# Incorporating Ranking Context for End-to-End BERT Re-ranking

Xiaoyang Chen[1,2], Kai Hui[3⋆], Ben He[1,2✉], Xianpei Han[2], Le Sun[2], and Zheng Ye[4✉]

[1] University of Chinese Academy of Sciences, Beijing, China
[2] Institute of Software, Chinese Academy of Sciences, Beijing, China
chenxiaoyang19@mails.ucas.ac.cn, benhe@ucas.ac.cn, {xianpei,sunle}@iscas.ac.cn
[3] Amazon Alexa, Berlin, Germany kai.hui.bj@gmail.com
[4] South-Central University for Nationalities, Wuhan, China yezheng@scuec.edu.cn

**Abstract.** Ranking context has been shown crucial for the performance of learning to rank. Its use for the BERT-based re-rankers, however, has not been fully explored. In this work, an end-to-end BERT-based ranking model has been proposed to incorporate the ranking context by modeling the interactions between a query and multiple documents in the same ranking jointly, using the pseudo relevance feedback to adjust the relevance weightings. Extensive experiments on standard TREC test collections confirm the effectiveness of the proposed model in improving the BERT-based re-ranker with low extra computation cost.

## 1 Introduction

Recent advances in information retrieval have shown promising performance gain by utilizing large-scale pre-trained transformer-based language models like BERT [12,27,40,57]. Most of these models, however, consider query-document pairs independently. Actually, unlike in ordinal classification, the main goal of a ranking problem is to optimize ranking lists given queries, making the consideration of the context of the ranking important, such as the *local ranking context* in terms of cross-document interactions [2,43,44]. There have been many successful attempts to incorporate the ranking context, mostly in learning-to-rank-based methods. In early works, loss functions have been proposed to optimize on top of a pair or a list of documents [5,28,31,55], modeling the cross-document interactions at loss level, achieving superior performance on L2R benchmark [46]. In addition, a groupwise ranking framework for multivariate scoring functions is proposed [2] to determine the relevance scores of a group of documents jointly, taking handcrafted learning-to-rank features as query-document presentations and using stack of dense layers to evaluate the relevance. More recently, a neural learning-to-rank model named SetRank [43] is proposed to directly learn a

⋆ Now at Google AI.

ranking model defined on document sets, employing a stack of multi-head self-attention blocks to learn the embedding for all documents jointly, successfully incorporating the local ranking context and leading to promising improvements.

To the best of our knowledge, however, such ranking context has not been successfully used to enhance the state-of-the-art neural ranking models based on pre-trained language models, like BERT. Indeed, as mentioned in [45], using pairwise loss when employing BERT for re-ranking does not lead to improvements. Beyond single query-document pairs, duoBERT [41] concatenates two documents and the query before feeding into BERT layers, and the output from BERT is trained to learn pairwise comparisons between two documents. However, there exists no straightforward extension to incorporate the full local ranking context using duoBERT as BERT model can not encode very long sequence. Inspired by the success of SetRank [43], in this work, we aim to develop a novel model that could incorporate the ranking context on top of the BERT-based contextualized ranking models, advancing the state-of-the-art BERT-ranker. In a nutshell, comparing with SetRank, BERT-based ranker requires the learning of the encoder during the incorporation of the ranking context, requiring novel framework to enable the end-to-end training. Besides, due to the complexity and huge size of the BERT model [13], special designs are desired to enable the joint modeling of hundreds or even thousands of documents.

To bridge this gap, we propose a groupwise BERT-based ranking model, *Co-BERT*, which is equipped to consider the ranking context. In the groupwise scorer, inspired by [43], candidate documents are grouped together and their interaction representations are passed through several BERT layers to model the ranking context, before projecting the outputs into ranking scores. This groupwise scorer and the BERT encoder for individual query-document pairs are trained end-to-end with pointwise loss. Therein, the groupwise scorer should be able to incorporate the ranking context for hundreds or even thousands of documents; however, individual batch can only include a limited number of documents due to the huge amount of parameters in BERT. To mitigate this dilemma, a ranking list is divided into groups of documents from the same ranking, and pseudo-relevance feedback (PRF) is exploited to capture the query-specific information, calibrating the relevance weightings among different groups.

Contributions in this paper are threefold. 1) We propose an end-to-end groupwise BERT-based ranking model, enabling the joint learning of the query-document interactions and the intra-documents ranking context over BERT. 2) A light-weight PRF-based calibration method is proposed to incorporate ranking context for long list of documents, further boosting the groupwise scorer with small extra computational cost. 3) Extensive evaluation demonstrates that Co-BERT can advance the effectiveness of the state-of-the-art BERT re-ranker. Besides, while providing improvements in ranking effectiveness, the extra computation cost of Co-BERT during inference is as least as 0.3% compared with a standard BERT re-ranker. Source code and data are publicly available at https://github.com/VerdureChen/Co-BERT.

## 2 Related Work

**BERT-based ranking and Pseudo Relevance Feedback (PRF).** Many existing works have attempted to apply BERT for ranking from different aspects, including training models with large amounts of data [40], scoring documents with sentence-level or passage-level information [12,20,27,54,57], multi-stage fine-tuning with BERT [41], pre-training BERT with various external signals [32,33,34], as well as combining BERT with existing neural models [37] or LTR methods [18]. Beyond that, two-tower retrievers [22,47,56], ColBERT [23], EPIC [36], TK [21], as well as PreTTR [35] pre-compute the passage representations to reduce query-time latency, and are further improved by TAS-Balanced [19], PAIR [48] and JPQ [60]. As can be seen, most of the mentioned BERT-based ranking models consider query-document pairs independently or use time-consuming pairwise loss, ignoring the ranking context based on more than two documents. There are also works that exploit PRF information to boost ranking. Padaki et al. [42] investigate several traditional keyword expansion approaches and find that they are not necessarily beneficial. Zheng et al. [62] propose BERT-QE that expands the original query by text snippets, instead of individual keywords, selected by a fine-tuned BERT ranker. Based on Transformer-XH [61], Yu et al. [58] propose the graph-based PGT model that utilizes a configurable number of feedback documents. PRF mechanism and query expansion approaches are also incorporated with dense retrievers to boost IR performances [50,53,59]. Unlike Co-BERT, the motivation of these works is to expand the queries to mitigate the vocabulary mismatch between queries and documents. Instead, Co-BERT aims to use the PRF signals to calibrate the relevance weightings for documents in different groups but from the same ranking, supplementing the groupwise scorer component in a light-weight fashion.

**Incorporating Ranking Context.** In early works, pairwise or listwise losses were used to learn from multiple documents [5,28,31,55]. Recently, the cross-document interactions are further incorporated into the ranking models. Ai et al. [1] employ a recurrent neural network to encode the top-ranked results, from which a context model learns to incorporate the query-specific feature distributions. They further develop a general framework for multivariate scoring functions, in which the relevance score of a document is determined by considering multiple other documents in the list [2]. Pasumarthi et al. [44] leverage the cross-document interaction by a self-attention based neural network, showing improved effectiveness and efficiency on several learning to rank (L2R) datasets. Pang et al. [43] propose a transformer-based L2R approach, SetRank, that directly learns a permutation-invariant ranking model defined on document sets. Very recently, Chen et al. [7] propose a listwise learning framework combining four pooling-based losses over three neural retrieval models. Feng et al. [14] apply Bi-LSTM and self-attention mechanism to model the contextual information to guide the generation of the recommendation results. Among these works, evaluation on learning to rank datasets shows performance gain of SetRank [43] over strong baselines. SetRank uses FNN to encode document features, and feeds the representations into Set-Transformer [26] to capture the local context information

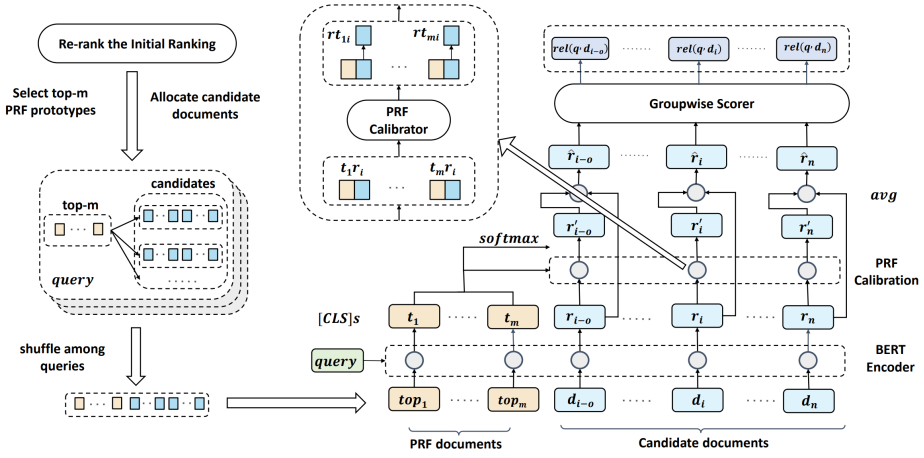


**Fig. 1.** Model architecture of Co-BERT.

from cross-document interactions, jointly scoring retrieved documents. However, when using cross-encoders, it becomes infeasible to put the BERT representations of the entire ranking list in the memory at one time. To the best of our knowledge, these existing models are built upon handcrafted features and do not have straightforward extensions to make uses of the recent pre-trained language models, e.g. BERT. This work extends this research direction for the state-of-the-art BERT-based re-ranker, proposing an end-to-end model that jointly learns the interaction representation and the ranking context.

## 3 Method

In this section, we present the Co-BERT model for document re-ranking, wherein the query-document interaction representations and the ranking context are learned jointly. The model architecture is summarized in Figure 1.

### 3.1 Overview

Given a query $q$ and $k$ ranked documents, e.g., from BM25, a re-ranking method aims to provide each document $d$ a relevance score $rel(q, d)$ that estimates to what degree document $d$ satisfies the query $q$. As shown in SetRank [43], referring to other candidate documents from the same ranking is important, wherein the query-document representations are in the form of handcrafted feature vectors before modeling the ranking context. Inspired by this, an end-to-end framework is proposed to boost the ranking of these $k$ documents by learning the query-document encoding and the ranking context together. The proposed model is composed of the cross-documents interaction encoder named groupwise scorer,

and the components to calibrate the relevance weighting among different batches for the same ranking using pseudo relevance feedback (PRF).

Recall that, when using cross attention to model the relevance with BERT [29], the token sequence from a query $q$ and from a document $d$ are first concatenated into $[CLS]Query[SEP]Document[SEP]$ before passing through multiple self-attention layers, and the interaction representation for $[CLS]$ is used to encode the relevance between the query and the document [13]. In the groupwise scorer, inspired by [2,43], instead of independently evaluating the relevance of individual documents, $n$ documents are considered together using a four-layers BERT model, and the relevance of these $n$ documents are evaluated jointly. Due to the huge amount of the trainable parameters in BERT, a ranking is split into multiple groups and a groupwise scorer models the documents within each group independently. To calibrate the relevance weighting among these groups, similar to [1], the top-$m$ out of the top-$k$ documents are used as pseudo relevance feedback (PRF) set, providing the query-specific context. When evaluating the relevance of a document, beyond directly using the interaction presentation $[CLS]$ between the token sequence from a document-query pair, we first use the $m$ interaction representations from PRF documents to calibrate it. Likewise in [12], since a document could be too long to be encoded using BERT model, we split a document into overlapped passages with the same length. Similar to BERT-QE [62], a BERT checkpoint pre-trained on MS MARCO [40] is used to score each passage relative to the query, and the passage with the highest score in each document is actually used in place of the original document in both training and inference. For brevity, we use the term "document" in the following.

### 3.2 End-to-end Groupwise Scorer

Given a query-document pair, the $[CLS]$ vector from the last layer of BERT is used as the query-document interaction representation. We denote the $k$ interaction vectors for the top-$k$ documents as $r_j$ where $j \in [1 \cdots k]$, each corresponds to one document to be evaluated, and $r_j$ is a $l$-dimension dense vector, e.g., $l = 768$ when using BERT-Base. As mentioned in [2], scoring individual documents independently could lead to sub-optimal ranker due to the comparing natural in the ranking problem. Inspired by SetRank [43], we propose a groupwise relevance scorer using BERT, hoping to evaluate the document relevance more effectively by encoding the cross-documents interactions from the same ranking. Due to the size of the BERT encoder, e.g., 110M parameters in BERT-Base, we group $n$ candidate documents ($n \leq k$) together, before modeling the relevance of these $n$ documents jointly. To maintain the cross-references among different groups, we employ a straightforward method by allowing an overlap with $o$ documents, in between neighbouring groups from the initial ranking. For the $n$ documents in a single group, their interaction representations are stacked into a sequence with length $n$, namely, $r_1, r_2, r_3, \cdots, r_n$. Thereafter, this sequence of interaction representations are passed through multiple layers of BERT, before being projected into $n$ relevance scores, which are used to rank the documents. Herein, a BERT model named *uncased_L-4_H-768_A-12*, with four layers, the hidden

size of 768, and, 12 attention heads, is used. We initialise this four-layers BERT model using pre-trained checkpoint from Google [16]. The choice of $n$ is up to the maximum batch size that is allowed by the hardware. Different from the existing ranking model incorporating ranking context, like SetRank [43], groupwise encoder takes the $[CLS]$ from the query-document encoding as input, enabling the end-to-end training of the query-document interaction representation and the ranking context modeling.

Recall that the transformer model [51] relies on the positional embedding to encode the position information. According to our pilot experiments, we do not configure the positional embedding within a group, and simply generate different groups following the initial ranking.

### 3.3 Light-Weight Pseudo Relevance Feedback

As mentioned, there exist multiple groups when modeling the ranking including many documents, namely, $n \leq k$. In this section, we further introduce a novel building block using PRF information to calibrate relevance weighting among different groups from the same ranking.

The top-$m$ documents are selected as the pseudo relevance feedback (PRF) set, which are used to provide the query-specific context among different groups. We first construct prototype representation for the interaction representations using these $m$ PRF documents. Similar to the computation of each $r_j$, the $m$ output embedding of the token $[CLS]$ from BERT, each for one of the PRF documents, encode the interaction between the query and the corresponding PRF set. In favor of the description, we denote these $m$ $[CLS]$ vectors as $t_i$ instead of using $r_\cdot$, where $i \in [1, \cdots, m]$. Thereafter, the $k$ interaction vectors from Section 3.2 are calibrated using these $m$ prototypes $t_i$ with a shallow BERT model of two layers before passing through the groupwise scorer. In particular, the interaction prototype $t_i$ and each interaction representation $r_j$ are stacked into a sequence with two tokens, namely, $t_i r_j$, before passing through the two-layer BERT. The calibrated interaction representation corresponding to $r_j$ using prototype $t_i$ from the two-layers BERT output sequence is denoted as $rt_{ij}$. Thereby, for each $r_j$, there are $m$ calibrated representations. Ultimately, we combine these $m$ calibrated presentations into one using a simple weighted average, where the weight is the relevance of the prototype $t_i$, as in Eq. 1, and $W_t$ and $b_t$ are trainable weights for the projection. Similar to the residual connection in the multi-head attention [51], as shown in Eq. 2, we average the calibrated interaction representation and the origin presentation and use the resulting vector as the inputs for the follow-up scorer. We show that this residual connection is important to the effectiveness in Section 5. In this work, for the two-layers BERT model in the calibration, we employ the configuration named *uncased_L-2_H-768_A-12*, which is with two layers, hidden size equaling 768, and 12 attention heads. We use the pre-trained BERT checkpoint from Google [15] to initialise this model.

$$r'_j = \sum_{i \in [1 \cdot m]} softmax(W_t t_i + b_t) \cdot rt_{ij} \tag{1}$$

$$\hat{r}_j = \frac{r_j + r'_j}{2} \tag{2}$$

### 3.4 End-to-end Training of the Model

Given a query $q$ and $k$ documents, we first select $m$ PRF documents using BERT ranker pre-trained on MS Marco [40]. Thereafter, the batch size is determined based on the constrains of GPU hardware. Therein, in each batch, $n$ candidate documents, together with the $m$ PRF documents are batched together. During training, cross-entropy loss is computed for individual documents as in Eq. 3, where $I_{pos}$ and $I_{neg}$ denote the sets of indexes for relevant and non-relevant documents, respectively, and $pr_j$ is the probability of the document $j$ being relevant according to the model. The probability is computed using a *softmax* function, namely, $pr_j = softmax(rel(q, d))$, where $rel(q, d)$ is the relevance score of $d$ given by Co-BERT.

$$\mathcal{L}(I_{pos}, I_{neg}, q, d_j) = -\sum_{j \in I_{pos}} \log(pr_j) - \sum_{j \in I_{neg}} \log(1 - pr_j) \tag{3}$$

Note that, we use pointwise loss as in Eq. 3 to train the groupwise scorer and leave the study of other losses to future work. The cross-documents interaction is implemented using the four-layers BERT-based groupwise scorer described in Section 3.2 and the two-layers BERT-based calibrator in Section 3.3.

## 4 Experiment Setup

### 4.1 Dataset and Metrics

We experiment on the widely-used Robust04 [52], GOV2 [8], and ClueWeb09-B [9] datasets. We employ 249 title queries for Robust04, 150 title queries for GOV2, and 200 title queries for ClueWeb09-B. Since we have similar observations on NDCG@20 and P@20, we report P@20 to enable the comparisons on the shallow pool; and MAP@1K is reported for deep pool. All statistical tests are based on the paired t-tests at $p < 0.05$ with corrections [6].

### 4.2 Baselines and Co-BERT variants

**DPH+KL**, the unsupervised DPH retrieval model [4] with Rocchio's query expansion using KL divergence [3,49] is used to generate the **initial ranking** of top-1k documents. The implementation from Terrier [38] has been adopted.
**BM25+RM3** is another unsupervised ranking model using pseudo relevance feedback signals [25]. We follow the experimental settings from [57], and the implementation from Anserini [30] with default settings is used.
**BERT-Base** is the BERT-Base ranker boosted by transfer learning. The model is initialised using a checkpoint that has been trained on MS Marco [40], before

being fine-tuned on target datasets using the top-1 passage from each relevant document as positive examples as in [62].

**BERT-Groupwise** is a multi-stage training method. Since there has been no existing work trying to integrate pre-trained language model encoders and groupwise methods, we implement the model by directly combining a SetRank [43] like groupwise model of four-layers BERT with a BERT-based encoder. Text representation of BERT-Base is saved before training groupwise scorer. Due to the pre-storage of text representation, the batch size of training is expanded to 500. Other configurations are similar to Co-BERT.

**duoBERT [41]** is a pairwise BERT re-ranker initialised using a BERT-Base checkpoint trained on MS Marco, which follows the default setting of the top-30 BERT-Base re-ranking and 512 sequence length.

**PGT [58]** is a pseudo relevance feedback method that uses a graph-based Transformer. In addition to the results on TREC 19&20 Deep Learning Track [10,11] as in Table 3, we also report our implementation on the other datasets.

**BERT-QE [62]** is a BERT re-ranking model exploiting the PRF signals. Unlike Co-BERT, BERT-QE is an inference framework and has not been trained end-to-end. In this work, to enable comparisons, we use the BERT-QE variances using three BERT-Base components (namely, BERT-QE-BBB), each for one of its phases. For fair comparisons, we re-implement BERT-QE with the same passage slicing and the same max sequence length as Co-BERT.

The following **variants of Co-BERT** are included for comparisons.

**Co-BERT** is the model as described in Section 3 using BERT-based groupwise scorer on top of the calibrated interaction representations based on PRF.

**Co-BERT with PRF calibrator only** is a variant of Co-BERT. The relevance of documents are evaluated independently using Eq. 4 without passing the batch of calibrated interaction representations into the groupwise scorer. In particular, we simply project individual $\hat{r}_j$ from Eq. 2 into a relevance score using a shared trainable weights $W_{rel}$ and $b_{rel}$ for each of the $k$ documents, as in Eq. 4.

$$rel(q, \mathcal{R}_m, d_j) = W_{rel}\hat{r}_j + b_{rel} \tag{4}$$

**Co-BERT with groupwise scorer only** is another variant of Co-BERT without using the PRF calibration, and only use the groupwise scorer described in Section 3.2. This means we do not use any feedback signals in the re-ranking, but still use the groupwise scorer for training and inference.

Note that, the efficient design in dense retrieval and contrastive learning [47,56] are deemed orthogonal to the use of the ranking context, and the dense retrieval models thus have not been included for comparisons. Moreover, the results for the baselines and the Co-BERT variants are based on the standalone ranking models *without* the interpolation with the unsupervised ranking score.

### 4.3 Model Training and Inference

**Data preparation.** Both training and inference are based on the top-1k documents from DPH+KL. Akin to [12], for BERT-Base, PGT, BERT-QE and

**Table 1.** Effectiveness of Co-BERT relative to baseline models. The gain/loss is reported relative to BERT-Base, on top of which the Co-BERT network architecture is established. The statistical significance at 0.05 relative to (PRF only), (groupwise only), and Co-BERT are denoted as †, ‡ and §, respectively.

| Model | Robust04 | | | Gov2 | | | ClueWeb09-B | | |
|---|---|---|---|---|---|---|---|---|---|
| | P@20 | MAP@1k | FPs | P@20 | MAP@1k | FPs | P@20 | MAP@1K | FPs |
| BM25+RM3 [30] | 0.3821 | 0.2903 | - | 0.5634 | 0.3350 | - | 0.2669 | 0.1819 | - |
| DPH+KL [38] | 0.3924 | 0.3046 | - | 0.5896 | 0.3605 | - | 0.2962 | 0.2019 | - |
| BERT-Base | $0.4430^{\S}$ | $0.3407^{\S}$ | +0% | $0.5725^{\ddagger\S}$ | $0.3531^{\ddagger\S}$ | +0% | $0.3285^{\S}$ | $0.2171^{\ddagger\S}$ | +0% |
| BERT-Groupwise | $0.4436^{\S}$ | $0.3408^{\S}$ | +0.3% | $0.5889^{\ddagger\S}$ | $0.3567^{\ddagger\S}$ | +0.1% | 0.3343 | $0.2223^{\ddagger\S}$ | +0.5% |
| duoBERT [41] | $0.4293^{\dagger\ddagger\S}$ | $0.3173^{\dagger\ddagger\S}$ | +14.6% | $0.5923^{\ddagger\S}$ | $0.3553^{\ddagger\S}$ | +3.5% | $0.3323^{\S}$ | $0.2163^{\ddagger\S}$ | +10.3% |
| PGT [58] | $0.4131^{\dagger\ddagger\S}$ | $0.3085^{\dagger\ddagger\S}$ | +50.1% | $0.5859^{\ddagger\S}$ | $0.3144^{\dagger\ddagger\S}$ | +12.0% | $0.2833^{\dagger\ddagger\S}$ | $0.1736^{\dagger\ddagger\S}$ | +35.4% |
| BERT-QE [62] | 0.4614 | 0.3555 | +86.1% | $0.6198^{\S}$ | $0.3662^{\ddagger\S}$ | +20.5% | $0.3152^{\ddagger\S}$ | $0.2131^{\ddagger\S}$ | +60.7% |
| (PRF only) | 0.4526 | 0.3480 | +1.3% | 0.5802 | 0.3550 | +0.3% | 0.3273 | 0.2153 | +1.0% |
| | (+2.2%) | (+2.1%) | - | (+1.3%) | (+0.5%) | - | (-0.4%) | (-0.8%) | - |
| (groupwise only) | 0.4500 | 0.3530 | +0.3% | 0.6493 | 0.3993 | +0.1% | 0.3457 | 0.2418 | +0.5% |
| | (+1.6%) | (+3.6%) | - | (+13.4%) | (+13.1%) | - | (+5.2%) | (+11.4%) | - |
| Co-BERT | **0.4629** | **0.3631** | +1.3% | **0.6668** | **0.4022** | +0.3% | **0.3598** | **0.2463** | +1.0% |
| | **(+4.5%)** | **(+6.6%)** | - | **(+16.5%)** | **(+13.9%)** | - | **(+9.5%)** | **(+13.5%)** | - |

Co-BERT, the documents are chunked using sliding windows of 150 words with an overlap of 75 words. As mentioned in Section 3.1, for all four models, the most relevant passage is selected using a BERT ranker pre-trained on MS Marco [40] to represent individual documents. To feed individual query-paragraph (i.e. the text chunk with 150 words) pairs into the model, query and paragraph are concatenated with a maximum sequence length of 256.

**Batching and loss function.** We train BERT-Base and Co-BERT using cross-entropy loss as in Eq. 3 for five epochs with a batch size of 64 on one NVIDIA TITAN RTX 24G. For Co-BERT, according to preliminary results, we configure the number of PRF documents for calibration as $m = 4$, the number of candidate documents in individual group as 60 ($n = 60$), and the overlap between the neighbouring groups is set to four ($o = 4$). During training, we randomly shuffle the batches before feeding them into the model. The Adam optimizer [24] is used with the learning rate schedule from [40]. We configure the initial learning rate as 3e-6, and the warming up steps are set to the 10% of the total training steps.

**Cross-validation.** Similar to the configuration in DRMM [17], we use 5-fold cross-validation to report the results with a 3-1-1 split. The query partition on Robust04 follows the settings from [12]. On GOV2 and ClueWeb09-B, queries are partitioned by the order of TREC query id in a round-robin manner. The average performance on the test splits from all folds is reported.

## 5 Results

In this section, we examine the effectiveness and efficiency of Co-BERT relative to baseline models, before studying how groupwise mechanism and PRF calibrator work with BERT. Finally, we report the results on the TREC Deep Learning track query sets [10,11] for further comparisons.

**Table 2.** Impacts of the residual connections in Eq. 2. Two alternative feeding orders of batches during training are also investigated. Relative comparison in terms of percentage (in bracket) in comparisons with BERT-Base is also reported. Statistical significance at levels 0.05 is denoted with † and ‡, relative to BERT-Base and Co-BERT, respectively.

| Model | Robust04 | | | Gov2 | | |
|---|---|---|---|---|---|---|
| | P@20 | NDCG@20 | MAP@1K | P@20 | NDCG@20 | MAP@1K |
| BERT-Base | 0.4430 | 0.5109 | 0.3407 | 0.5725 | 0.5040 | 0.3531 |
| Co-BERT (Random training) | **0.4629** | **0.5213** | **0.3631** | **0.6668** | **0.5781** | **0.4022** |
| (w/o residual connection in Eq. 2) | 0.4554 | 0.5102 | $0.3567^{\dagger}$ | $0.6326^{\dagger\ddagger}$ | $0.5484^{\dagger\ddagger}$ | $0.3951^{\dagger}$ |
| | (-1.6%) | (-2.2%) | (-1.9%) | (-6.0%) | (-5.9%) | (-2.0%) |
| (Train following initial ranking) | 0.4422 | 0.5029 | $0.3457^{\ddagger}$ | $0.6211^{\dagger\ddagger}$ | $0.5308^{\dagger\ddagger}$ | $0.3728^{\dagger\ddagger}$ |
| | (-4.7%) | (-3.6%) | (-5.1%) | (-8.0%) | (-9.4%) | (-8.3%) |
| (Train reversing initial ranking) | 0.4454 | 0.5026 | $0.3429^{\ddagger}$ | $0.6322^{\dagger\ddagger}$ | $0.5429^{\dagger\ddagger}$ | $0.3799^{\dagger\ddagger}$ |
| | (-4.0%) | (-3.7%) | (-5.9%) | (-6.0%) | (-7.0%) | (-6.3%) |

### 5.1 Overall Performance of Co-BERT

Given a query, different BERT-based ranking models, including the variants of Co-BERT model described in Section 4, are used to re-rank the top-1k documents from DPH+KL. We also include two classical unsupervised ranking models, namely, BM25+RM3 and DPH+KL, for references. The ranking effectiveness are summarised on both shallow (P@20) and deep pool (MAP@1K) in Table 1.

**Effectiveness of Co-BERT.** According to Table 1, Co-BERT outperforms all of the unsupervised baselines. As both BERT-Base and Co-BERT have been initialised using the ranking model pre-trained on MS Marco [40], and are fine-tuned in the same way. Thereby, we are assured that the performance difference between Co-BERT and BERT-Base comes from the novel model architecture introduced in Section 3. Actually, Co-BERT also achieves better results than the most recent transformer-based ranking models using PRF signals and query expansion such as PGT [58] and BERT-QE [62], confirming the superior effectiveness of the complete Co-BERT, especially on the deep pool.

**Efficiency of Co-BERT.** The FLOPs, i.e. the number of floating point operations, of various BERT-based models are reported in Table 1, in the form of the relative comparisons to BERT-Base. From Table 1, comparing with BERT-Base, it can be seen that Co-BERT only requires an extra 1.3% computation overheads when significantly boosting the effectiveness on both shallow (4.5%) and deep pool (6.6%) on Robust04; meanwhile, with only 0.3% extra computation cost, Co-BERT could provide more than 13% boosts on both shallow and deep pools on GOV2. Remarkably, though being able to outperform BERT-Base in most cases, the extra computation cost of Co-BERT is actually limited.

### 5.2 Study of Groupwise Ranking

**End-to-End training plays an important role.** As shown in Table 1, although the batch size is large, the detached groupwise architecture of BERT-Groupwise shows little benefit on Robust04 when compared to BERT-Base, and

can only achieve marginal improvements on GOV2 and ClueWeb09-B. However, Co-BERT with groupwise scorer only, using the same model component as BERT-Groupwise but with end-to-end training, can significantly improve the performances compared to BERT-Base. On the deep pool in terms of MAP@1K, more than 13% boosts have been observed on both GOV2 and ClueWeb09-B. On shallow pool, the end-to-end training method can also improve P@20 by 1.6%, 13.4% and 5.2%, on the three datasets used, respectively. Recall that BERT-Groupwise attempts to apply groupwise scorer directly to the text representation using a SetRank-like approach. When the query-document representation pre-generated by a fine-tuned BERT ranker is used for groupwise scorer training, the effectiveness of groupwise ranking is limited, demonstrating the importance of the end-to-end training for the BERT-based groupwise ranker.

**Impacts of feeding order.** As mentioned in Section 3.4, when the total number of documents for ranking (namely, $k$) is too large to be fed into single batch, we have to group $n < k$ documents into batches during training and inference, and then feed the data for training after random shuffling. We investigate two alternative ways for the feeding order of training data, namely, training following initial ranking and training reversing initial ranking. Training following initial ranking means when feeding training batches for the same query, the batches are ordered following the initial ranking. On the contrary, when training following the reversed order in initial ranking, the batches are fed in the reversed order of the initial ranking. Note that, among different epochs, the training data is still shuffled among queries to avoid over-fitting. For brevity, we only report results from Robust04 and Gov2, as results obtained on ClueWeb09-B and Gov2 lead to similar observations. According to the results in Table 2, it can be seen that, with the alternative feeding order for the training data, Co-BERT could still outperform BERT-Base on GOV2. Such alternative order, however, leads to at least 3.5% drops among all different metrics on both dataset and the resulting models are significantly worse than Co-BERT trained using fully shuffled batches.

### 5.3 Study of Light-Weight PRF Calibrator

As can be seen in Table 1, when only using the PRF calibrator without the groupwise scorer, on Robust04, the PRF-calibrator-only variant can outperform BERT-Base with up to 2% margin on two metrics. While on GOV2 and ClueWeb09-B, Co-BERT does not show advantage over BERT-Base. However, when being used with groupwise scorer, namely the full Co-BERT, the PRF calibrator is able to further enhance the effectiveness, although groupwise has already made a significant improvement over BERT-Base. Recall that the purpose of the PRF calibrator is to provide a lightweight performance boost to the groupwise BERT scorer. The above findings confirm the ability of the PRF calibrator in improving the groupwise BERT ranker with relatively low extra computational overhead as shown in Section 5.1. Moreover, as described in Section 3.3, the averaging operation in Eq. 2 adds back the origin interaction representation after the PRF calibration, providing more direct connections between early layers and the scorer layers. In Table 2, we report the results of Robust04 and

**Table 3.** Effectiveness of Co-BERT on TREC DL query sets.

| Model | TREC DL 19 | | | TREC DL 20 | | |
|---|---|---|---|---|---|---|
| | MRR@10 | NDCG@10 | MAP@1K | MRR@10 | NDCG@10 | MAP@1K |
| BERT-Base | 0.9280 | **0.6999** | 0.4715 | 0.7847 | 0.6776 | **0.4553** |
| PGT [58] | 0.9297 | 0.6938 | 0.4232 | 0.8108 | 0.6818 | 0.4184 |
| Co-BERT | **0.9581** | 0.6996 | **0.4838** | **0.8391** | **0.6992** | 0.4505 |

Gov2 without the averaging operation, the performances of Co-BERT drops on all metrics of the results. The same phenomenon is also observed on ClueWeb09-B. This highlights the importance to add this skip connection after calibrating the interaction representation using pseudo relevance feedback.

### 5.4 Effectiveness on TREC DL

We additionally report the results on the TREC Deep Learning track query sets [10,11] using the MS MARCO passage corpus [39]. TREC DL 19 & 20 contains 43 and 54 queries respectively, which are manually annotated by NIST on a four-point scale. As the MS Marco document set is similar to the passage set in nature, we only report on the latter for brevity. BM25 is used as the initial ranker and the official metrics, MRR@10, NDCG@10 and MAP@1k, are reported. We compare our model with BERT-Base and PGT [58] by re-ranking the top-1000 documents from BM25. According to the results in Table 3, Co-BERT obtains higher scores than PGT in all metrics, however, Co-BERT's performance is overall comparable to BERT-Base. A likely cause for the insignificant difference between Co-BERT and BERT-base is the QA-oriented nature of the MS Marco dataset, which normally has only one prototype answer for a given question. Due to the lack of diversity in the relevant passages for each query, groupwise ranking may not benefit from highlighting different relevant content by the cross-attention.

## 6 Conclusion

In this paper, we propose an end-to-end BERT-based re-ranking models, named Co-BERT, wherein the relevances of a group of documents are modeled jointly. Evaluation on three standard TREC test collections, namely, Robust04, GOV2, and Clueweb09-B, demonstrates that the proposed Co-BERT could advance the state-of-the-art BERT-based ranking model by a considerable margin. In addition, the results highlight the importance of the end-to-end training of a groupwise BERT ranker, as opposed to the groupwise ranking over the pre-trained text representation using a SetRank-like approach. Finally, the lightweight PRF calibrator is shown to be able to provide a further performance boost over the groupwise ranker with small extra computation overhead.

**Acknowledgements**. This work is supported by National Key R&D Program of China (2020AAA0105200).

## References


1. Ai, Q., Bi, K., Guo, J., Croft, W.B.: Learning a deep listwise context model for ranking refinement. In: SIGIR. pp. 135–144. ACM (2018)
2. Ai, Q., Wang, X., Bruch, S., Golbandi, N., Bendersky, M., Najork, M.: Learning groupwise multivariate scoring functions using deep neural networks. In: ICTIR. pp. 85–92. ACM (2019)
3. Amati, G.: Probability models for information retrieval based on divergence from randomness. Ph.D. thesis, University of Glasgow, UK (2003)
4. Amati, G., Ambrosi, E., Bianchi, M., Gaibisso, C., Gambosi, G.: Fub, IASI-CNR and university of tor vergata at TREC 2007 blog track. In: Proceedings of The Sixteenth Text REtrieval Conference. NIST Special Publication, vol. 500-274, pp. 1–10. National Institute of Standards and Technology (2007)
5. Cao, Z., Qin, T., Liu, T.Y., Tsai, M.F., Li, H.: Learning to rank: from pairwise approach to listwise approach. In: Proceedings of the 24th international conference on Machine learning. pp. 129–136 (2007)
6. Carterette, B.A.: Multiple testing in statistical analysis of systems-based information retrieval experiments. ACM Trans. Inf. Syst. **30**(1), 4:1–4:34 (2012). https://doi.org/10.1145/2094072.2094076, https://doi.org/10.1145/2094072.2094076
7. Chen, Z., Eickhoff, C.: Poolrank: Max/min pooling-based ranking loss for listwise learning & ranking balance. CoRR **abs/2108.03586** (2021), https://arxiv.org/abs/2108.03586
8. Clarke, C.L.A., Craswell, N., Soboroff, I.: Overview of the TREC 2004 terabyte track. In: Proceedings of the Thirteenth Text REtrieval Conference. NIST Special Publication, vol. 500-261, pp. 1–9. National Institute of Standards and Technology (2004)
9. Clarke, C.L.A., Craswell, N., Soboroff, I.: Overview of the TREC 2009 web track. In: Voorhees, E.M., Buckland, L.P. (eds.) Proceedings of The Eighteenth Text REtrieval Conference, TREC 2009, Gaithersburg, Maryland, USA, November 17-20, 2009. NIST Special Publication, vol. 500-278. National Institute of Standards and Technology (NIST) (2009), http://trec.nist.gov/pubs/trec18/papers/WEB09.OVERVIEW.pdf
10. Craswell, N., Mitra, B., Yilmaz, E., Campos, D.: Overview of the TREC 2020 deep learning track. CoRR **abs/2102.07662** (2021), https://arxiv.org/abs/2102.07662
11. Craswell, N., Mitra, B., Yilmaz, E., Campos, D., Voorhees, E.M.: Overview of the TREC 2019 deep learning track. CoRR **abs/2003.07820** (2020), https://arxiv.org/abs/2003.07820
12. Dai, Z., Callan, J.: Deeper text understanding for IR with contextual neural language modeling. In: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 985–988. ACM (2019)
13. Devlin, J., Chang, M., Lee, K., Toutanova, K.: 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. pp. 4171–4186. Association for Computational Linguistics (2019)
14. Feng, Y., Hu, B., Gong, Y., Sun, F., Liu, Q., Ou, W.: GRN: generative rerank network for context-wise recommendation. CoRR **abs/2104.00860** (2021), https://arxiv.org/abs/2104.00860

15. Google-Research: bert_uncased_L-2_H-768_A-12 (2020), https://storage.googleapis.com/bert_models/2020_02_20/uncased_L-2_H-768_A-12.zip
16. Google-Research: bert_uncased_L-4_H-768_A-12 (2020), https://storage.googleapis.com/bert_models/2020_02_20/uncased_L-4_H-768_A-12.zip
17. Guo, J., Fan, Y., Ai, Q., Croft, W.B.: A deep relevance matching model for ad-hoc retrieval. In: Proceedings of the 25th ACM International Conference on Information and Knowledge Management. pp. 55–64. ACM (2016)
18. Han, S., Wang, X., Bendersky, M., Najork, M.: Learning-to-rank with BERT in tf-ranking. CoRR **abs/2004.08476** (2020)
19. Hofstätter, S., Lin, S., Yang, J., Lin, J., Hanbury, A.: Efficiently teaching an effective dense retriever with balanced topic aware sampling. In: Diaz, F., Shah, C., Suel, T., Castells, P., Jones, R., Sakai, T. (eds.) SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021. pp. 113–122. ACM (2021). https://doi.org/10.1145/3404835.3462891, https://doi.org/10.1145/3404835.3462891
20. Hofstätter, S., Mitra, B., Zamani, H., Craswell, N., Hanbury, A.: Intra-document cascading: Learning to select passages for neural document ranking. In: Diaz, F., Shah, C., Suel, T., Castells, P., Jones, R., Sakai, T. (eds.) SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021. pp. 1349–1358. ACM (2021). https://doi.org/10.1145/3404835.3462889, https://doi.org/10.1145/3404835.3462889
21. Hofstätter, S., Zlabinger, M., Hanbury, A.: Interpretable & time-budget-constrained contextualization for re-ranking. In: 24th European Conference on Artificial Intelligence. Frontiers in Artificial Intelligence and Applications, vol. 325, pp. 513–520. IOS Press (2020)
22. Karpukhin, V., Oguz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., Yih, W.t.: Dense passage retrieval for open-domain question answering. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 6769–6781 (2020)
23. Khattab, O., Zaharia, M.: Colbert: Efficient and effective passage search via contextualized late interaction over BERT. In: Proceedings of the 43rd International ACM SIGIR conference on Research and Development in Information Retrieval. pp. 39–48. ACM (2020)
24. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: 3rd International Conference on Learning Representations. pp. 1–15 (2015)
25. Lavrenko, V., Croft, W.B.: Relevance-based language models. In: Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 120–127. ACM (2001)
26. Lee, J., Lee, Y., Kim, J., Kosiorek, A.R., Choi, S., Teh, Y.W.: Set transformer: A framework for attention-based permutation-invariant neural networks. In: Chaudhuri, K., Salakhutdinov, R. (eds.) Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA. Proceedings of Machine Learning Research, vol. 97, pp. 3744–3753. PMLR (2019), http://proceedings.mlr.press/v97/lee19d.html
27. Li, C., Yates, A., MacAvaney, S., He, B., Sun, Y.: PARADE: passage representation aggregation for document reranking. CoRR **abs/2008.09093** (2020), https://arxiv.org/abs/2008.09093
28. Li, H.: A short introduction to learning to rank. IEICE TRANSACTIONS on Information and Systems **94**(10), 1854–1862 (2011)

29. Lin, J., Nogueira, R., Yates, A.: Pretrained transformers for text ranking: BERT and beyond. CoRR **abs/2010.06467** (2020)
30. Lin, J.J., Crane, M., Trotman, A., Callan, J., Chattopadhyaya, I., Foley, J., Ingersoll, G., MacDonald, C., Vigna, S.: Toward reproducible baselines: The open-source IR reproducibility challenge. In: Advances in Information Retrieval - 38th European Conference on IR Research. Lecture Notes in Computer Science, vol. 9626, pp. 408–420. Springer (2016)
31. Liu, T., Joachims, T., Li, H., Zhai, C.: Introduction to special issue on learning to rank for information retrieval. Inf. Retr. **13**(3), 197–200 (2010)
32. Ma, X., Guo, J., Zhang, R., Fan, Y., Ji, X., Cheng, X.: PROP: pre-training with representative words prediction for ad-hoc retrieval. In: Lewin-Eytan, L., Carmel, D., Yom-Tov, E., Agichtein, E., Gabrilovich, E. (eds.) WSDM '21, The Fourteenth ACM International Conference on Web Search and Data Mining, Virtual Event, Israel, March 8-12, 2021. pp. 283–291. ACM (2021). https://doi.org/10.1145/3437963.3441777, https://doi.org/10.1145/3437963.3441777
33. Ma, X., Guo, J., Zhang, R., Fan, Y., Li, Y., Cheng, X.: B-PROP: bootstrapped pre-training with representative words prediction for ad-hoc retrieval. In: Diaz, F., Shah, C., Suel, T., Castells, P., Jones, R., Sakai, T. (eds.) SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021. pp. 1318–1327. ACM (2021). https://doi.org/10.1145/3404835.3462869, https://doi.org/10.1145/3404835.3462869
34. Ma, Z., Dou, Z., Xu, W., Zhang, X., Jiang, H., Cao, Z., Wen, J.: Pre-training for ad-hoc retrieval: Hyperlink is also you need. CoRR **abs/2108.09346** (2021), https://arxiv.org/abs/2108.09346
35. MacAvaney, S., Nardini, F.M., Perego, R., Tonellotto, N., Goharian, N., Frieder, O.: Efficient document re-ranking for transformers by precomputing term representations. In: Proceedings of the 43rd International ACM SIGIR conference on Research and Development in Information Retrieval. pp. 49–58. ACM (2020)
36. MacAvaney, S., Nardini, F.M., Perego, R., Tonellotto, N., Goharian, N., Frieder, O.: Expansion via prediction of importance with contextualization. In: Proceedings of the 43rd International ACM SIGIR conference on Research and Development in Information Retrieval. pp. 1573–1576. ACM (2020)
37. MacAvaney, S., Yates, A., Cohan, A., Goharian, N.: CEDR: contextualized embeddings for document ranking. In: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 1101–1104. ACM (2019)
38. Macdonald, C., McCreadie, R., Santos, R.L.T., Ounis, I.: From puppy to maturity: Experiences in developing terrier. In: Proceedings of the SIGIR 2012 Workshop on Open Source Information Retrieval. pp. 60–63. University of Otago, Dunedin, New Zealand (2012)
39. Nguyen, T., Rosenberg, M., Song, X., Gao, J., Tiwary, S., Majumder, R., Deng, L.: Ms marco: A human generated machine reading comprehension dataset. In: CoCo@ NIPS (2016)
40. Nogueira, R., Cho, K.: Passage re-ranking with BERT. CoRR **abs/1901.04085** (2019)
41. Nogueira, R., Yang, W., Cho, K., Lin, J.: Multi-stage document ranking with BERT. CoRR **abs/1910.14424** (2019)
42. Padaki, R., Dai, Z., Callan, J.: Rethinking query expansion for bert reranking. In: European Conference on Information Retrieval. pp. 297–304. Springer (2020)

43. Pang, L., Xu, J., Ai, Q., Lan, Y., Cheng, X., Wen, J.: Setrank: Learning a permutation-invariant ranking model for information retrieval. In: SIGIR. pp. 499–508. ACM (2020)
44. Pasumarthi, R.K., Wang, X., Bendersky, M., Najork, M.: Self-attentive document interaction networks for permutation equivariant ranking. CoRR **abs/1910.09676** (2019)
45. Qiao, Y., Xiong, C., Liu, Z., Liu, Z.: Understanding the behaviors of bert in ranking. ArXiv **abs/1904.07531** (2019)
46. Qin, T., Liu, T.Y., Xu, J., Li, H.: Letor: A benchmark collection for research on learning to rank for information retrieval. Information Retrieval **13**(4), 346–374 (2010)
47. Qu, Y., Ding, Y., Liu, J., Liu, K., Ren, R., Zhao, W.X., Dong, D., Wu, H., Wang, H.: Rocketqa: An optimized training approach to dense passage retrieval for open-domain question answering. In: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 5835–5847 (2021)
48. Ren, R., Lv, S., Qu, Y., Liu, J., Zhao, W.X., She, Q., Wu, H., Wang, H., Wen, J.: PAIR: leveraging passage-centric similarity relation for improving dense passage retrieval. In: Zong, C., Xia, F., Li, W., Navigli, R. (eds.) Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021. Findings of ACL, vol. ACL/IJCNLP 2021, pp. 2173–2183. Association for Computational Linguistics (2021). https://doi.org/10.18653/v1/2021.findings-acl.191, https://doi.org/10.18653/v1/2021.findings-acl.191
49. Rocchio, J.: Relevance feedback in information retrieval. In: The SMART retrieval system: experiments in automatic document processing, pp. 313–323. Prentice Hall, Englewood, Cliffs, New Jersey (1971)
50. Tang, H., Sun, X., Jin, B., Wang, J., Zhang, F., Wu, W.: Improving document representations by generating pseudo query embeddings for dense retrieval. In: Zong, C., Xia, F., Li, W., Navigli, R. (eds.) Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021. pp. 5054–5064. Association for Computational Linguistics (2021). https://doi.org/10.18653/v1/2021.acl-long.392, https://doi.org/10.18653/v1/2021.acl-long.392
51. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: NIPS. pp. 5998–6008 (2017)
52. Voorhees, E.M.: Overview of the TREC 2004 robust track. In: Proceedings of the Thirteenth Text REtrieval Conference. NIST Special Publication, vol. 500-261, pp. 1–10. National Institute of Standards and Technology (2004)
53. Wang, X., Macdonald, C., Tonellotto, N., Ounis, I.: Pseudo-relevance feedback for multiple representation dense retrieval. In: Hasibi, F., Fang, Y., Aizawa, A. (eds.) ICTIR '21: The 2021 ACM SIGIR International Conference on the Theory of Information Retrieval, Virtual Event, Canada, July 11, 2021. pp. 297–306. ACM (2021). https://doi.org/10.1145/3471158.3472250, https://doi.org/10.1145/3471158.3472250
54. Wu, Z., Mao, J., Liu, Y., Zhan, J., Zheng, Y., Zhang, M., Ma, S.: Leveraging passage-level cumulative gain for document ranking. In: The Web Conference 2020. pp. 2421–2431. ACM / IW3C2 (2020)
55. Xia, F., Liu, T.Y., Wang, J., Zhang, W., Li, H.: Listwise approach to learning to rank: theory and algorithm. In: Proceedings of the 25th international conference on Machine learning. pp. 1192–1199 (2008)

56. Xiong, L., Xiong, C., Li, Y., Tang, K.F., Liu, J., Bennett, P., Ahmed, J., Overwijk, A.: Approximate nearest neighbor negative contrastive learning for dense text retrieval. arXiv preprint arXiv:2007.00808 (2020)
57. Yilmaz, Z.A., Yang, W., Zhang, H., Lin, J.: Cross-domain modeling of sentence-level evidence for document retrieval. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing. pp. 3488–3494. Association for Computational Linguistics (2019)
58. Yu, H., Dai, Z., Callan, J.: PGT: pseudo relevance feedback using a graph-based transformer. CoRR **abs/2101.07918** (2021)
59. Yu, H., Xiong, C., Callan, J.: Improving query representations for dense retrieval with pseudo relevance feedback. CoRR **abs/2108.13454** (2021), https://arxiv.org/abs/2108.13454
60. Zhan, J., Mao, J., Liu, Y., Guo, J., Zhang, M., Ma, S.: Jointly optimizing query encoder and product quantization to improve retrieval performance. CoRR **abs/2108.00644** (2021), https://arxiv.org/abs/2108.00644
61. Zhao, C., Xiong, C., Rosset, C., Song, X., Bennett, P.N., Tiwary, S.: Transformer-xh: Multi-evidence reasoning with extra hop attention. In: 8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020. OpenReview.net (2020), https://openreview.net/forum?id=r1eIiCNYwS
62. Zheng, Z., Hui, K., He, B., Han, X., Sun, L., Yates, A.: Bert-qe: Contextualized query expansion for document re-ranking. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings. pp. 4718–4728 (2020)