeling robust reformulation patterns. We take $(q_{j-k}, d_{j-k}, q_j, d_j), j \in [k+1, i], k \in [1, W]$ as an example to describe out edge building method.

[2] The first session query q_1 has no search context.

[3] The query reformulation patterns are not the focus of this paper. We leave the use of other more detailed reformulation patterns for our future work.

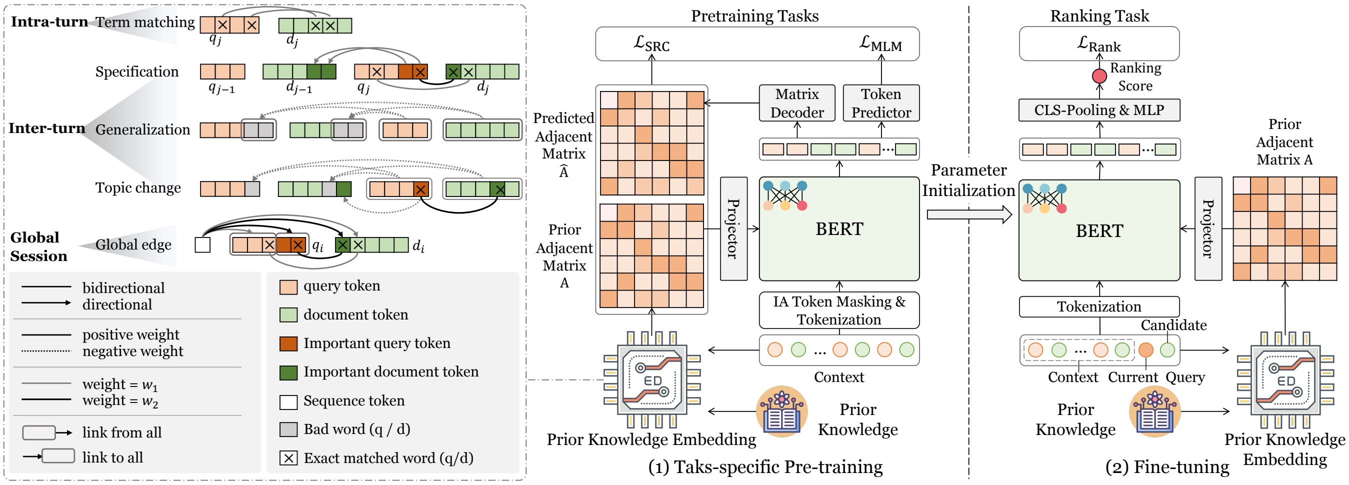

Figure 1: The architecture of LOCK. The left part shows an example to visualize the construction of the prior adjacent matrix for the input sequence. The right part demonstrates the framework of our method. "IA" is short for "importance-aware".

(1) **Specification.** Typically, it is challenging for users to clearly define the search intent at the beginning of a session when it is complex or ambiguous. They may therefore issue a brief and general query. As the search context develops, they will revise the query by adding terms to describe their search intent more precisely. Thus, given two query (q_1, q_2), if the token set of q_i is the subset of q_2, we view this query reformulation as *specification*. The added terms reflect the fine-grained user intent that is hard to be clarified by the original query. The previous study [30] also concluded that the clicked documents of previous queries are crucial sources of the added terms as they may inspire the next search intent. Thus, we consider that the added terms in the query should pay more attention to their source terms in the previously clicked documents. This approach can offer rich context information about the added terms and help the model understand user intent changes.

Therefore, for a query pair (q_{j-k}, q_j), where q_j is a specified query of q_{j-k}, we first identify the added tokens in q_j and their corresponding source tokens in d_{j-k}. These tokens in q_j form a list, which is denoted as S^{q_j}; for the source tokens in d_{j-k}, we have a list $S^{d_{j-k}}$. Then, we establish directional edges from the added tokens to their source tokens to enhance their attention on the source tokens, *i.e.*, $\forall\ t_m \in S^{q_j}, t_n \in S^{d_{j-k}}$, we have:

$$\mathbf{A}[\phi(t_m), \phi(t_n)] = w_1, \quad \text{if } t_m = t_n. \tag{4}$$

Further, because added tokens present more specific user intents, the term-matching of these important tokens should be promoted to emphasize their effects on relevance modeling. Thus, for the added tokens in the query q_j, we improve the weights of their term-matching edges (if have) *i.e.*, $\forall\ t_m \in S^{q_j}, t_n \in T^{d_j}$,

$$\mathbf{A}[\phi(t_m), \phi(t_n)] = \mathbf{A}[\phi(t_n), \phi(t_m)] = w_2 (> w_1), \quad \text{if } t_m = t_n. \tag{5}$$

(2) **Generalization.** In contrast to specification, generalization represents the situation where the user finds some tokens of previous queries that have negative impacts on searching for desired information. Thus, they remove undesired tokens to transform the query into a more general one. Therefore, given two queries (q_1, q_2), if the token set of q_2 is the subset of q_1, we view this reformulation as *generalization*. Similarly, we denote the removed token list of q_{j-k} as $G^{q_{j-k}}$. $G^{d_{j-k}}$ is used to denote the tokens in the document d_{j-k} that are exactly matched with those removed tokens in the query q_{j-k}. We believe that paying attention to these undesired tokens may impair the relevance modeling of the later query. Therefore, we link directional edges with negative weights from all tokens of the later query and document to previous undesired tokens to restrain these negative interactions, *i.e.*,

$$\mathbf{A}[\phi(t_m), \phi(t_n)] = -w_1, \quad \forall\ t_m \in T^{q_j}, t_n \in G^{q_{j-k}}; \tag{6}$$

$$\mathbf{A}[\phi(t_m), \phi(t_n)] = -w_1, \quad \forall\ t_m \in T^{d_j}, t_n \in G^{q_{j-k}}; \tag{7}$$

$$\mathbf{A}[\phi(t_m), \phi(t_n)] = -w_1, \quad \forall\ t_m \in T^{q_j}, t_n \in G^{d_{j-k}}; \tag{8}$$

$$\mathbf{A}[\phi(t_m), \phi(t_n)] = -w_1, \quad \forall\ t_m \in T^{d_j}, t_n \in G^{d_{j-k}}. \tag{9}$$

(3) **Topic change.** The last pattern of query reformulation is adding and removing terms to adjust the search topic, which implies a change in user interests. Thus, we call it *topic change*. It can be viewed as a combination of the previous two kinds of query reformulation. Consequently, for the query change process belonging to topic change, we conduct the operations of specification and generalization to add prior edges. It is worth noting that to avoid the interference of stopwords with query reformulation classification, we remove stopwords when classifying query reformulation patterns and identifying added/removed tokens.

3.3.3 *Modeling Global Session Signals.*

To serve the downstream ranking task, PLM-based methods [25, 33, 42] usually introduce a global token ([CLS] token in Eq. 3.3) and view its output as the relevance representation of the entire session. Meanwhile, the current query is the most important part of the user behavior sequence, which directly describes the current search intent. Consequently, we believe that the global token should assign more attention to the current query's tokens, hence capturing accurate relevance features and enhancing the document ranking. Concretely, recall that the token list of the current query q_i is denoted as T^{q_i} (defined in

Equation (1)), in which the added tokens are denoted as S^{q_i} (defined in Section 3.3.2). Here, we also consider the tokens in the candidate d_i exactly match with the current query, which is denoted as Q^{d_i}. It contains a sub-list, S^{d_i}, consisting of the document tokens that exactly match the query's added terms, S^{q_i}. Finally, we establish edges from the global token to the tokens in both T^{q_i} and Q^{d_i} to emphasize the current query's information. This can be defined as:

$$\mathbf{A}[0, \phi(t_m)] = \begin{cases} w_2, \forall\ t_m \in S^{q_i}; \\ w_1, \forall\ t_m \in T^{q_i}, t_m \notin S^{q_i}; \end{cases} \tag{10}$$

$$\mathbf{A}[0, \phi(t_n)] = \begin{cases} w_2, \forall\ t_n \in S^{d_i}; \\ w_1, \forall\ t_n \in Q^{d_i}, t_n \notin S^{d_i}. \end{cases} \tag{11}$$

The intuition here is: for the exact-matched tokens of the current query q_i, if they are added tokens, they should be assigned higher attention values; and if they are not, lower attention values are assigned. We do the same for the candidate document d_i as well.

3.3.4 From Adjacent Matrix to Prior Attention Bias. The BERT model mainly consists of multiple Transformer layers with multi-head self-attention modules. Therefore, for the l-th layer and h-th head, we denote the self-attention matrix before softmax as $\mathbf{M}^{l,h} \in \mathbb{R}^{L_I \times L_I}$. We project the prior adjacent matrix $\mathbf{A}$ into the attention bias and add it to the $\mathbf{M}^{l,h}$ for generating the final attention distribution $\hat{\mathbf{M}}^{l,h}$ as follows:

$$\hat{\mathbf{M}}^{l,h} = \mathrm{softmax}(\mathbf{M}^{l,h} + \alpha^{l,h}\mathbf{A}), \tag{12}$$

where $\alpha^{l,h}$ is a trainable scalar, controlling the impact of the prior attention bias on different Transformer layers and heads.

3.4 Task-specific Pre-training

In previous sections, we illustrated our proposed way to embed task-specific prior knowledge into BERT via attention bias. However, the BERT model is pre-trained on large-scale text corpora without incorporating prior attention bias, directly using prior attention bias when fine-tuning it on session data may lead to its inadaption for the downstream task and limit the ranking performance. To solve this problem, we introduce a task-specific pre-training stage that optimizes the prior knowledge-enhanced PLMs on the session data with two adaption objectives.

Firstly, we retain a widely used and effective pre-training task, Masked Language Modeling (MLM), for transferring the BERT's ability at language modeling to the distribution of session data. Furthermore, our input tokens have different importance, *e.g.*, added tokens are more important than removed tokens. Thus, it is better to let the MLM task focus more on predicting the important words to promote the model's capability of capturing semantic and prior knowledge. So, we propose an **importance-aware token masking**. Specifically, based on the prior adjacent matrix A, we leverage the in-degree of tokens, $\mathrm{ID}(t)$, to measure their importance and produce the masked probability $\beta(t)$ as:

$$\beta(t_i) = \mathrm{softmax}_{t \in I}\left(\mathrm{ID}\left(t_i\right)\right). \tag{13}$$

Then, the loss function is represented as:

$$\mathcal{L}_{\mathrm{MLM}} = -\sum\nolimits_{t_i \in \mathcal{M}} \log p(t_i|\tilde{I}), \tag{14}$$

where $\mathcal{M}$ denotes the masked token set that is sampled based on $\beta(\cdot)$ and $\tilde{I}$ is the input sequence after masking strategy.

Secondly, in our preliminary experiment, we found that, without the guidance of objective functions, it is difficult to digest the prior knowledge embedded in attention biases for the model. Therefore, we devise a simple decoder $\mathrm{Dec}(\cdot)$ to reconstruct the adjacent matrix $\hat{\mathbf{A}}$ from the BERT's output representations $\mathbf{h} \in \mathbb{R}^{L_I \times d}$ as:

$$\hat{\mathbf{A}} = \mathrm{Dec}(\mathbf{h}) = \mathbf{hWh}^{\top}. \tag{15}$$

d is hidden states' dimension, $\mathbf{W}$ is a parameter, and $\hat{\mathbf{A}}_{i,j}$ is the predicted weight of the edge from the i-th token to the j-th one.

Note that the weights of the edges are used to indicate the relative importance of different edge types, so we only focus on the prediction of relative values rather than absolute ones. We call this kind of reconstruction task **soft reconstruction task** (SRC). In this situation, the MSE loss function is unsuitable to measure the discrepancy between $\mathbf{A}$ and $\hat{\mathbf{A}}$. Consequently, we adopt a hierarchical margin loss function to measure how well the model learns the embedded prior knowledge. We expect that the mean value of the predicted edge weights of linked token pairs is higher than that of unlinked token pairs. Therefore, the loss function is defined as:

$$\mathcal{L}_{\mathrm{SRC}} = \sum\nolimits_{i=1}^{2} \max\left(0, \beta - \left(\mu(\hat{\mathbf{A}}_{w_i}) - \mu(\hat{\mathbf{A}}_{w_{i-1}})\right)\right), \tag{16}$$

where $\hat{\mathbf{A}}_{w_i}$ denotes the predicted value list of all token pairs with the edge weight equal to w_i, $\mu(\cdot)$ returns the mean value of the input, and β is a margin hyperparameter. Overall, the loss function of the task-specific pre-training is:

$$\mathcal{L}_{\mathrm{tpt}} = \lambda_1 \mathcal{L}_{\mathrm{MLM}} + \lambda_2 \mathcal{L}_{\mathrm{SRC}}, \tag{17}$$

where λ_1 and λ_2 are the hyperparameters that balance the weight of different loss functions.

3.5 Fine-tuning

Finally, we fine-tune the prior knowledge-enhanced BERT model on session data by a hinge-loss-based pair-wise loss function, which follows previous studies [6]. The loss is calculated as below:

$$\mathcal{L}_{\mathrm{rank}} = \sum_{\{d^+, d^-\} \in \mathcal{D}_i} \max\left(0, \gamma - \left(\mathrm{score}(d^+) - \mathrm{score}(d^-)\right)\right), \tag{18}$$

where $\mathcal{D}_i$ is the candidate set of query q_i, d^+, d^- denote the positive/negative documents that are labeled based on the click signals, and γ is a hyperparameter of margin. $\mathrm{score}(d)$ denotes the ranking score of a candidate d, which is yielded as follows:

$$\mathrm{score}(d_i) = f(\mathbf{h}_{[\mathrm{CLS}]}), \tag{19}$$

where $\mathbf{h}_{[\mathrm{CLS}]}$ is the output representation of global token from our model and $f : \mathbb{R}^d \to \mathbb{R}$ is an MLP layer.

4 Experiment

4.1 Datasets and Evaluation Metrics

4.1.1 Datasets. We experiment on three public datasets with search sessions, AOL [22], Tiangong-ST [7], and TREC 2014 Session Track.[4] Though the MSMARCO conversational dataset is also a choice,[5] its sessions are established manually instead of extracted from real

[4] https://trec.nist.gov/data/session2014.html

[5] https://github.com/microsoft/MSMARCO-Conversational-Search

search logs. However, our work aims to learn real user intents better from user behaviors, so we do not use this dataset.

The AOL dataset is provided by [2], which groups search logs into sessions. Each query in the training set and validation set has five candidate documents and fifty candidates retrieved by BM25 are provided for each query in the test set. The details of candidate construction can be found at [1]. The TREC 2014 Session Track (TREC for brevity) released 1,021 query sessions for 60 different topics, where each session has a relevant topic and each query contains the top ten results. The dataset also provides relevance judgments against the session topic at a 6-grade scale: spam (-2), not relevant (0), relevant (1), highly relevant (2), key (3), and navigational (4). We filter out some invalid sessions that lack relevance judgments. Due to the limited amount of data, we only use the TREC sessions as the test set to evaluate the generalization of experimental models trained on the AOL training dataset. The Tiangong-ST dataset collected 18-day search logs from a Chinese search engine, where ten candidates were provided for each query. In the test set, the dataset provides an annotated relevance score (0-4) for the last query of each session. Thus, this dataset contains two test sets, (1) Tiangong-ST-Click. It contains all queries in the test sessions except the last one. We regard click signals as relevance labels for experiments. (2) Tiangong-ST-Human. It covers the last queries of all test sessions with human-annotated relevance labels. To decrease memory load and improve the model's efficiency, we view the documents' titles as their content, following existing studies [2, 6, 33, 42].

The statistical information of datasets is presented in Table 4.

4.1.2 Evaluation Metrics.

Three common metrics are used to evaluate models' performance, *i.e.*, Mean Average Precision (MAP), Mean Reciprocal Rank (MRR), and Normalized Discounted Cumulative Gain at position k (NDCG@k, $k \in \{1, 3, 5, 10\}$). For the TREC and Tiangong-ST-Human with 5-scale human-annotated relevance labels, MAP and MRR are inappropriate for evaluation since they cannot consider relevance scales. Thus, following the suggestion from [7], we focus on NDCG@k of these datasets. We use TREC's official evaluation tool (trec_eval) [13] to evaluate all models.

4.2 Baselines

To evaluate the effectiveness of our model, we compare it with the following three types of baselines:

(1) **Ad-hoc ranking.** These models focus on evaluating matching scores between queries and candidate documents without information from search contexts. **ACR-I** [15] uses convolutional neural networks (CNNs) to embed queries and documents, then produces ranking scores by vector similarity function. **ACR-II** [15] uses CNNs to capture the fine-grained interactions from the matching map between query and document terms. **Duet** [21] combines interaction-based and representation-based features to learn more reliable ranking scores. **BERT** [9]. We fine-tune BERT by concatenating a query and a candidate document as the input to predict the relevance score by CLS-Pooling.

(2) **Context-aware document ranking with multi-task learning.** These methods leverage a multi-task framework to promote context-aware document ranking. **M-NSRF** [1] uses recurrent neural networks to jointly optimize next query prediction and context-aware document ranking tasks. **M-Match-Tensor** [1] (M-Match for brevity) is an improved version of M-NSF that models the contextual embeddings for terms of query and document. **CARS** [2] proposes to introduce implicit feedback from contextual information and optimizes the model by a multi-task framework consisting of query suggestion and document ranking tasks. **ASE** [6] designs three generative tasks and employs PLMs with an encoder-decoder structure to model the session data.

(3) **BERT-based context-aware document ranking.** These models fine-tune the BERT model and use its semantic modeling capabilities to complete ranking tasks. **HBA-Transformer** [25] (HBA for brevity) utilizes BERT with self-designed high-level Transformer structures to conduct context-aware document ranking. **COCA** [42] adopts contrastive learning with three data augmentation strategies to pre-train BERT and improves its robustness for encoding the session information. It is a classic case of applying BERT to context-aware document ranking.

We provide the implementation details in Appendix A due to limited space.

4.3 Overall Performances

(1) **Compared with all baselines, our model, LOCK, performs the best on most metrics.** It proves that incorporating task-specific prior knowledge as attention biases can improve the BERT's performance on the context-aware document ranking. Our analysis is that the embedded prior knowledge can alleviate the hardness of capturing task-specific knowledge, which can smooth the optimization surface and prevent the model from falling into a local optimum. Meanwhile, the pre-training can adapt the enhanced BERT to the data distribution and model structure of our task. It further eases the model optimization at the fine-tuning stage.

(2) **Our model performs the best within BERT-based models on all datasets.** To validate the effectiveness of embedded task-specific prior knowledge, it is appropriate to compare our model with other BERT-based methods. From Table 2, it is noticeable that LOCK produces the best results among all BERT-based models (including the BERT-based Ad-hoc model), which verifies the positive impact of incorporating heuristic knowledge to guide the optimization of BERT on the context-aware document ranking task. Also, it supports our hypothesis that learning adequate task-specific knowledge cannot be accomplished by merely fine-tuning the BERT model on large-scale search logs.

(3) **Our model outperforms all multi-task-based models on the AOL, TREC and Tiangong-ST-Click datasets.** We discover that ASE can yield comparable results with LOCK on the Tiangong-ST-Human dataset, but it performs significantly worse than LOCK on the TREC dataset. This may be because ASE is fine-tuned in a multi-task manner with multiple generative tasks. The overuse of data-driven learning tasks will make the model capture subtle characteristics of the datasets rather than the general features beneficial to the context-aware document ranking task. As a result, ASE performs well on the test samples from the same source of training data, while providing worse results in the scenario of zero-shot testing. Even without multi-task learning, LOCK can still significantly outperform ASE on the other three datasets, which validates the usefulness and generalizability of embedded prior knowledge for enhancing model performance on the task.

Table 2: Overall results of all models. "‡" and "†" indicate the model outperforms all baselines significantly in paired t-test at $p < 0.01$ and 0.05 level (with Bonferroni correction). The best and second-best results are in bold and underlined, respectively. Note that MAP and MRR cannot consider relevance scales, and are not applicable to TREC and Tiangong-ST-Human datasets.

Dataset	Metric	Ad-hoc Ranking				Multi-task Learning				BERT-based		
		ARC-I	ARC-II	Duet	BERT	M-NSRF	M-Match	CARS	ASE	HBA	COCA	LOCK
AOL	MAP	0.3361	0.3834	0.4038	0.4217	5264	0.4459	0.4297	0.5650	0.5281	0.5500	**0.5733**‡
	MRR	0.3475	0.3951	0.4111	0.5353	0.4326	0.4572	0.4408	0.5752	0.5384	0.5601	**0.5834**‡
	NDCG@1	0.1988	0.2428	0.2492	0.3807	0.2737	0.3020	0.2816	0.4144	0.3773	0.4024	**0.4240**‡
	NDCG@3	0.3108	0.3564	0.3822	0.5223	0.4025	0.4301	0.4117	0.5682	0.5241	0.5478	**0.5769**‡
	NDCG@5	0.3489	0.4026	0.4246	0.5584	0.4458	0.4697	0.4542	0.6007	0.5624	0.5849	**0.6094**‡
	NDCG@10	0.3953	0.4486	0.4675	0.5914	0.4886	0.5103	0.4971	0.6283	0.5951	0.6160	**0.6364**‡
TREC-Session	NDCG@1	0.2868	0.2949	0.2954	0.3456	0.2580	0.2592	0.2709	0.4314	0.3957	0.4351	**0.4754**‡
	NDCG@3	0.3260	0.3571	0.3605	0.4686	0.3404	0.3413	0.3528	0.5238	0.4952	0.5297	**0.5538**‡
	NDCG@5	0.4015	0.4150	0.4388	0.5472	0.4180	0.4129	0.4361	0.5950	0.5735	0.6017	**0.6161**‡
	NDCG@10	0.6059	0.6137	0.6203	0.6803	0.6080	0.6087	0.6185	0.7132	0.6998	0.7190	**0.7319**‡
Tiangong-ST-Click	MAP	0.6597	0.6729	0.6745	0.7450	0.6836	0.6778	0.6909	0.7459	0.6957	0.7481	**0.7518**†
	MRR	0.6826	0.6954	0.7026	0.7673	0.7065	0.6993	0.7134	0.7684	0.7171	0.7696	**0.7741**†
	NDCG@1	0.5315	0.5458	0.5738	0.6367	0.5609	0.5499	0.5677	0.6349	0.5726	0.6386	**0.6477**†
	NDCG@3	0.6383	0.6553	0.6511	0.7373	0.6698	0.6636	0.6764	0.7419	0.6807	0.7445	**0.7478**
	NDCG@5	0.6946	0.7086	0.6955	0.7824	0.7188	0.7199	0.7271	0.7828	0.7292	0.7858	**0.7882**†
	NDCG@10	0.7509	0.7608	0.7621	0.8157	0.7691	0.7646	0.7746	0.8166	0.7781	0.8180	**0.8209**†
Tiangong-ST-Human	NDCG@1	0.7088	0.7131	0.7577	0.7636	0.7124	0.7311	0.7385	**0.7884**	0.7612	0.7769	0.7697
	NDCG@3	0.7087	0.7237	0.7354	0.7641	0.7308	0.7233	0.7386	**0.7727**	0.7518	0.7576	0.7721
	NDCG@5	0.7317	0.7379	0.7548	0.7753	0.7489	0.7427	0.7512	0.7839	0.7639	0.7703	**0.7849**
	NDCG@10	0.8691	0.8732	0.8829	0.8942	0.8795	0.8801	0.8837	**0.8996**	0.8896	0.8932	0.8978

Table 3: Results of ablation studies on the AOL dataset.

Models	MAP	MRR	NDCG@1	NDCG@5
LOCK (Full)	0.5733	0.5834	0.4240	0.6094
w/o Term Matching Edges	0.5665	0.5766	0.4167	0.6021
w/o Adding Term Edges	0.5656	0.5757	0.4154	0.6013
w/o Removing Term Edges	0.5660	0.5759	0.4160	0.6015
w/o Global Edges	0.5646	0.5747	0.4151	0.5990
w/o Task-specific Pre-training	0.5705	0.5805	0.4207	0.6057
w/o Soft Reconstruction	0.5722	0.5823	0.4229	0.6077

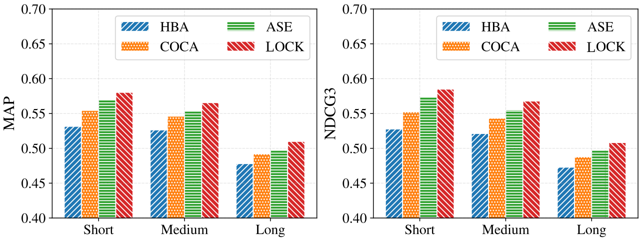

Figure 2: Experimental results of our model on different session lengths.

4.4 Further Analysis

4.4.1 Ablation Study. To further verify the effectiveness of our modules, we conduct several ablation studies on the AOL dataset as follows. The experimental results are illustrated in Table 3.

(1) **Effectiveness of different types of prior edges**. To fully capture the user intents in the context of search sessions, we introduce four types of prior knowledge and simulate them by corresponding prior edges, including term matching, adding term, removing term, and global edges. To validate the impact of such prior knowledge on our model, we abandon these edges and construct four variants of our model. The performances presented in the second to fifth rows of Table 3 all decline significantly. It implies that our prior edges can capture knowledge from the perspectives of term matching, intent revolution, and query importance, which are all favorable for our task. Optimizing BERT without explicitly embedded prior knowledge makes it hard to learn such useful knowledge, hence resulting in worse ranking results.

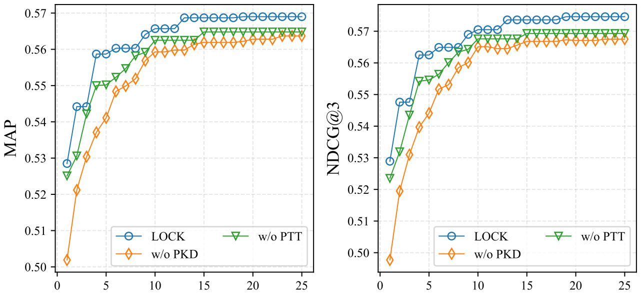

Figure 3: Performance growth curves of model variants.

(2) **Effectiveness of the task-specific pre-training.** To avoid the inadaption of the prior knowledge-enhanced BERT to our task, we conduct task-specific pre-training before fine-tuning it. To test the impact of this process, we train two variant models, one drops the soft reconstruction task, and the other one abandons the task-specific pre-training process. The decreased results presented in Table 3 imply the necessity of task-specific pre-training. We also notice that the performance degradation of these two variants is smaller than that of other variants without some prior knowledge. It reveals that LOCK's superiority mostly stems from task-specific prior knowledge rather than merely using task-specific pre-training.

4.4.2 *Effect of Session Lengths.*

The session length is a key factor that determines the richness of contextual information. To test the generality of our model on different session lengths, we split the test session into three groups, including short sessions (length≤ 2), medium sessions (length = 3 or 4), and long sessions (length > 4). We compare our model with several context-aware document ranking methods and present the experimental results on the left side of Figure 2. The results imply that our model outperforms all baselines across all session groups. It validates the robustness of our model in various search contexts. We attribute this advantage to the use of various task-specific knowledge. For example, the exact matching signals enable our model to capture more ad-hoc relevance signals, leading to high-quality ranking results for short sessions with less context information. For long sessions with abundant context, the prior edges encoding query reformulations enhance our model to capture the change of user intents more precisely.

4.4.3 *Model Convergence.*

To validate our hypothesis that embedded prior knowledge and the task-specific pre-training process can accelerate model convergence, we present the performance curves during fine-tuning LOCK and its two variants in Figure 3. *w/o* TPT drops the task-specific pre-training stage, and *w/o* PKD further drops the embedded prior knowledge. Concretely, we train all models for five epochs during the fine-tuning stage and validate the model's performance every 0.2 epoch, resulting in 25 steps. The results shown in Figure 3 suggest the convergence rate and performance upper bound of the models are LOCK > *w/o* TPT > *w/o* PKD. The potential reason is that, without the task-specific pre-training, the BERT is not adapted to the data distribution of the ranking task and the model structure with attention biases, making it hard to converge during fine-tuning. When we further drop the embedded prior knowledge, it is challenging to implicitly learn the task-specific knowledge from the downstream data, which may also hurt the model optimization. These results support our aforementioned hypothesis. Interestingly, the initial performance of *w/o* PKD is significantly worse than the other two models. We speculate that the published checkpoint of BERT is not a proper starting point for our task, since the learning tasks and data distributions of BERT's pre-training stage are unrelated to the downstream task. It also confirms the necessity of the task-specific pre-training stage.

4.4.4 *Impact of Amount of Training Data.*

The amount of training data determines the upper bound of the task-specific knowledge the model can learn for such loss-guided models, which substantially impacts the model performance. Therefore, we train LOCK on four different training sets with different data proportions to explore their effect. Results are compared with two strong baselines (COCA and ASE) and shown in Figure 4. It is obvious that even with 20% of the training data, LOCK can still achieve comparable performance with COCA. When LOCK is trained on 60% of the training data, it can perform similarly to ASE. The experimental results verify the effectiveness of LOCK in the case of less training data, which may inspire its application of few-shot learning. We think the reason may be that there are two paths to learning knowledge of the downstream task for our model, *i.e.*, capturing implicit knowledge from training data, and learning heuristic knowledge from prior attention bias. Thus, when the training data is limited with less implicit knowledge, LOCK can learn useful matching patterns from external knowledge. It also proves the importance of incorporating task-specific prior knowledge into BERT-based models.

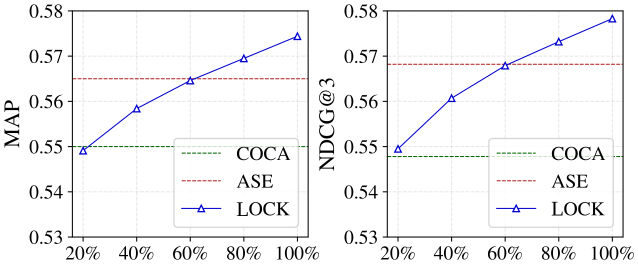

Figure 4: Results of different training data amounts.

Furthermore, We provide some generalization experiments and case studies in Appendix B and C.

5 Conclusion

Existing PLM-based context-aware document ranking methods usually fine-tune PLMs on search logs, hoping to automatically learn task-specific knowledge from session data. However, session data contains complicated user intents and diverse search patterns. It is hard to sufficiently capture task-specific knowledge solely relying on the training data for the context-aware document ranking task. Consequently, we proposed to explicitly embed task-specific prior knowledge to enhance the BERT's ability on the context-aware document ranking task. Specifically, based on three types of prior knowledge, *i.e.*, term-matching signal, user intent evolution modeling, and current query modeling, we formulated them into attention biases by introducing prior edges to guide the optimization of the BERT on our task. Further, we leveraged the task-specific pre-training with MLM and SRC tasks to adapt the BERT to our task. Experiments confirm the effectiveness and convergence of LOCK. In this paper, we focused on embedding prior knowledge into self-attention modules to enhance the BERT model. In the future, it is possible to design an embedding method that is suitable for more kinds of PLMs to improve the generality of our method.

Acknowledgments

Zhicheng Dou is the corresponding author. This work was supported by National Natural Science Foundation of China No. 62272467, the fund for building world-class universities (disciplines) of Renmin University of China. The work was partially done at the Engineering Research Center of Next-Generation Intelligent Search and Recommendation, MOE.

References

[1] Wasi Uddin Ahmad, Kai-Wei Chang, and Hongning Wang. 2018. Multi-Task Learning for Document Ranking and Query Suggestion. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. https://openreview.net/forum?id=SJ1nzBeA-

[2] Wasi Uddin Ahmad, Kai-Wei Chang, and Hongning Wang. 2019. Context Attentive Document Ranking and Query Suggestion. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2019, Paris, France, July 21-25, 2019*, Benjamin Piwowarski, Max Chevalier, Éric Gaussier, Yoelle Maarek, Jian-Yun Nie, and Falk Scholer (Eds.). https://doi.org/10.1145/3331184.3331246

[3] Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The Long-Document Transformer. *CoRR* abs/2004.05150 (2020).

[4] Lila Boualili, Jose G. Moreno, and Mohand Boughanem. 2020. MarkedBERT: Integrating Traditional IR Cues in Pre-Trained Language Models for Passage Retrieval *(SIGIR '20)*. Association for Computing Machinery, New York, NY, USA, 1977–1980. https://doi.org/10.1145/3397271.3401194

[5] Huanhuan Cao, Daxin Jiang, Jian Pei, Enhong Chen, and Hang Li. 2009. Towards Context-Aware Search by Learning a Very Large Variable Length Hidden Markov Model from Search Logs. In *Proceedings of the 18th International Conference on World Wide Web* (Madrid, Spain) *(WWW '09)*. New York, NY, USA. https://doi.org/10.1145/1526709.1526736

[6] Haonan Chen, Zhicheng Dou, Yutao Zhu, Zhao Cao, Xiaohua Cheng, and Ji-Rong Wen. 2022. Enhancing User Behavior Sequence Modeling by Generative Tasks for Session Search. In *CIKM*.

[7] Jia Chen, Jiaxin Mao, Yiqun Liu, Min Zhang, and Shaoping Ma. 2019. TianGong-ST: A New Dataset with Large-Scale Refined Real-World Web Search Sessions. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management* (Beijing, China) *(CIKM '19)*. New York, NY, USA. https://doi.org/10.1145/3357384.3358158

[8] Wanyu Chen, Fei Cai, Honghui Chen, and Maarten de Rijke. 2018. Attention-Based Hierarchical Neural Query Suggestion. In *The 41st International ACM SIGIR Conference on Research &amp; Development in Information Retrieval* (Ann Arbor, MI, USA) *(SIGIR '18)*. New York, NY, USA. https://doi.org/10.1145/3209978.3210079

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota. https://doi.org/10.18653/v1/N19-1423

[10] Dongyi Guan and Hui Yang. 2014. Query Aggregation in Session Search. In *Proceedings of the 3rd Workshop on Data-Driven User Behavioral Modeling and Mining from Social Media* (Shanghai, China) *(DUBMOD '14)*. New York, NY, USA. https://doi.org/10.1145/2665994.2666001

[11] Dongyi Guan, Sicong Zhang, and Hui Yang. 2013. Utilizing Query Change for Session Search. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Dublin, Ireland) *(SIGIR '13)*. New York, NY, USA. https://doi.org/10.1145/2484028.2484055

[12] Qipeng Guo, Xipeng Qiu, Xiangyang Xue, and Zheng Zhang. 2019. Low-Rank and Locality Constrained Self-Attention for Sequence Modeling. 27, 12 (nov 2019), 2213–2222. https://doi.org/10.1109/TASLP.2019.2944078

[13] Christophe Van Gysel and Maarten de Rijke. 2018. Pytrec_eval: An Extremely Fast Python Interface to trec_eval. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR 2018, Ann Arbor, MI, USA, July 08-12, 2018*, Kevyn Collins-Thompson, Qiaozhu Mei, Brian D. Davison, Yiqun Liu, and Emine Yilmaz (Eds.). https://doi.org/10.1145/3209978.3210065

[14] Ruining He, Anirudh Ravula, Bhargav Kanagal, and Joshua Ainslie. 2021. RealFormer: Transformer Likes Residual Attention. In *ACL/IJCNLP (Findings) (Findings of ACL, Vol. ACL/IJCNLP 2021)*. Association for Computational Linguistics, 929–943.

[15] Baotian Hu, Zhengdong Lu, Hang Li, and Qingcai Chen. 2014. Convolutional Neural Network Architectures for Matching Natural Language Sentences. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2* (Montreal, Canada) *(NIPS'14)*. Cambridge, MA, USA.

[16] Jeff Huang and Efthimis N. Efthimiadis. 2009. Analyzing and Evaluating Query Reformulation Strategies in Web Search Logs. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management* (Hong Kong, China) *(CIKM '09)*. Association for Computing Machinery, New York, NY, USA, 77–86. https://doi.org/10.1145/1645953.1645966

[17] Bernard J. Jansen, Danielle L. Booth, and Amanda Spink. 2009. Patterns of Query Reformulation during Web Searching. *J. Am. Soc. Inf. Sci. Technol.* 60, 7 (jul 2009), 1358–1371.

[18] Tianyang Lin, Yuxin Wang, Xiangyang Liu, and Xipeng Qiu. 2021. A Survey of Transformers. *CoRR* abs/2106.04554 (2021).

[19] Ilya Loshchilov and Frank Hutter. 2019. Decoupled Weight Decay Regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. https://openreview.net/forum?id=Bkg6RiCqY7

[20] Jiyun Luo, Sicong Zhang, and Hui Yang. 2014. Win-Win Search: Dual-Agent Stochastic Game in Session Search. In *Proceedings of the 37th International ACM SIGIR Conference on Research &amp; Development in Information Retrieval* (Gold Coast, Queensland, Australia) *(SIGIR '14)*. New York, NY, USA. https://doi.org/10.1145/2600428.2609629

[21] Bhaskar Mitra, Fernando Diaz, and Nick Craswell. 2017. Learning to Match Using Local and Distributed Representations of Text for Web Search. In *Proceedings of the 26th International Conference on World Wide Web* (Perth, Australia) *(WWW '17)*. Republic and Canton of Geneva, CHE. https://doi.org/10.1145/3038912.3052579

[22] Greg Pass, Abdur Chowdhury, and Cayley Torgeson. 2006. A Picture of Search. In *Proceedings of the 1st International Conference on Scalable Information Systems* (Hong Kong) *(InfoScale '06)*. New York, NY, USA. https://doi.org/10.1145/1146847.1146848

[23] Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron Courville. 2018. FiLM: Visual Reasoning with a General Conditioning Layer. *Proceedings of the AAAI Conference on Artificial Intelligence* 32, 1 (Apr. 2018). https://doi.org/10.1609/aaai.v32i1.11671

[24] Jonathan Pilault, Amine Elhattami, and Christopher J. Pal. 2021. Conditionally Adaptive Multi-Task Learning: Improving Transfer Learning in NLP Using Fewer Parameters & Less Data. In *ICLR*. OpenReview.net.

[25] Chen Qu, Chenyan Xiong, Yizhe Zhang, Corby Rosset, W. Bruce Croft, and Paul N. Bennett. 2020. Contextual Re-Ranking with Behavior Aware Transformers. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*, Jimmy Huang, Yi Chang, Xueqi Cheng, Jaap Kamps, Vanessa Murdock, Ji-Rong Wen, and Yiqun Liu (Eds.). https://doi.org/10.1145/3397271.3401276

[26] Eun Youp Rha, Matthew Mitsui, Nicholas J. Belkin, and Chirag Shah. 2016. Exploring the Relationships between Search Intentions and Query Reformulations. In *Proceedings of the 79th ASIS&amp;T Annual Meeting: Creating Knowledge, Enhancing Lives through Information &amp; Technology* (Copenhagen, Denmark) *(ASIST '16)*. American Society for Information Science, USA, Article 48, 9 pages.

[27] Soo Young Rieh and Hong (Iris) Xie. 2006. Analysis of Multiple Query Reformulations on the Web: The Interactive Information Retrieval Context. *Inf. Process. Manage.* 42, 3 (may 2006), 751–768. https://doi.org/10.1016/j.ipm.2005.05.005

[28] Stephen Robertson and Hugo Zaragoza. 2009. The Probabilistic Relevance Framework: BM25 and Beyond. *Found. Trends Inf. Retr.* 3, 4 (apr 2009). https://doi.org/10.1561/1500000019

[29] Xuehua Shen, Bin Tan, and ChengXiang Zhai. 2005. Context-Sensitive Information Retrieval Using Implicit Feedback. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (Salvador, Brazil) *(SIGIR '05)*. New York, NY, USA. https://doi.org/10.1145/1076034.1076045

[30] Marc Sloan, Hui Yang, and Jun Wang. 2015. A term-based methodology for query reformulation understanding. *Inf. Retr. J.* 18, 2 (2015). https://doi.org/10.1007/s10791-015-9251-5

[31] Alessandro Sordoni, Yoshua Bengio, Hossein Vahabi, Christina Lioma, Jakob Grue Simonsen, and Jian-Yun Nie. 2015. A Hierarchical Recurrent Encoder-Decoder for Generative Context-Aware Query Suggestion. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management* (Melbourne, Australia) *(CIKM '15)*. New York, NY, USA. https://doi.org/10.1145/2806416.2806493

[32] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is All You Need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems* (Long Beach, California, USA) *(NIPS'17)*. Red Hook, NY, USA.

[33] Shuting Wang, Zhicheng Dou, and Yutao Zhu. 2023. Heterogeneous Graph-based Context-aware Document Ranking. In *WSDM '23: The Sixteenth ACM International Conference on Web Search and Data Mining, Singapore, Singapore, February 27-March 3, 2023*. ACM. https://doi.org/10.1145/3539597.3570390

[34] Ryen W. White, Paul N. Bennett, and Susan T. Dumais. 2010. Predicting Short-Term Interests Using Activity-Based Search Context. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management* (Toronto, ON, Canada) *(CIKM '10)*. New York, NY, USA. https://doi.org/10.1145/1871437.1871565

[35] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-Art Natural Language Processing. In *EMNLP (Demos)*. Association for Computational Linguistics, 38–45.

[36] Tingyu Xia, Yue Wang, Yuan Tian, and Yi Chang. 2021. Using Prior Knowledge to Guide BERT's Attention in Semantic Textual Matching Tasks. In *Proceedings of the Web Conference 2021* (Ljubljana, Slovenia) *(WWW '21)*. Association for Computing Machinery, New York, NY, USA, 2466–2475. https://doi.org/10.1145/3442381.3449988

[37] Biao Xiang, Daxin Jiang, Jian Pei, Xiaohui Sun, Enhong Chen, and Hang Li. 2010. Context-Aware Ranking in Web Search. In *Proceedings of the 33rd International*

ACM SIGIR Conference on Research and Development in Information Retrieval (Geneva, Switzerland) *(SIGIR '10)*. New York, NY, USA. https://doi.org/10.1145/1835449.1835525

[38] Chenyan Xiong, Zhuyun Dai, Jamie Callan, Zhiyuan Liu, and Russell Power. 2017. End-to-End Neural Ad-Hoc Ranking with Kernel Pooling. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Shinjuku, Tokyo, Japan) *(SIGIR '17)*. Association for Computing Machinery, New York, NY, USA, 55–64. https://doi.org/10.1145/3077136.3080809

[39] Baosong Yang, Zhaopeng Tu, Derek F. Wong, Fandong Meng, Lidia S. Chao, and Tong Zhang. 2018. Modeling Localness for Self-Attention Networks. In *EMNLP*. Association for Computational Linguistics, 4449–4458.

[40] Chengxuan Ying, Guolin Ke, Di He, and Tie-Yan Liu. 2021. LazyFormer: Self Attention with Lazy Update. *CoRR* abs/2102.12702 (2021). arXiv:2102.12702 https://arxiv.org/abs/2102.12702

[41] Weiqiu You, Simeng Sun, and Mohit Iyyer. 2020. Hard-Coded Gaussian Attention for Neural Machine Translation. In *ACL*. Association for Computational Linguistics, 7689–7700.

[42] Yutao Zhu, Jian-Yun Nie, Zhicheng Dou, Zhengyi Ma, Xinyu Zhang, Pan Du, Xiaochen Zuo, and Hao Jiang. 2021. Contrastive Learning of User Behavior Sequence for Context-Aware Document Ranking. In *CIKM '21: The 30th ACM International Conference on Information and Knowledge Management, Virtual Event, QLD, Australia, November 1-5, 2021*.

[43] Yutao Zhu, Jian-Yun Nie, Yixuan Su, Haonan Chen, Xinyu Zhang, and Zhicheng Dou. 2022. From Easy to Hard: A Dual Curriculum Learning Framework for Context-Aware Document Ranking. In *CIKM*.

[44] Xiaochen Zuo, Zhicheng Dou, and Ji-Rong Wen. 2022. Improving Session Search by Modeling Multi-Granularity Historical Query Change. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining* (Virtual Event, AZ, USA) *(WSDM '22)*. Association for Computing Machinery, New York, NY, USA, 1534–1542. https://doi.org/10.1145/3488560.3498415

A Implementation Details

We employ the pre-trained BERT provided by HuggingFace [35]. Following [25, 42], the maximum length of input sequences is set as 128 to trade off efficiency and effectiveness. For sequences longer than 128, we truncate them by popping the head tokens of query-document pairs. For the construction of the prior adjacent matrix, we set the window size k as 2 and the edge weights as $w_1 = 1, w_2 = 2$. For the task-specific pre-training stage, we set the masked probability as 0.3 for the MLM task and set the margin value, β, as 1 for the soft reconstruction task. The loss weights for two tasks are set to 1, *i.e.*, $\lambda_1 = \lambda_2 = 1$. We pre-train the BERT model for ten epochs with 128 batch size, and the learning rate is 5e-5 with a linear decay. For the ranking task, the margin value, γ, is set to 1. The pre-trained model is fine-tuned over five epochs with 128 batch size. The 5e-5 learning rate linearly decays during fine-tuning. AdamW [19] is adopted as the optimizer in both training stages. Note that the time and memory cost of the pre-training stage are comparable to that of the fine-tuning since their data sources are both session data. Our code is released in github https://github.com/ShootingWong/LOCK.

Table 4: Statistics of two datasets. "Query", "Document", "Session", and "Relevant documents" are abbreviated to "Qry", "Doc", "Sess", and "Rel". (Back to the main paper, 4.1.2)

Items	AOL			Tiangong-ST			TREC
	Train	Valid	Test	Train	Valid	Test	Test
# Sessions	219,748	34,090	29,369	143,155	2,000	2,000	962
# Queries	566,967	88,021	76,159	344,806	5,026	6,420	1970
# Qry / Sess	2.58	2.58	2.59	2.41	2.51	3.21	2.05
# Doc / Qry	5	5	50	10	10	10	10
Avg. Qry Len	2.86	2.85	2.9	2.89	1.83	3.46	3.57
Avg. Doc Len	7.27	7.29	7.08	8.25	6.99	9.18	10.98
# Rel / Qry	1.08	1.08	1.11	0.94	0.53	3.65	3.32

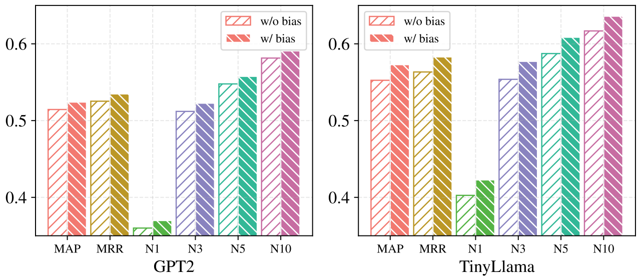

Figure 5: Performance of different LLMs.

B Generalization on Different Language Models

Large language models (LLMs) have demonstrated superior performance across various NLP tasks due to excellent language understanding and generation abilities. However, the complex information structure blocks the application of LLMs on the context-aware document ranking task. To confirm the generalization of our methods across different LLMs, we select GPT and Llama as two representative models and compare their performance with and without our prior task-specific knowledge. Due to the limited GPU resources, we practically fine-tune GPT2 and Tinyllama to conduct our experiment. The results are shown in Figure 5. Obviously, introducing prior knowledge significantly enhances the LLMs' performance in our task. This phenomenon further proves our assumption that directly fine-tuning language models in the downstream task makes it hard to sufficiently grasp the task-specific knowledge, hence limiting the model performance. Our proposed method explicitly injects this prior knowledge into language models, thereby enhancing the guidance and convergence of model optimization.

C Case study

To verify that our model is able to guide the BERT's attention distributions to be consistent with task-specific prior knowledge, we conduct a case study to compare some keywords' attention distributions produced by our model and COCA (a classic BERT-based context-aware document ranking model without embedding task-specific prior knowledge). The visualization results are presented in Table 5. Similar to Table 1, we indicate the keywords by black boxes and highlight session words with green squares based on their attention values, where the darker the color, the higher the attention.

We illustrate the two models' attention distributions for "design", an important added term in the current query q. It is evident that the word "design" also appears in the previously clicked document d_l. Therefore, the user may be inspired by the viewed document to add this word, and the added term should pay more attention to the same word in the d_l to simulate this evolution of user intent. The comparison implies that LOCK's attention distributions match our expectations, but the BERT-based model without prior knowledge enhancement neglects this signal. Also, for our model, the "design" in the query q also pays high attention to the candidate d's exactly matched term, which is also disregarded by COCA. These results reveal that our proposed model with embedded prior knowledge can better capture the evolution of user intent and critical relevance signals than BERT-based models, yielding better results for users.

Table 5: A case to visualize the attention distributions.

(a) Attention distribution of "*design*" from LOCK	
q_l	business logo
d_l	logo design usa based 100 money back guarantee
q	business logo design des moines iowa
d	logo design web design graphic design

(b) Attention distribution of "*design*" from COCA	
q_l	business logo
d_l	logo design usa based 100 money back guarantee
q	business logo design des moines iowa
d	logo design web design graphic design