ISSN 1000-9000(Print)
/1860-4749(Online)
CODEN JCTEEM

JCST

Vol.37 No.4 July 2022

COMPUTER

Journal of Computer Science & Technology

SPONSORED BY INSTITUTE OF COMPUTING TECHNOLOGY
THE CHINESE ACADEMY OF SCIENCES &
CHINA COMPUTER FEDERATION
SUPPORTED BY NSFC
CO-PUBLISHED BY SCIENCE PRESS &
SPRINGER

Wu ZJ, Liu YQ, Mao JX *et al.* Leveraging document-level and query-level passage cumulative gain for document ranking. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 37(4): 814–838 July 2022. DOI 10.1007/s11390-022-2031-y

Leveraging Document-Level and Query-Level Passage Cumulative Gain for Document Ranking

Zhi-Jing Wu[1,2] (吴之璟), Yi-Qun Liu[1,2,*] (刘奕群), *Distinguished Member, ACM, CCF, Senior Member, IEEE*
Jia-Xin Mao[3] (毛佳昕), *Member, ACM, CCF*, Min Zhang[1,2] (张 敏), *Senior Member, ACM, CCF*, and
Shao-Ping Ma[1,2] (马少平)

[1] *Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China*
[2] *Beijing National Research Center for Information Science and Technology, Tsinghua University, Beijing 100084, China*
[3] *Gaoling School of Artificial Intelligence, Renmin University of China, Beijing 100084, China*

E-mail: wuzhijing.joyce@gmail.com; yiqunliu@tsinghua.edu.cn; maojiaxin@ruc.edu.cn
{z-m, msp}@tsinghua.edu.cn

Received November 19, 2021; accepted June 29, 2022.

Abstract Document ranking is one of the most studied but challenging problems in information retrieval (IR). More and more studies have begun to address this problem from fine-grained document modeling. However, most of them focus on context-independent passage-level relevance signals and ignore the context information. In this paper, we investigate how information gain accumulates with passages and propose the context-aware Passage Cumulative Gain (PCG). The fine-grained PCG avoids the need to split documents into independent passages. We investigate PCG patterns at the document level (DPCG) and the query level (QPCG). Based on the patterns, we propose a BERT-based sequential model called Passage-level Cumulative Gain Model (PCGM) and show that PCGM can effectively predict PCG sequences. Finally, we apply PCGM to the document ranking task using two approaches. The first one is leveraging DPCG sequences to estimate the gain of an individual document. Experimental results on two public ad hoc retrieval datasets show that PCGM outperforms most existing ranking models. The second one considers the cross-document effects and leverages QPCG sequences to estimate the marginal relevance. Experimental results show that predicted results are highly consistent with users' preferences. We believe that this work contributes to improving ranking performance and providing more explainability for document ranking.

Keywords document ranking, neural network, passage cumulative gain

1 Introduction

Document ranking is one of the main challenges in Information Retrieval (IR) research. Given a query and a set of documents, document ranking aims to assign a relevance score for each query-document pair and then ranks them in descending order according to the scores. Many ranking models including unsupervised models (e.g., BM25 [1] and language models [2, 3]) and supervised models (e.g., learning to rank [4, 5] and deep ranking [6]) have been proposed to address this problem. These models usually capture relevance signals at the whole document level. However, when humans judge the relevance of documents (e.g., the assessment in TREC ad hoc task [7]), a document is typically considered relevant if any part of the document contains useful information. The relevant parts could be in any position of a long document according to the Scope Hypothesis [1]. Fig. 1 shows an example document retrieved for the query "changes to assessment criteria of

Regular Paper

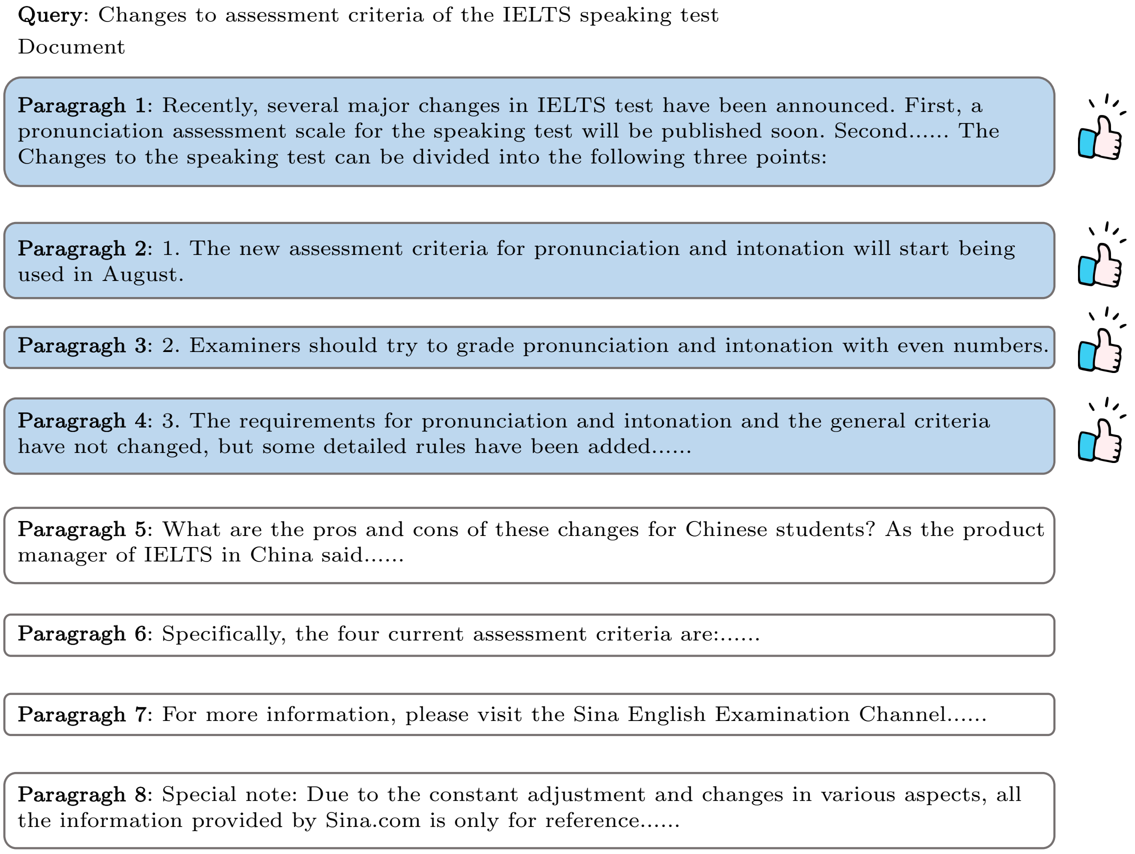
Special Section of Xia Peisu Young Scholars Forum 2021

A preliminary version of the paper was published in the Proceedings of WWW 2020.

This work was supported by the National Natural Science Foundation of China under Grant No. 61732008 and Tsinghua University Guoqiang Research Institute.

*Corresponding Author

©Institute of Computing Technology, Chinese Academy of Sciences 2022



Query: Changes to assessment criteria of the IELTS speaking test

Document

Paragragh 1: Recently, several major changes in IELTS test have been announced. First, a pronunciation assessment scale for the speaking test will be published soon. Second...... The Changes to the speaking test can be divided into the following three points:

Paragragh 2: 1. The new assessment criteria for pronunciation and intonation will start being used in August.

Paragragh 3: 2. Examiners should try to grade pronunciation and intonation with even numbers.

Paragragh 4: 3. The requirements for pronunciation and intonation and the general criteria have not changed, but some detailed rules have been added......

Paragragh 5: What are the pros and cons of these changes for Chinese students? As the product manager of IELTS in China said......

Paragragh 6: Specifically, the four current assessment criteria are:......

Paragragh 7: For more information, please visit the Sina English Examination Channel......

Paragragh 8: Special note: Due to the constant adjustment and changes in various aspects, all the information provided by Sina.com is only for reference......

Fig. 1. Example of high-gain documents to the query "changes to assessment criteria of IELTS speaking test". The document is relevant and meets the query's information needs, although only the first four paragraphs are relevant.

the IELTS speaking test". We can see that the document is composed of eight paragraphs, and only four of them are relevant to the query. The document was assessed to be a "high-gain document" which can totally satisfy users' information needs in our user study (see Section 3 for more details). However, it may be challenging to capture these local relevance signals if we only focus on whole-document-level features.

To help ranking models capture local relevance signals, several studies propose to estimate document relevance based on fine-grained passage-level relevance signals. In these studies [8–10], documents are split into passages based on textual discourse units (discourse passage), the subject of the content (semantic passage), or a fixed-length window (window passage). Local relevance signals are obtained from these passages and then combined to generate the document-level relevance scores. To better combine the local signals, different strategies are also proposed. Several researchers employ maximum, minimum, or weighted summation functions to generate document-level relevance and further compare the results to help understand the relationship between passage-level relevance and document-level relevance [11, 12]. Other methods leverage deep neural networks to learn the relationship [13, 14]. These efforts lead to improved ranking performances by introducing fine-grained relevance signals.

Since most of these studies estimate passage-level relevance signals independently without considering the context information, the estimated relevance may be inaccurate in many circumstances. For example, the second and the third paragraphs of the example document in Fig. 1 are too short for algorithms to estimate their relevance accurately. However, the last sentence of the first paragraph, "the changes to the speaking test can be divided into the following three points", indicates that the following content is relevant to the changes of the speaking test. Ignoring context information may lead to inaccurate estimation of passage-level relevance. A better solution should take the context information into consideration.

Unlike these existing studies, we try to estimate the passage cumulative gain (PCG) for relevance estimation rather than context-independent passage-level relevance signals. We assume that users will follow the sequential order while reading an article according to the findings in user reading behavior analysis [15]. Then, we focus on how the gain (i.e., useful information for the query) accumulates passage by passage when users read documents from top to bottom. With this framework, we avoid questions of how to split a document into independent passages and how to aggregate relevance scores of independent passages to get document-level relevance.

The cumulative gain (CG) has been used to evaluate ranking performance at both the query level [16,17] and the session level [18]. We study the cumulative gain at the passage level (PCG) to capture context-aware fine-grained relevance signals in this work. We first investigate the patterns of PCG within a document (i.e., DPCG). Considering that users may read multiple documents after submitting a query to the search engine, we also investigate the patterns of multi-document PCG (i.e., query-level passage cumulative gain, QPCG). It describes how the information gain accumulates across multiple documents in a whole query session. We further compare relevance, DPCG, QPCG, and usefulness labels to investigate the advantages of DPCG and QPCG. Then we try to model the PCG sequence with deep recurrent neural networks to predict the sequence automatically. Finally, we leverage the DPCG and QPCG sequences to improve the performance of document ranking. Besides ranking documents by declining estimated relevance scores, we also verify the potential of QPCG sequences on the marginal relevance [19] estimation task, which considers both the document's relevance scores and its similarity with previously selected documents. To summarize, we investigate the following research questions.

- *RQ*1. During users' information-seeking processes, how does their information gain accumulate passage by passage both within a document and across multiple documents?
- *RQ*2. Can we effectively predict the sequences of document-level and query-level passage cumulative gain based on the raw text of queries and documents?
- *RQ*3. Can the passage cumulative gain be applied to estimate document relevance and improve the performance of document ranking models?
- *RQ*4. Considering that the relevance of a document is influenced by the documents ranked before it, can the query-level passage cumulative gain be applied to estimate marginal relevance for document ranking?

To shed light on these research questions, we collect DPCG and QPCG annotations through lab studies on two datasets, an ad hoc retrieval dataset TianGong-PDR [12] and an exploratory search dataset SearchSuccess [20]. Based on the datasets, we firstly investigate the patterns of DPCG and QPCG to answer RQ1. Then we define the PCG prediction as a sequence prediction task and propose a novel Passage Cumulative Gain Model (PCGM). It employs BERT [21] to learn initial representations for query-passage pairs and incorporates the observed patterns into an LSTM [22] to predict PCG sequences effectively. Finally, we leverage this model to estimate a relevance score for a document and test the ranking performance of PCGM. We further test the effectiveness of PCGM on the marginal relevance estimation task. To summarize, the main contributions are as follows.

1) We collect the fine-grained document-level and query-level passage cumulative gain annotations[1] for an ad hoc retrieval dataset TianGong-PDR [12] and an exploratory search dataset SearchSuccess [20].

2) We provide a thorough analysis of the patterns by which the passage information gain accumulates both within a document and across multiple documents. It helps us better understand how information gain is perceived when users seek useful information.

3) Through the comparison of relevance, DPCG, and QPCG labels, we show the advantages of passage cumulative gain compared with relevance.

4) We show that the sequence of DPCG and QPCG can be effectively predicted by incorporating the observed patterns into a BERT-based deep neural network.

5) We employ the DPCG sequences into document ranking models and show their effectiveness in improving ranking performance. We leverage the QPCG sequences to estimate the marginal relevance. Experimental results show that predicted results are highly consistent with users' preferences.

Note that an early version [23] of this paper was accepted by the Web Conference 2020. We make substantial progress compared with the conference version, especially in extending the passage cumulative gain from the document level to the query level. The collected annotations and analysis of query-level passage cumulative gain (QPCG) in the first three contributions help us understand how the information gain accumulates in users' information-seeking process. Experiments based on the QPCG annotations in the last two contributions

[1] http://www.thuir.cn/data-pdr/, May 2022.

show the effectiveness of this fine-grained gain on gain prediction and marginal relevance estimation tasks.

2 Related Work

2.1 Passage-Level Relevance

Callan[8] proposed that with the increase of documents' length, it is natural to consider the fine-grained relevance (e.g., passage-level relevance) in ranking tasks. Plenty of studies[8, 12, 24] have investigated the fine-grained passage-level relevance signals. It is shown to be useful in understanding the relevance judgment process and effective in improving the performance of document ranking[12, 14, 25]. There are several methods to split a document into passages in previous passage retrieval research. Callan[8] categorized most of them into three types: discourse, semantic, and window passages. Discourse passages are obtained by splitting documents based on textual discourse units such as sentences, paragraphs, and sections[12]. Semantic passages are derived from documents based on the subject or content of the text[8]. Window passages do not take the logical structure or semantic information of documents into consideration but consist of a fixed number of words[10]. Hearst and Plaunt[26] split documents into blocks of several sentences and further grouped these blocks into passages based on the cosine similarity of neighboring blocks. This splitting method considers the subtopic structure of the document.

After splitting documents into passages, relevance signals are obtained from these passages, which is called passage-level relevance. Users can obtain useful information from relevant passages faster than from long documents[27]. Several studies have utilized the passage-level relevance to generate document-level relevance scores and improved the performance of document ranking[11, 12, 24]. For example, the highest passage-level relevance score of all passages was taken as the document-level relevance score by Liu and Croft[24]. Neural models[13] are also utilized to model both the passage-level and document-level matching signals and are shown effectively on the document ranking task. Recently, Wu *et al.*[12] collected a four-grade relevance annotation for each passage within a document. They employed maximum, minimum, or weighted summation functions on these passage-level relevance annotations to estimate the document-level relevance. It helps better understand the relationship between passage-level relevance and document-level relevance.

In this work, documents in our dataset are well-organized according to their semantic structure. Therefore, we split documents into passages according to the discourse unit (i.e., paragraph). Different from the studies which calculate relevance scores for each passage separately, we consider the context and investigate a context-aware passage cumulative gain (PCG).

2.2 Document Ranking Models

For document ranking task, a large number of methods have been proposed, including classical probability models (e.g., BM25[1]), feature-based learning-to-rank models[4, 5, 28], and neural ranking models[6]. Neural ranking models have been shown effective at learning ranking scores automatically from the raw text of queries and documents. We mainly review neural ranking models in recent years.

Existing neural ranking models can be categorized into representation-based models and interaction-based models. The first one aims to build good representations of queries and documents. The second one aims to build local interactions between the query and the document, and then aggregate each interaction to learn a relevance score. For example, Hu *et al.*[29] proposed CNN-based ARC-I and ARC-II for matching two sentences. The former gets the representation of the query and the document, and then compares the two representations to predict the ranking score. The latter conducts an interaction between the matrixes of the query and the document, and then predicts the ranking score. Deep Relevance Matching Model (DRMM)[30] uses matching histogram mapping as the input and combines a feed forward matching network and a term gating network to model query term importance. MatchPyramid[31] considers text matching as a problem of image recognition and addresses the problem by convolution approaches. Position-Aware Convolutional Recurrent Relevance Matching (PACRR)[28] uses convolutional layers to capture term matching and positional information based on the query-document interactions. Kernel-based Neural Ranking Model (KNRM)[32] uses a kernel-pooling technique to extract multi-level soft match features between the query and the document.

Recently, fine-grained passage-level matching signals have been leveraged to address the document ranking task. Pang *et al.*[14] considered the human relevance judgment process and proposed DeepRank. It first detects relevant locations, then determines local relevance, and finally aggregates local relevance to get the document-level ranking score. Fan *et al.*[13] followed the same idea and proposed Hierarchical Neural Matching Model (HiNT). Li *et al.*[33] proposed Reading Inspired Model (RIM) based on users' reading behavior patterns. It first captures the sentence-level relevance signals and then models the document-level relevance according to reading heuristics from the human. Pre-trained neural language models have also been used in document ranking. BERT[21] is an effective pre-trained language model trained on a large-scale, open-domain corpus and can be used to obtain the representation of texts. Based on BERT, Dai and Callan[11] took the concatenation of the query and passages as input, and then used the maximum, first, and summation of matching scores of query-passage pairs as document-level ranking scores. The experimental results show the effectiveness of BERT on the document ranking task.

Since our focus in this work is to verify whether the structure of our proposed passage cumulative gain model (PCGM) is effective in PCG sequence predicting and document ranking, we directly use the most popular original BERT model to obtain the initial passage embeddings according to Dai and Callan[11], and then update the embeddings using RNN.

2.3 Document Dependence and Marginal Relevance

Most of the existing ranking models follow the classical probability ranking principle (PRP)[34]. It assumes the relevance of a document is independent of the relevance of other retrieved documents. However, this assumption is not always reasonable in real situations. Goffman[35] first recognized that the relevance score of a document is affected by the content of documents ranked before it. After users read one document in interactive information retrieval (IIR), they may choose to read another document or leave the system. Users' actions are affected by the content they read before. Therefore, Fuhr[36] extended the classical PRP considering the effort and the probability of a user selecting the following document.

Users' relevance judgment on a document is influenced by the documents they previously examined. Many researchers have paid attention to this problem and tried to model the cross-document effects to better estimate document-level relevance. Zuccon *et al.*[37] proposed the Quantum Probability Ranking Principle (QPRP) to model the dependent relevance and overcome the independent relevance assumption of the classical PRP. Carbonell and Goldstein[19] proposed the Maximal Marginal Relevance (MMR) method. It iteratively selects the document with a high relevance score to the query and low similarity score to the selected documents into the ranking list. Most of the existing implicit diversification methods are proposed based on the idea of MMR method. For example, Chen and Karger[38] assumed a document is irrelevant once it has been included in the ranking list to maximize the implicit diversity. The marginal relevance estimation task is similar to the sentence selection task in TREC Novelty Track[2] which aims to select relevant and novel sentences given a TREC topic and a list of documents. However, the track task directly ranks sentences based on their relevance and novelty, while most of the marginal relevance estimation related studies rank documents.

In this work, we use a BERT-based sequential network to model the clicked document sequence in a query session. It incorporates the information of previous passages, including the passages in previous clicked documents, to the current passage when estimating the current PCG score. Therefore, our model implicitly captures the cross-document effects. The marginal relevance score of each document can be obtained from the estimated PCG scores.

3 Passage Cumulative Gain

In this section, we first describe the definition of passage cumulative gain, at both the document level (i.e., DPCG) and the query level (i.e., QPCG). Then we collect DPCG and QPCG annotations for two public datasets: TianGong-PDR and SearchSuccess.

3.1 Definition

We start by defining document-level passage cumulative gain (DPCG, i.e., the PCG proposed by Wu *et al.*[23]). Given a query and a document, considering that users usually follow the sequential order while reading an article[15], we assume that the gain (i.e., useful information for the query) obtained by the users accumulates passage by passage when users read a document from top to bottom. Formally, given a query q and

[2] Ian Soboroff. Overview of the Trec 2004 novelty track, 2005. https://www.nist.gov/publications/overview-trec-2004-novelty-track, May 2022.

a document $d = \{p_1, p_2, ..., p_n\}$, where n is the number of passages in the document and p_i is the i-th passage, the DPCG labels of d can be described as a sequence $DPCG_d = \{g_1, g_2, ..., g_n\}$, where g_i $(1 \leqslant i \leqslant n)$ denotes the degree of gain that the user obtains from the first i passages in d. Therefore, g_n is the degree of gain that the user obtains from the whole document d. We also use g_n to denote the document-level cumulative gain (DLCG) of d.

Then we extend the document-level passage cumulative gain to the query-level one. We define the query-level passage cumulative gain (QPCG), which is based on the content the user reads in a query session (i.e., the process of searching a specific query). After users submit a query in a query session, search engine result pages containing a sequence of documents are shown. They click into some of the documents and read the landing pages carefully to find useful information for the query (i.e., gain). We assume that the users' gain accumulates passage by passage when they read a sequence of clicked documents in the query session. Formally, given a query q and a sequence of clicked documents $D = \{d_1, d_2, ..., d_m\}$, where m is the number of clicked documents in the query and d_i is the i-th clicked document, the QPCG labels of q can be described as a sequence $QPCG_q = \{g^1_1, g^1_2, ..., g^1_{l_1}, ..., g^m_1, g^m_2, ..., g^m_{l_m}\}$, where l_i is the number of passages in the document d_i, and g^i_j $(1 \leqslant i \leqslant m, 1 \leqslant j \leqslant l_i)$ denotes the degree of gain that the user obtains from the first $i-1$ clicked documents and the first j passages in d_i. Therefore, $g^m_{l_m}$ is the degree of gain that the user obtains from the whole query session q. We use $g^m_{l_m}$ to denote the query-level cumulative gain (QLCG) of q.

In this work, we take one paragraph as one passage following Wu *et al.* [12] and use a four-grade DPCG and QPCG judgment scale.

- *DPCG*: *Document-Level Passage Cumulative Gain.* It describes the process of gain cumulating passage by passage within a document. $DPCG = \{g_1, g_2, ..., g_n\}$, where n is the number of passages in the document, and g_i $(1 \leqslant i \leqslant n)$ denotes the degree of gain that the user obtains from the first i passages.
- *DLCG*: *Document-Level Cumulative Gain.* It denotes the degree of gain of the whole document. $DLCG = DPCG(n) = g_n$.
- *QPCG*: *Query-Level Passage Cumulative Gain.* $QPCG = \{g^1_1, g^1_2, ..., g^1_{l_1}, ..., g^m_1, g^m_2, ..., g^m_{l_m}\}$, where m is the number of clicked documents in the query session, l_i is the number of passages in the i-th clicked document, and g^i_j $(1 \leqslant i \leqslant m, 1 \leqslant j \leqslant l_i)$ denotes the degree of gain that the user obtains from the first $i-1$ clicked documents and the first j passages in the i-th document.
- *QLCG*: *Query-Level Cumulative Gain.* It denotes the degree of gain of the whole query session. $QLCG = QPCG(M) = g^m_{l_m}$, where $M = \sum_{i=1}^{m} l_i$.

3.2 Data Collection

We conduct multiple lab studies to collect DPCG and QPCG annotations. We first collect DPCG annotations for an ad hoc retrieval dataset TianGong-PDR, which is used to train our ranking models. We also annotate both DPCG and QPCG for an exploratory search dataset SearchSuccess, which contains users' query sessions and the content of clicked documents. In this subsection, we will introduce the lab studies in detail.

3.2.1 DPCG Annotation for TianGong-PDR Dataset

We first collect the DPCG annotations for a recent and public ad hoc retrieval dataset, TianGong-PDR[③] [12]. TianGong-PDR consists of 70 general interest queries from search logs of the Sogou search engine, 70 manually generated search intent descriptions, and 1 050 documents from a Chinese news corpus THUCNews[④]. There are 15 documents for each query and 564 words per document on average. These news documents are well-organized and of high quality. Some query examples and their corresponding information needs are shown in Table 1, which we translate from Chinese into English. We conduct a lab-based study to collect the DPCG annotations for this dataset.

Annotation Instructions. We use a four-grade DPCG annotation in the study. It reflects the degree of gain. The instructions for the four-grade DPCG annotation are as follows: (0) no gain: there is no useful information for the information needs behind the query in the content you have read; (1) low gain: based on the content you have read, the information needs can be slightly satisfied; (2) moderate gain: based on the content you have read, the information needs can be fairly satisfied; (3) high gain: based on the content you have read, the information needs can be totally satisfied.

Procedure. We first ask participants to read the annotation instructions carefully and then guide them to

[③] http://www.thuir.cn/data-pdr/, May 2022.

[④] http://thuctc.thunlp.org/, May 2022.

Table 1. Search Query Examples (Translated from Chinese)

Dataset	Query	Information Need
TianGong-PDR [12]	Tips for kitchen decoration	You are preparing to decorate the house and want to know some tips for kitchen decoration
	Reasons for the rise of oil price	Recently the oil price has risen. You want to know the possible reasons behind it
SearchSuccess [20]	Cancer treatment	You want to investigate methods for cancer treatment and their advantages and disadvantage
	Characteristics of particulate pollution	You want to investigate the characteristics of particulate pollution in China, including its compositions and geographical characteristics

finish two training tasks to get familiar with the experimental system quickly. Each annotation task involves a query-document pair. At the beginning of each task, the system shows the search query text, search intent description, and the first passage in the document to participants. The participants need to annotate the first passage's DPCG degree after reading it. Next, both the first and the second passages are presented. Participants need to give the DPCG judgment for the first two passages together. The same step repeats until all the passages in the document have been shown. Finally, we obtain a sequence of DPCG annotations for a query-document pair from a participant.

Participants. There are 70 queries and 15 documents for each query in the dataset. Documents belonging to the same query are randomly allocated into one of 15 groups without repetition. Therefore, there are 70 query-document pairs in each group. Each participant needs to annotate one group. The tasks in a group are shown in random to avoid ordering effects. Three different participants annotate each group. By posting posters around the campus and on social networks, we recruit 45 participants in this study, 19 males and 26 females. They are all undergraduate and graduate students from a university. Their ages range from 18 to 29, and their majors vary from natural science and engineering to humanities and sociology. All of them have basic Chinese reading skills and daily search experience using Chinese search engines. It takes participants around 1.5 hours to finish 70 tasks. Each is paid around $15 as compensation.

3.2.2 *QPCG and DPCG Annotation for SearchSuccess Dataset*

We then collect the QPCG annotations for an exploratory search dataset, SearchSuccess Dataset⑤ [20]. It contains 166 search sessions collected through a laboratory user study. The search tasks are designed from environment, medicine, and politics domains. Since this dataset is used for search evaluation in exploratory search, the information needs behind these search tasks can hardly be satisfied by only one document. A search session usually contains multiple query sessions and multiple clicked documents. There are 652 query sessions in this dataset. Table 1 shows two query examples and their information needs. The queries are submitted by users, and documents are from web pages. The user study collects user interactions (e.g., clicks) and explicit feedback (e.g., query satisfaction, usefulness, and relevance of clicked documents). We use the search queries and clicked documents for annotation.

Instructions and Procedure. We use a four-grade QPCG annotation, which is the same as the four-grade DPCG annotation introduced in Subsection 3.2.1. Different from the DPCG annotation task that involves only one query-document pair, each QPCG annotation task involves one search query and a sequence of users' clicked documents under this query. At the beginning of each task, the participant needs to read the query, the search intent description, and the first passage of the first clicked document. Then he/she needs to annotate the four-grade QPCG degree for the first passage after reading it. Next, both the first passage and the second passage are shown, and the participant gives the QPCG degree for the first two passages. When the participant has read all the passages within the first document, the first passage of the second clicked document is shown. The same step repeats until all the clicked documents have been shown.

Participants. We filter out the query sessions in which users click on less than two documents, and finally, 234 query sessions are kept. There are 750 clicked documents in these 234 query sessions. We randomly divide these query sessions into nine groups and recruit 27 participants for the QPCG annotation. Three participants annotate each query group. For further com-

⑤http://www.thuir.cn/data-SearchSuccess/, May 2022.

paring DPCG and QPCG, we also collect the DPCG annotations for all of the 750 query-document pairs. These pairs are randomly divided into 10 groups, and we recruit another 30 participants for the DPCG annotation. Therefore, for a query session q and a sequence of clicked documents $D = \{d_1, d_2, ..., d_n\}$, we obtain three DPCG sequences for each document d_i in D and three QPCG sequences for query session q. It takes participants about 1.5 hours to finish 26 QPCG annotation tasks or 75 DPCG annotation tasks. Each is paid about \$15 as compensation.

3.3 Collected Dataset

For the TianGong-PDR dataset, we obtain three DPCG sequences for each query-document pair. For the SearchSuccess dataset, we obtain three QPCG sequences for each query session and three DPCG sequences for each query-document pair. We use Hayes and Krippendorff's α [39] for ordinal data to measure the inter-person agreement of annotations. The values of Hayes and Krippendorff's α for TianGong-PDR DPCG, SearchSuccess DPCG, and SearchSuccess QPCG annotations are 0.625, 0.667, and 0.624, respectively. It indicates a moderate agreement level. We use the median of three labels from three participants as the final label.

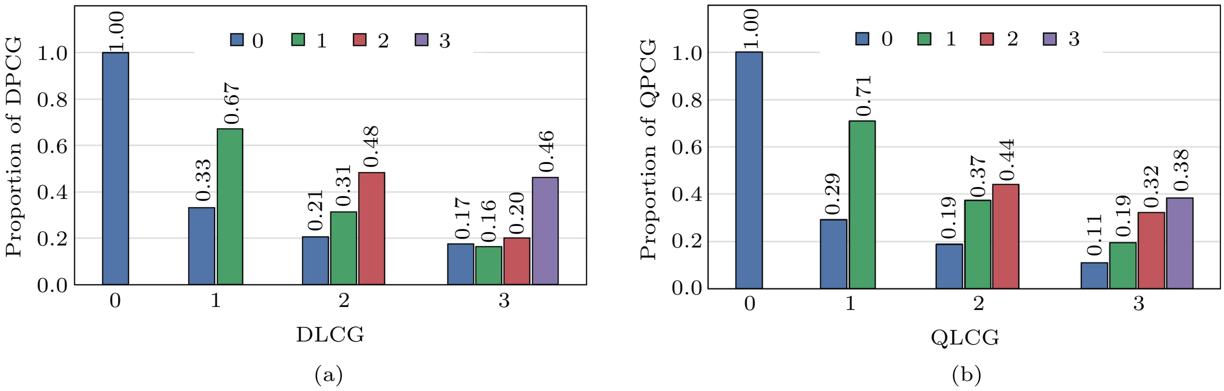
Some details about the datasets are shown in Table 2. In the TianGong-PDR dataset, about 21.5% of documents can fully satisfy the information needs ($DLCG = 3$). The high-gain documents contain more passages and more words than other documents on average. In the SearchSuccess dataset, the clicked documents can fully satisfy the information needs in about 31.6% queries. There are 9.7 passages in a document on average. It shows that the documents from News websites[6] in the TianGong-PDR dataset are longer than those from the general webpages in the SearchSuccess dataset.

In Fig. 2, we plot the distributions of (0) no-gain, (1) low-gain, (2) moderate-gain, and (3) high-gain DPCG/QPCG annotations in documents/queries at different DLCG/QLCG levels. In no-gain documents (i.e., $DLCG = 0$) and no-gain queries (i.e., $QLCG = 0$), all DPCG and QPCG annotations are zero. The DPCG/QPCG degree with the largest proportion in the other three kinds of documents/queries matches their DLCG/QLCG degree. Within the high-gain documents (i.e., $DLCG = 3$) and high-gain queries

Table 2. Distributions of DLCG and DPCG in the TianGong-PDR Dataset [23], the QLCG and QPCG in the SearchSuccess Dataset

Value	DLCG			DPCG	QLCG			QPCG
	Proportion	Avg.#P	Avg.#W	Proportion	Proportion	Avg.#D	Avg.#P	Proportion
0	0.390	10.8	536	0.527	0.038	2.67	17.2	0.181
1	0.208	10.5	548	0.230	0.209	2.53	19.5	0.326
2	0.187	10.9	585	0.136	0.436	3.09	27.3	0.316
3	0.215	11.8	605	0.107	0.316	3.88	45.6	0.177
All	1.000	11.0	562	1.000	1.000	3.21	31.1	1.000

Note: Avg.#P and Avg.#W mean the average number of passages and words within documents, respectively. Avg.#D means the average number of documents in a query session.



(a)

(b)

Fig.2. Joint distributions of (a) document-level cumulative gain (DLCG) and document-level passage cumulative gain (DPCG) [23], and (b) query-level cumulative gain (QLCG) and query-level passage cumulative gain (QPCG). Avg.#D means the average number of documents in a query session.

[6] https://rss.sina.com.cn/, May 2022.

(i.e., $QLCG = 3$), there are still 17%/11% no-gain DPCG/QPCG annotations on average. It indicates that users did not find useful information at the early stage of reading the document.

4 Patterns of DPCG and QPCG

4.1 Patterns of Document-Level Passage Cumulative Gain

We analyze how the document-level passage information gain accumulates when users are seeking useful information in a document based on the collected data. We first look into how the DPCG annotations change after users read one more passage. The transition probabilities $P(g_i = x \mid g_{i-1} = y)$, where g_i is the DPCG annotation for the first $i-1$ passages ($2 \leqslant i \leqslant n$, n is the number of passages within the document), are shown in Fig.3. For example, "0.905" indicates that when g_{i-1} is zero, the probability for $g_i = 0$ is 0.905.

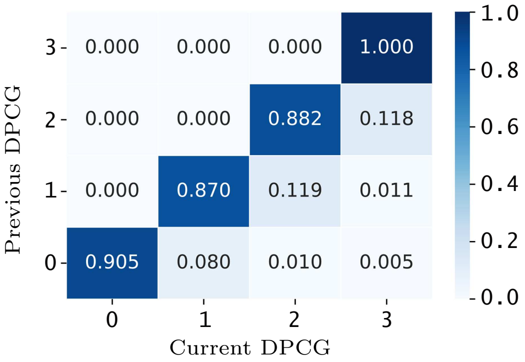
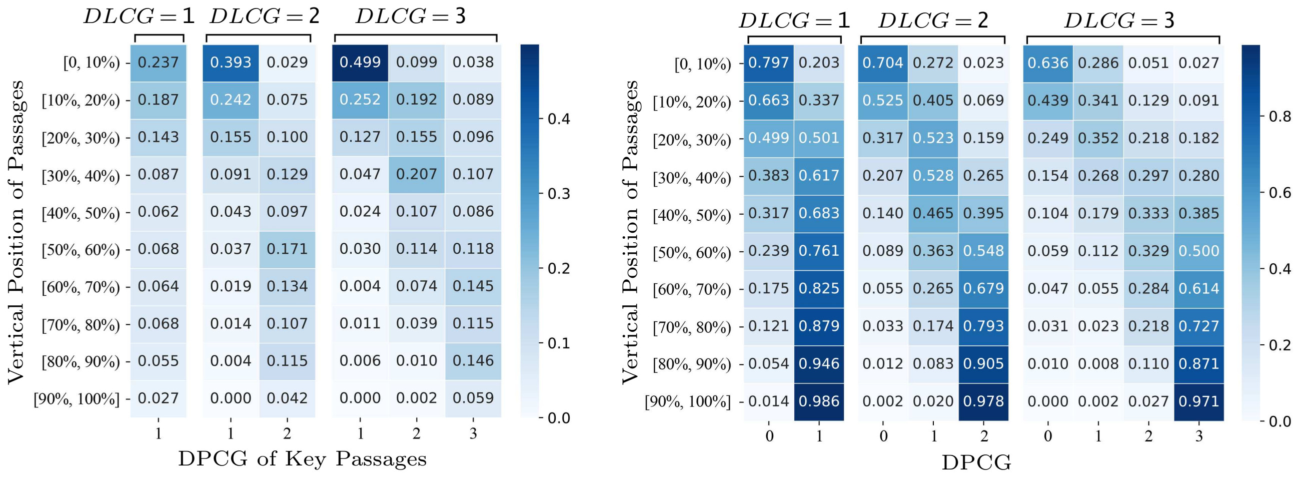


Fig.3. Transition probabilities of document-level passage cumulative gain (DPCG) [23].

We find that the probabilities that g_{i-1} is greater than g_i are all zero. It shows that the PCG sequence of a document in our collected data is always non-decreasing. In other words, the useful information captured by users accumulates as they read more and more passages. It may be because the documents in the TianGong-PDR dataset are news articles. They are well written, structured, and remain trustworthy throughout. For example, many news documents start with an introductory sentence that already summarizes the story (e.g., the high-gain document example in Fig.1; for query "changes to assessment criteria of IELTS speaking test", there is a summarization sentence in the first paragraph: "The changes to the speaking test can be divided into the following three points."). They often follow the "inverted pyramid" writing structure. There is no document where it seems promising at the start, but later on, the reader discovers something strange, loses all trust in the content, and reduces the DPCG grades. Probabilities on the diagonal line are the largest in all four columns, followed by the probabilities for $g_i - g_{i-1} = 1$, while probabilities for $g_i - g_{i-1} > 1$ are rather small. Therefore, we can summarize that when DPCG increases from g_{i-1} to g_i, the increment is most likely to be 1.

We define the passage where the DPCG annotation is different from the previous one as the "key passage". Since the DPCG sequence is non-decreasing, the i-th passage is a key passage only if g_i is greater than g_{i-1}. The values of DPCG annotations increase at key passages. There are three kinds of key passages: low-gain key passages ($DPCG = 1$), moderate-gain key passages ($DPCG = 2$), and high-gain key passages ($DPCG = 3$). We split passages within a document into 10 parts according to their vertical positions and analyze the distribution of vertical positions of key passages. A small value of vertical positions indicates that the passage is at the beginning of the document. Fig.4(a) shows distributions of key passages in low-gain, moderate-gain, and high-gain documents. We do not plot the distribution in no-gain documents because there is no key passage in no-gain documents. The values in the figure are the proportions of key passages. For example, "0.237" in the first row means that in low-gain documents, 23.7% of low-gain key passages are located in the top 10% part of documents. Similarly, "0.099" in the first row means that 9.9% of moderate-gain key passages are located in the top 10% part of documents in high-gain documents.

We find that the proportions of low-gain key passages tend to decay as the vertical position increases. It indicates that users usually obtain some useful information at the beginning of documents, except no-gain documents. The higher the document's cumulative gain, the higher the vertical position of low-gain key passages and moderate-gain passages. When looking into the vertical position where the value of DPCG becomes the same as the value of DLCG, we find that the most likely position is lower as the DLCG increases. Most of the low-gain key passages are in the 0%–30% part of low-gain documents, while most of the moderate/high-gain key passages are in the 30%–90% part of moderate/high-gain documents. There are still 14.6% of high-gain key passages in the 80%–90% part of high-gain documents. It shows that useful information can locate in any position of a document.



(a) (b)

Fig.4. Distributions of (a) key passages and (b) DPCG annotations at different vertical positions in low-gain ($DLCG = 1$), moderate-gain ($DLCG = 2$), and high-gain ($DLCG = 3$) documents[23].

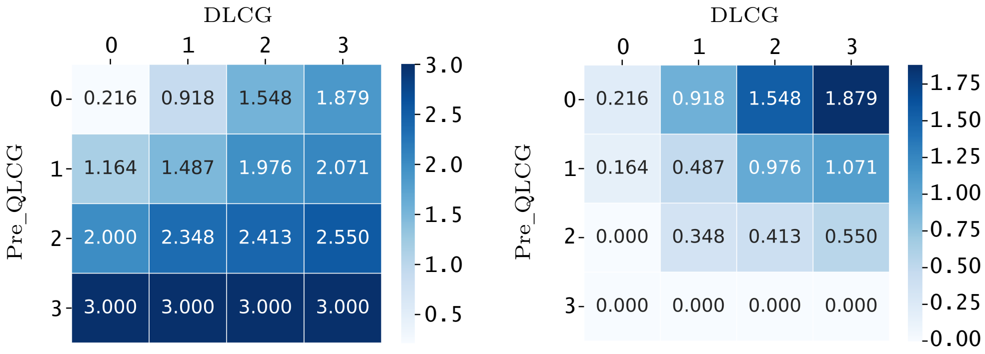
Fig. 4(b) shows distributions of DPCG at different vertical positions. "0.797" means that in 0%–10% part of low-gain documents, the probability that DPCG equals zero is 0.797. We observe that in low-gain documents, the probability that DPCG equals DLCG reaches 0.5 at the position of 20%–30%, while in moderate-gain and high-gain documents, the probabilities reach 0.5 at the position of 50%–60%, which is lower than that in low-gain documents. It indicates that as DLCG increases, more passages need to be read to judge an accurate DLCG.

Summary. Answering RQ1, we find that the DPCG sequence is non-decreasing (i.e., the current DPCG is equal to or greater than the previous one). The value of the i-th DPCG in the DPCG sequence of a document is determined by the content of the top i passages and is highly related to the previous DPCG. The higher the DLCG is, the lower the position where DPCG reaches DLCG. Users need to read more passages to judge an accurate DLCG as DLCG increases.

4.2 Patterns of Query-Level Passage Cumulative Gain

Different from the DPCG sequence that describes how the gain cumulates within a document, the QPCG sequence shows how the gain cumulates in the whole query session. A user sometimes reads more than one document in a query session. The increment of QPCG when reading the i-th clicked document is affected by the DLCG of d_i, and maybe also affected by the documents that users have read previously (i.e., d_0, ..., d_{i-1}). We study the relationship among the pre_QPCG, the DLCG, and the post_QPCG of a document. The pre_QPCG refers to the QPCG level before users read a document in a query session. For the first document that users read, its pre_QPCG is equal to zero. The post_QPCG refers to the QPCG level after users read a document in a query session. For the last document that users read, its post_QPCG is equal to the QLCG of this query session. The gain refers to the increment of QPCG after users read a document in a query session (i.e., $gain = post\_QPCG - pre\_QPCG$).

The average values of post_QPCG are shown in Fig. 5. For example, "1.487" in Fig. 5(a) means that when the pre_QPCG equals 1, the post_QPCG reaches 1.487 on average after users read a low-gain document. "0.487" in Fig. 5(b) means that when the pre_QPCG equals 1, the QPCG increases by 0.487 on average after users read a low-gain document. We find the followings. 1) When the pre_QPCG is 0/1, the gain of 20%/16% no-gain documents is not zero in our dataset. It may be because the four-grade QPCG is coarse-grained. Annotations are limited by the four-level scale. For some of the documents which contain a little useful information, participants in the DPCG annotation task give no-gain judgments. In contrast, those in the QPCG annotation task give low-gain judgments. If we use a more fine-grained annotation scale, such as the 100-level scale[40], the annotation will be more flexible. It is one of the aspects that we can improve in the future. 2) In most cases, the relationships among the pre_QPCG, the DLCG, and the post_QPCG satisfy $post\_QPCG \leqslant \min(upper\_bound, pre\_QPCG + DLCG)$, where *upper_bound* is the upper bound of the

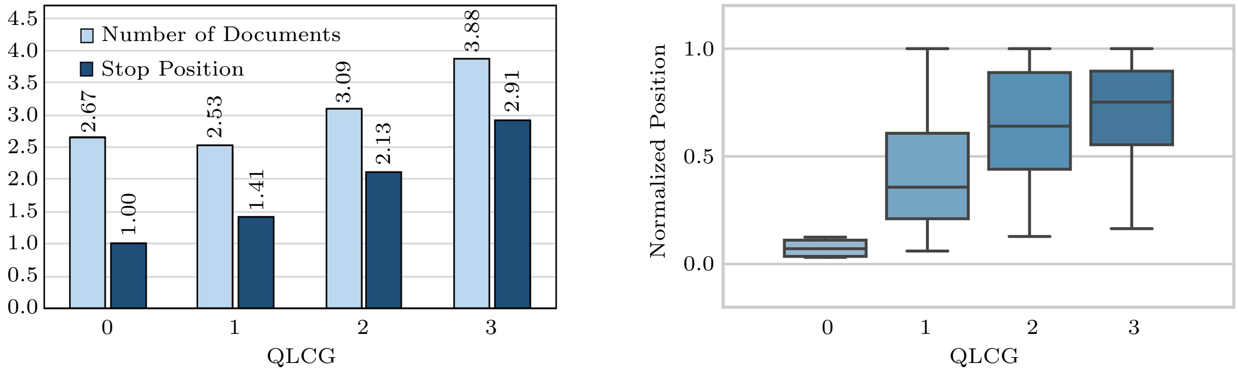


(a) (b)

Fig.5. Average values of (a) post_QPCG and (b) gain with different pre_QPCG values in no-gain ($DLCG = 0$), low-gain ($DLCG = 1$), moderate-gain ($DLCG = 2$), and high-gain ($DLCG = 3$) documents.

QPCG label. In this work, the upper_bound is 3. For example, when the pre_QPCG is 2 and DLCG is 1, the post_QPCG reaches 2.348 on average, which is less than the sum of pre_QPCG and DLCG. Some low-gain documents do not bring in gain increment for the query session. It may be because the useful information in these documents is duplicated with the content that users have read. It is worth noting that in some cases, post_QPCG is less than DLCG. Some documents are annotated as high-gain documents (i.e., they can totally satisfy the information needs) in the DPCG annotation task. However, in the QPCG annotation task, the average of the post_QPCG of these documents does not reach 3. It may be because users have formative expectations [41] when doing the annotation task. In a QPCG annotation task, more than one document is shown to users. They will expect more useful information than the situation when only one document is provided. Therefore, it is easier to get fully satisfied with the DPCG annotation task. 3) For documents with the same DLCG, the higher the pre_QPCG, the lower the gain. For example, when the pre_QPCG is 0, 1, 2, or 3, the gain of low-gain documents is 0.918, 0.487, 0.348, and 0, respectively. It indicates that higher pre_QPCG makes the gain more difficult to increase.

In Subsection 4.1, we find that the higher the DLCG is, the lower the position where DPCG reaches DLCG. The values of DPCG stop increasing from this position. We denote this position as the stop position. We use the number of documents and the number of passages to measure the stop position of QPCG. Fig.6(a) shows the average number of documents and stop position when the QLCG is 0, 1, 2, and 3, respectively. For example, when the QLCG is equal to 3, there are 3.88 clicked documents in a query session on average, while the QPCG reaches 3 after users read 2.91 documents on average. It indicates that sometimes users choose to continue clicking and reading more documents even if



(a) (b)

Fig.6. (a) Average position where the QPCG stops increasing (i.e., how many documents make the QPCG reach the QLCG) and the average number of documents. (b) Distribution of normalized positions where the QPCG stops increasing (i.e., what percentage of passages makes the QPCG reach the QLCG).

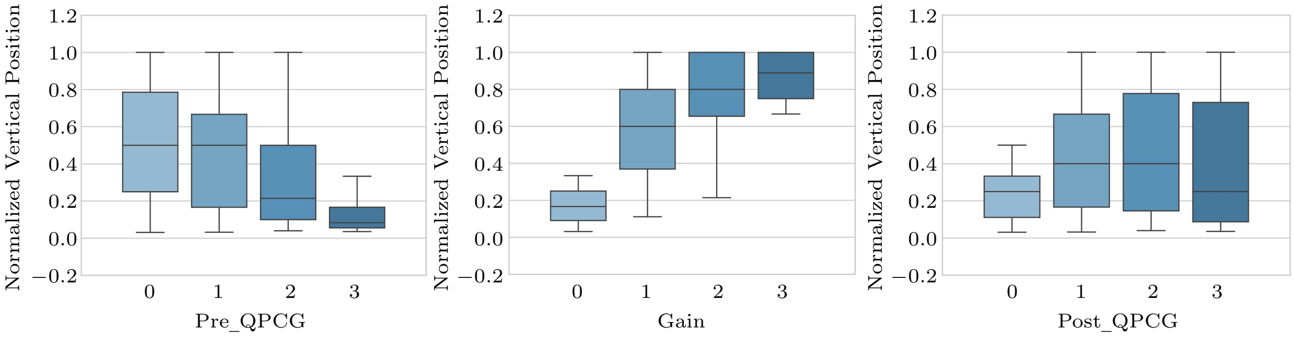
the information needs have been satisfied. In the query sessions whose QLCG is equal to 2, the QPCG reaches 2 after users read 2.13 documents on average. Then they continue reading about one document and find there is no more useful information. They choose to end this query session. It is similar to the query sessions whose QLCG is 1. We also plot the distribution of normalized stop positions, which are measured by the numbers of passages in Fig.6(b). Similar to the findings in Subsection 4.1, we find that the higher the QLCG is, the more the passages the users need to read to reach the QLCG.

Then we focus on where the QPCG stops increasing within each document in the query session. The stop position of DPCG is related to the DLCG of the document. The higher the DLCG, the lower the stop position. In a query session, the stop position of QPCG within a document may also be related to pre_QPCG, gain, and post_QPCG. We plot the distribution of the normalized position where the QPCG stops increasing within a document in Fig. 7. The position is calculated by the percentage of passages. We find that as pre_QPCG increases, the stop position becomes lower. As the gain of the document increases, the stop position becomes higher. However, there is no apparent relationship between the stop position and post_QPCG. We further calculate the Pearson correlation coefficient (PCC) between the stop position and these three factors. The PCC values show that pre_QPCG correlates with the stop position negatively, and the gain correlates with the stop position positively.

Summary. By analyzing the QPCG sequence, we find that in most cases, the increment of QPCG after users read a document is less than its DLCG label. It may be because 1) the useful information in this document is duplicated with the content users have read, 2) there exists the formative expectation that users expect more useful information in a query session than the situation when only one document is provided. We also find that the phenomenon of QPCG stopping increasing before the query session ends is common. When the QPCG stops increasing, users read about one document on average and then end the query session. Concerning the stop increasing position of QPCG within a document, it highly correlates with the pre_QPCG negatively and correlates with the gain positively.

4.3 Comparison of Relevance, DPCG, and QPCG

Besides the DPCG and QPCG annotations, the SearchSuccess dataset includes each document's relevance and usefulness labels in query sessions. We further make a comparison of these different measures to give an in-depth analysis of cumulative gain. We regard usefulness as the golden measure to judge the goodness of a document because the usefulness labels are given by the user when he/she performs the query session. Usefulness labels accurately reflect how useful the document is for addressing the information needs. We calculate the PCC between usefulness and relevance/DLCG/gain (i.e., *post_QPCG−pre_QPCG*). The PCC values of relevance, DLCG, and gain are 0.265, 0.341, 0.364, respectively. The PCC of usefulness and relevance is the lowest among the three measures. It shows that the gap between relevance and usefulness is the biggest. DLCG is more similar to usefulness compared with relevance. PCC of gain is slightly higher than that of DLCG. It indicates that DLCG performs better than relevance on usefulness estimation. Because usefulness is affected by the documents users



(b) (b) (c)

Fig. 7. Distribution of the normalized vertical position where the QPCG stops increasing within a document with different (a) pre_QPCG, (b) gain, and (c) post_QPCG. The normalized vertical position is calculated by the percentage of passages in the document.

have read before, the gain, which also considers this factor, achieves the best performance.

We plot the joint distribution of relevance and DLCG, the joint distribution of DLCG and gain in Fig.8, to analyze their differences. For example, "0.103" in the first row of Fig.8(a) means that there are 10.3% documents whose relevance and DLCG labels are both zero. We find the followings. 1) For only 40.4% documents, their relevance labels are equal to DLCG labels. These two labels are not consistent in more than half documents. 2) There are 14.4% documents whose relevance labels are zero, while DLCG labels are more than zero. It may be because though the document's primary topic is irrelevant to the information needs, a small part of the document contains useful information, and even the direct answer when the information need is a factoid question. Through fine-grained DPCG annotation, we can locate the useful information and provide these useful passages to users to avoid reading a long irrelevant document. 3) There are 31.9% documents whose relevance labels are greater than DLCG labels. Even if a document is highly relevant to information needs, it may not contain useful information. From the joint distribution of DLCG and gain in Fig.8(b), we find that: 1) for 46.4% documents, their DLCG labels are equal to the increment of QPCG labels; 2) there are 47.6% documents whose gain values in a query session are less than their DLCG labels. It is consistent with our findings in Subsection 4.2.

Summary. By comparing the relevance, DLCG, and gain, we find that DLCG performs better on usefulness estimation than relevance. Because the gain is affected by the documents that users have read, it achieves slightly better performance than DLCG.

5 Passage Cumulative Gain Model

We have shown in Section 4 that the DPCG and QPCG sequences are non-decreasing, and the values in sequences are related to the previous value. In this section, we propose a Passage Cumulative Gain Model (PCGM). It leverages context-aware sequence information to address the PCG (i.e., DPCG and QPCG) sequence prediction task. The framework of PCGM is illustrated in Fig. 9. It consists of three components: passage encoder, sequential encoder, and output layer.

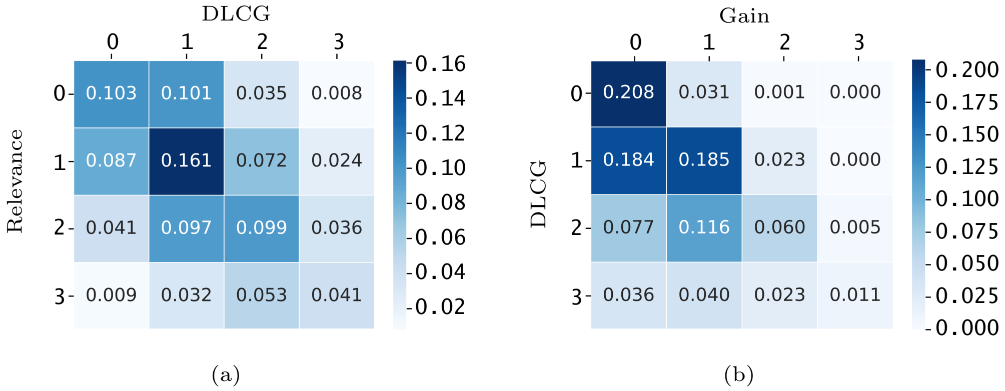
5.1 Passage Encoder

We continue to use the notation introduced in Subsection 3.1. To capture the semantic matching between query q and each passage p_i, we use the pre-trained Chinese BERT `BERT-Base-Chinese`[7] to obtain a representation for each passage. As shown in Fig.9, we use the output embedding of the first token as the representation for the entire query-passage pair:

$$\boldsymbol{P}_i = \texttt{BERT}(q, p_i).$$

5.2 Sequential Encoder

According to the definition of DPCG and QPCG, g_i represents the gain of the first i passages $\{p_1, p_2, ..., p_i\}$. g_i is not only determined by p_i, but also related to the former passages. Therefore, we use a recurrent neural network LSTM to model the passages. We show that the PCG sequence is non-decreasing and g_i is related to g_{i-1} in Section 4. Therefore g_{i-1} should also be taken as an input when modeling the i-th passage. The initial PCG g_0 is set to 0. Then we concatenate the



(a) (b)

Fig.8. Joint distributions of (a) relevance and DLCG, and (b) DLCG and gain.

[7] https://github.com/google-research/bert/blob/master/multilingual.md, May 2022.

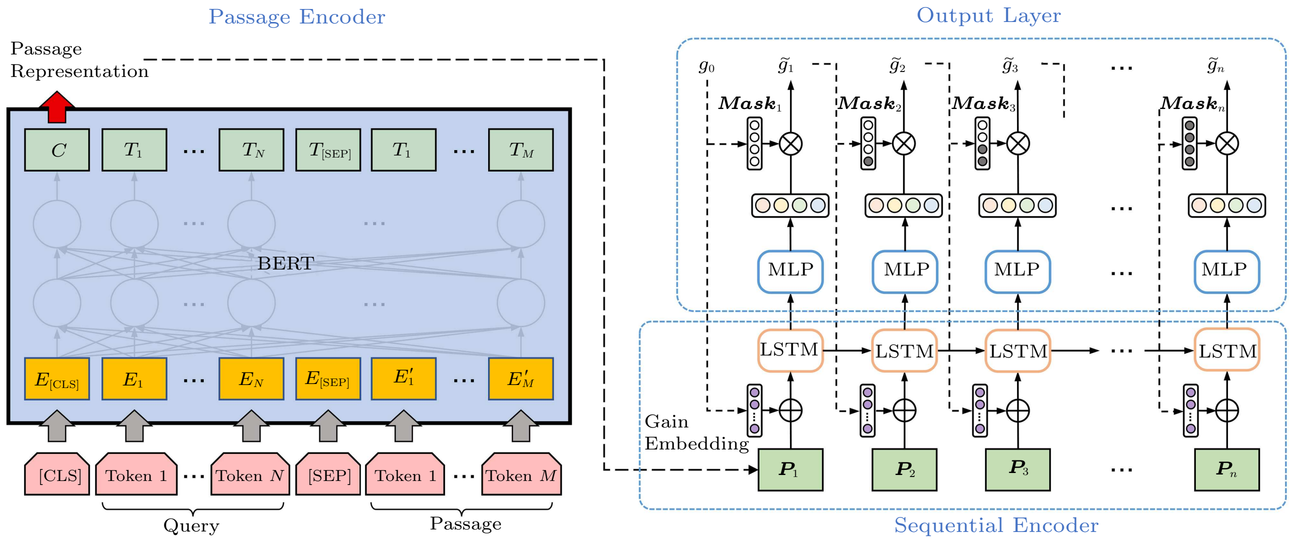


Fig.9. Model architecture of PCGM [23].

i-th passage representation $\boldsymbol{P}_i$ and the previous gain embedding $\boldsymbol{E}_{i-1}$ as the input of an LSTM cell.

$$\begin{aligned}
&g_0 = 0,\\
&\boldsymbol{E}_{i-1} = GainEmbedding(g_{i-1}),\\
&\boldsymbol{U}_i = (\boldsymbol{P}_i, \boldsymbol{E}_{i-1}),\\
&\boldsymbol{V}_1, \boldsymbol{V}_2, ..., \boldsymbol{V}_n = \mathrm{LSTM}(\boldsymbol{U}_1, \boldsymbol{U}_2, ..., \boldsymbol{U}_n),
\end{aligned}$$

where $GainEmbedding$ is an embedding layer, and $\boldsymbol{V}_i$ denotes the output vector of LSTM at the i-th step. Through LSTM, we update the passage representation by adding the content information and PCG of previous passages into it. Thus, we consider $\boldsymbol{V}_i$ as a context-aware passage representation.

5.3 Output Layer

After LSTM, we use a multilayer perceptron (MLP) with two fully connected layers to get a four-dimensional vector for the four PCG grades. The activation function we use is tanh. We also apply a dropout layer between the two fully connected layers to avoid the over-fitting problem.

$$\begin{aligned}
&\boldsymbol{V}_i' = \tanh(\boldsymbol{W}_v \boldsymbol{V}_i + \boldsymbol{b}_v),\\
&\boldsymbol{V}_i'' = \mathrm{dropout}(\boldsymbol{V}_i'),\\
&\boldsymbol{O}_i = \boldsymbol{W}_o \boldsymbol{V}_i'' + \boldsymbol{b}_o,
\end{aligned}$$

where $\boldsymbol{W}_v \in \mathbb{R}^{|\boldsymbol{V}_i'| \times |\boldsymbol{V}_i|}$, $\boldsymbol{b}_v \in \mathbb{R}^{|\boldsymbol{V}_i'|}$ and $\boldsymbol{W}_o \in \mathbb{R}^{|\boldsymbol{O}_i| \times |\boldsymbol{V}_i''|}$, $\boldsymbol{b}_o \in \mathbb{R}^{|\boldsymbol{O}_i|}$, $|\boldsymbol{O}_i| = 4$. We adopt a gain mask to keep the predicted PCG sequence monotonically incremental as we found in Subsection 3.3. The mask vector $\boldsymbol{Mask}_i$ is a four-dimensional binary vector for the four grades of PCG annotations. With the previous PCG g_{i-1}, only the PCG grades that are not less than g_{i-1} are possible to be predicted. We adopt an element-wise product between $\boldsymbol{Mask}_i$ and $\boldsymbol{O}_i$. Finally, in the output layer, we use softmax to obtain the predicted probabilities $\boldsymbol{P}_i$ for the four PCG grades.

$$\begin{aligned}
\boldsymbol{Mask}_i &= (m_0^i, m_1^i, m_2^i, m_3^i),\\
m_j^i &= \begin{cases} 0, \text{ if } j < g_{i-1},\\ 1, \text{ if } j \geqslant g_{i-1}, \end{cases}\\
\boldsymbol{P}_i &= \mathrm{softmax}(\boldsymbol{Mask}_i \odot \boldsymbol{O}_i)\\
&= (P(g_i = 0),\ P(g_i = 1),\\
&\quad P(g_i = 2),\ P(g_i = 3)).
\end{aligned}$$

We use stochastic gradient descent (SGD) to update the parameters of PCGM and adopt cross-entropy as the loss function for PCG sequence prediction:

$$\mathcal{L}_\theta = -\tfrac{1}{n}\textstyle\sum_{i=1}^{n} \log(P(g_i)) + \beta||\Delta_\theta||^2,$$

where θ is the parameter set of PCGM, n is the number of passages within the document, $P(g_i)$ is the predicted probability of the PCG label g_i, and β is the weight for L2 normalization.

To summarize, PCGM is a BERT-based sequential model that incorporates context-aware sequence information, including both passages' textual information and PCG signals. In Section 6, Section 7, and Section 8, we investigate the effectiveness of PCGM and the effect of the gain embedding and the gain mask.

6 Passage Cumulative Gain Prediction

In this section, we aim to answer RQ2: can we effectively predict the sequences of document-level and query-level passage cumulative gain based on the raw text of queries and documents? To address this research question, we use the PCGM introduced in Section 5 to predict DPCG sequences of documents in the TianGong-PDR dataset and QPCG sequences of query sessions in the SearchSuccess dataset. We compare the performance of PCGM with several baseline models. Finally, we analyze the effect of different components (i.e., the gain embedding and the gain mask) in PCGM through ablation experiments.

6.1 Experimental Settings

Baselines. We adopt two baseline methods, including a feature-based traditional machine learning model GBDT and a feature-based deep learning model LSTM. Extracted passage-level features are used as input of these two baselines. We extract eight text-based features for each passage, including the passage length (the number of words in the passage), the average TF, IDF, and TF × IDF values of query terms in the passage, scores of BM25 and three language models.

- *GBDT.* Since the GBDT cannot capture the context information, the input features consist of two parts when predicting the i-th PCG value. The first part contains the eight extracted features of the i-th passage. The second part contains the maximum, the minimum, and the mean values of the features of the top $i-1$ passages. Therefore, the length of the feature vector is 32 for the GBDT baseline. We consider the prediction task a four-category classification task and finally get a four-dimensional probability vector for each passage.

- *LSTM.* Each passage is represented by an eight-dimensional vector. We feed the sequence of passage vectors into an LSTM network. We use a multilayer perceptron and softmax to get a four-dimensional vector for each passage, which is taken as the predicted probabilities of the four grades of PCG.

Model Settings. We train two models for DPCG and QPCG prediction tasks, respectively. We also implement ablation experiments to further investigate the effectiveness of the gain embedding and the gain mask. We remove both the gain embedding and the gain mask (i.e., `PCGM w/o Embed and Mask`), only the gain embedding (i.e., `PCGM w/o Embed`), and only the gain mask (i.e., `PCGM w/o Mask`) to see how the model performance changes. PCGM for the DPCG prediction task is trained on the TianGong-PDR dataset. PCGM for the QPCG prediction task is trained on the SearchSuccess dataset. Details about these two datasets are described in Subsection 3.3. We divide the dataset into five sets and conduct five-fold cross-validation. We use four sets as the training sets and one set as the test set in each fold. Early stopping with the patience of 10 epochs is adopted during the training process on each fold.

To obtain the query-aware passage embeddings for PCGM, we use a public and effective implementation of BERT[8] based on PyTorch. We concatenate the query description and passage as the input and directly use the output embedding of the first token as the passage embedding. When predicting the DPCG/QPCG sequences, we set the maximum sequence length of LSTM and PCGM to 20/120, the largest passage number of a document/query session. For the LSTM baseline, the dimension of the passage embedding based on extracted features is 8, and that of the hidden vectors is 8. For the PCGM, the dimension of the passage embedding obtained by BERT is 768 and those of the hidden vectors and gain embeddings are 100 and 150, respectively. We use the $(i-1)$-th DPCG/QPCG labels for generating the gain embedding and the gain mask inputs when predicting the i-th DPCG/QPCG. Parameters are optimized using the Adam [42] with a batch size of 32, a learning rate of 0.001, and a dropout rate of 0.1.

Evaluation Metrics. The prediction outputs are the probabilities of the PCG value equal to 0, 1, 2, or 3. We evaluate the results using three metrics: the Log-likelihood (LL), the accuracy, and the Pearson correlation coefficient (PCC). When calculating the accuracy, we regard the task as a classification task. The PCG value with the maximum predicted probability is taken as the predicted class. When calculating the PCC, we use the expectation of predicted PCG probabilities as the predicted ranking score.

6.2 Results and Analysis

Overall Results. Overall performance is shown in Table 3. For the two baselines, LSTM performs better than GBDT, which shows that the context-aware model is more effective in modeling PCG sequences than the context-free model, although we extract features from the context as the input of the context-free

[8] https://github.com/huggingface/transformers, May 2022.

Table 3. PCG Prediction Performance of Different Methods over the TianGong-PDR Dataset

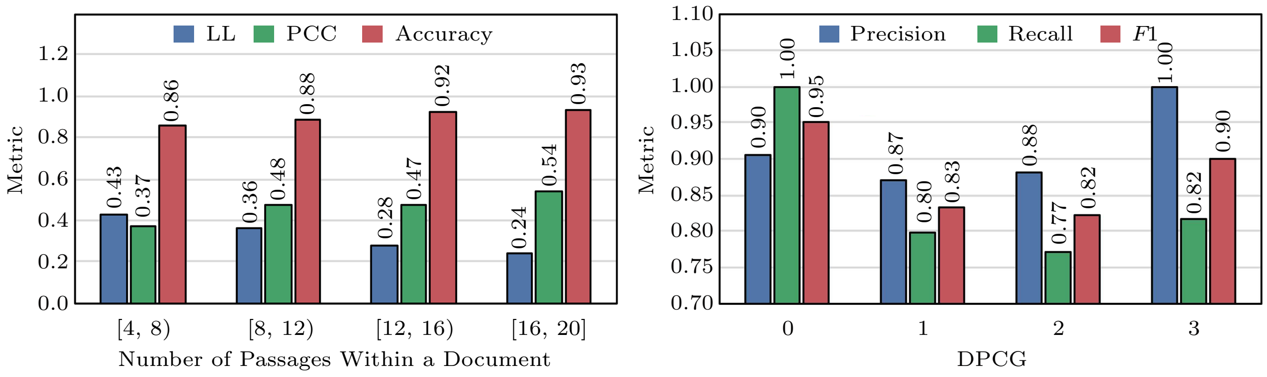
Model	DPCG[23]			QPCG		
	LL	PCC	Accuracy	LL	PCC	Accuracy
GBDT	1.260 4*	0.381 7*	0.490 6*	1.311 4*	0.302 8*	0.369 8*
LSTM	1.085 4*	0.432 7	0.527 2*	1.118 0*	0.734 5*	0.485 8*
`PCGM w/o Embed and Mask`	1.030 3*†	0.431 8	0.548 3*†	1.081 5*	0.734 5*	0.508 2*
`PCGM w/o Embed`	**0.336 2**†	**0.460 6**	**0.892 6**†	0.276 0†	**0.828 0**†	**0.912 7**†
`PCGM w/o Mask`	0.340 2†	0.459 2	**0.892 6**†	0.275 9†	**0.828 0**†	**0.912 7**†
PCGM	0.338 6†	0.459 6	**0.892 6**†	**0.274 8**†	0.826 1†	0.911 2†

Note: "*/†" denotes that compared with PCGM/LSTM (the best baseline), the performance difference is statistically significant using Tukey's HSD test. The best results in each group are marked in bold.

model. The framework of `PCGM w/o Embed and Mask` is the same as the LSTM baseline except for the passage encoder. `PCGM w/o Embed and Mask` outperforms the LSTM baseline as well. This shows that passage embeddings obtained by BERT are more effective than extracted passage features. Our PCGM performs significantly better than the GBDT and LSTM baselines on LL and accuracy, demonstrating that we can effectively predict DPCG and QPCG sequences using the BERT and sequence model.

Model Ablation. We remove the gain embedding and the gain mask from PCGM, both or one at a time, and observe the impact on the performance compared with the full model. The performance of `PCGM w/o Embed and Mask` is significantly worse than that of PCGM, while the performance of `PCGM w/o Embed` and `PCGM w/o Mask` is similar to that of PCGM. This shows that when using the real previous PCG label to generate the gain embedding and gain mask, one of them is enough to take full advantage of the previous PCG information. `PCGM w/o Embed` performs better than `PCGM w/o Mask`. Compared with the gain embedding, the gain mask is more effective for ranking. It is worth noting that `PCGM w/o Embed` performs slightly better than PCGM. This may be because the gain mask generated from the real previous PCG label is sufficient. Adding an extra gain embedding increases the model complexity and makes it more challenging.

Experimental Analysis. We further analyze the DPCG prediction performance of PCGM over documents of different lengths and passages with different DPCG labels, as shown in Fig.10. We use the number of passages within a document as the document length and find that as the length increases, the performance of PCGM also increases. This shows that PCGM can still capture the context information well, even in long documents. We use the *argmax* of probabilities of four DPCG grades as the predicted DPCG grade and analyze the precision, recall, and $F1$-score over passages with different DPCG labels. The results show that no-gain passages can be totally and correctly predicted, and all of the passages which are predicted as high-gain passages are indeed high-gain passages. According to the $F1$ scores, PCGM performs better over no-gain and high-gain passages than over low-gain and moderate-gain passages.



(a) (b)

Fig.10. PCG prediction performance of PCGM on (a) documents of different lengths and (b) passages with different DPCGs[23].

7 Document Ranking

In this section, we aim to answer RQ3: Can the passage cumulative gain be applied to estimate document relevance and improve the performance of document ranking models? To address this research question, we apply PCGM to predict the ranking scores of documents over the TianGong-PDR dataset (experiment 1). Furthermore, we test the performance of PCGM over another public document ranking test set, NTCIR-14 Web Chinese test collection (experiment 2). We compare the performance with several advanced baseline models to show the effectiveness of PCGM.

7.1 Experiment 1: Ranking on TianGong-PDR Dataset

Baselines. In this experiment, we choose three types of baselines to compare with PCGM: the classical probabilistic ranking model, document-level BERT-based neural ranking models, and passage-level BERT-based neural ranking models.

- *BM25*[1]. Although a number of neural ranking models have been proposed, BM25 is still a challenging baseline to beat[33].
- *BERT-Doc*[21]. We use the pre-trained Chinese BERT and fine-tune its output layer for predicting document-level ranking scores.

For passage-level BERT-based baselines, we adopt BERT-MaxP, BERT-FirstP, and BERT-SumP according to Dai and Callan[11].

- *BERT-MaxP*. The document score is determined by the maximum score of passages within the document.
- *BERT-FirstP*. The document score is determined by the score of the first passage.
- *BERT-SumP*. The document score is determined by summing all predicted passage scores.

Experimental Settings. We directly use the models trained for the DPCG sequence prediction task in Section 6 to test the ranking performance on the TianGong-PDR dataset, including PCGM, `PCGM w/o Embed`, `PCGM w/o Mask`, and `PCGM w/o Embed and Mask`. Note that we use the $(i-1)$-th DPCG label as input to generate the gain embedding and the gain mask for predicting the i-th DPCG value during the training process. While during the test process in this subsection, we use the DPCG values predicted by PCGM at the $(i-1)$-th step as the input of the i-th step. Specifically, we sample a DPCG value according to the $(i-1)$-th step's predicted probabilities and take this value as the previous DPCG input in the i-th step. For each data in the test set, we repeat the testing process 100 times and use the mean of 100 predicted DPCG probabilities as the predicted probability. Finally, we use the expectation of predicted probabilities of the last passage as the document's predicted ranking score.

For the BERT-based baselines, we use the same pre-trained BERT[9] in Section 6 and Mean squared error (MSE) as the loss function. The parameters are optimized by Adam[42] with a batch size of 32 and an initial learning rate of 5e$-$5 as same as those used by Devlin *et al.*[21]. For BERT-Doc, the query description and the entire document are concatenated as the input. The document would be truncated to 512 words if its length exceeded 512 words (the maximum input length). The DLCG label is used as the document's relevance label, and the model is trained to predict the relevance between the query and the document. To avoid overfitting, only the last linear layer is trained. For the passage-level BERT-based baselines, the query description and each passage are concatenated as the input. We first fine-tune the pre-trained BERT model based on the passage relevance annotations in the TianGong-PDR dataset and then use it to predict each passage's relevance in the test set. Only the last encoder and the output layer of BERT are trained to avoid overfitting.

Evaluation Metric. The DLCG label (i.e., the last value of the DPCG sequence) is used as the ground truth. We use three metrics to evaluate the ranking performance: nDCG[16], Q-measure[10], and nERR[43]. To examine the ranking performance of models at different ranking positions, we calculate nDCG at different cutoff positions, i.e., nDCG@{1, 3, 5, 10}. Since there are 15 documents for each query in the TianGong-PDR dataset, we also report the nDCG (i.e., nDCG@15), Q-measure, and nERR in the full ranked lists. We adopt Tukey's HSD test to examine the statistical significance of performance differences between different models.

Performance Comparison. Table 4 shows the ranking performance of different ranking models in the five-fold cross-validation on the TianGong-PDR dataset. For the two document-level ranking models, BERT-Doc performs better than BM25. It shows the capability of the pre-trained BERT. Among the three passage-

[9] https://github.com/huggingface/transformers, May 2022.

[10] Sakai T. New performance metrics based on multigrade relevance: Their application to question answering, 2004. https://research.nii.ac.jp/ntcir/workshop/OnlineProceedings4/OPEN/NTCIR4-OPEN-SakaiTrev.pdf, May 2022.

Table 4. Ranking Performance of Different Ranking Models over TianGong-PDR Dataset [23]

Model	nDCG					Q-Measure	ERR
	@1	@3	@5	@10	@15		
BM25	0.590	0.612	0.641	0.730	0.819	0.766	0.737
BERT-Doc	0.600	0.652	0.666	0.754	0.838	0.792	0.754
BERT-MaxP	0.555	0.596	0.625	0.731	0.809	0.763	0.713
BERT-FirstP	0.614	0.633	0.652	0.723	0.821	0.768	0.745
BERT-SumP	0.624	0.638	0.673	0.755	0.832	0.780	0.758
`PCGM w/o Embed and Mask`	0.617	0.632	0.663	0.748	0.827	0.777	0.747
`PCGM w/o Embed`	0.626	0.665	0.675	0.763	0.838	0.787	0.768
`PCGM w/o Mask`	0.645	0.670	0.685	0.767	0.843	0.794	0.777
PCGM	**0.688**	**0.686**	**0.696**	**0.780**	**0.850**	**0.798**	**0.800**

Note: The differences among models are not statistically significant using Tukey's HSD test. The best results in each group are marked in bold.

level BERT-based baselines, we can see that BERT-SumP performs the best, followed by BERT-FirstP. We consider that the performance of these three models is highly based on the effectiveness of their assumptions. BERT-SumP and BERT-Doc perform closely and win each other at different metrics. Overall, our PCGM model, which leverages BERT representations and fine-grained passage-level signals, gets the best ranking performance.

Model Ablation. When comparing PCGM and its sub-models, we can see that the design of the gain embedding and the gain mask is effective and can help PCGM achieve better performance. `PCGM w/o Mask` performs best among the three sub-models, indicating that the gain embedding is more effective than the gain mask in improving the performance of PCGM. The performance of `PCGM w/o Embed and Mask` is close to that of BERT-Doc, showing that our design of the gain embedding and the gain mask is effective to take advantage of DPCG in the document ranking task. When predicting the DPCG and QPCG sequences in Section 6, we use the DPCG/QPCG label of the previous passage to generate the gain embedding and the gain mask of the current input. Both of the gain embedding and the gain mask can provide accurate information about the previous PCG value. Therefore, either of them is enough to take full advantage of the previous PCG information. `PCGM w/o Embed` and `PCGM w/o Mask` both achieve similar performance with PCGM in Table 3. However, we use the DPCG values predicted by PCGM at the $(i-1)$-th step to generate the gain mask and the gain embedding of the input of the i-th step. In this situation, only using one of the gain mask and the gain embedding may not be enough to reflect all information about the previous DPCG ground truth value. Therefore, PCGM, which uses both the gain mask and the gain embedding, performs better than other ablation models as Table 4 shows.

7.2 Experiment 2: Ranking on NTCIR-14 Web Chinese Test Collection

In this subsection, we further examine whether PCGM trained with DPCG annotations on the TianGong-PDR dataset will still be effective on other public document ranking datasets. We evaluate our model over the NTCIR-14 Web Chinese test collection[11]. Documents in this test collection are the top-ranked documents by BM25 from a large-scale Chinese Web corpus, Sogou-T [44]. There are 79 queries and 4 816 documents in the NTCIR-14 Web Chinese test collection in total. It uses a four-grade relevance scale (irrelevant, fairly relevant, relevant, and highly relevant) and contains relevance annotations for all query-document pairs collected through high-quality crowdsourcing. We compare the performance of PCGM with plenty of ranking models: 1) BM25 and the BERT-based baseline models that are introduced in experiment 1; 2) a number of recently proposed neural ranking models including the ARC-I [29], ARC-II [29], DRMM [30], MatchPyramid [31], PACRR [28], KNRM [32], DeepRank [14], HiNT [13], and RIM [33] (see Subsection 2.2 for details).

Experimental Settings. Different from the news document of TianGong-PDR, the documents of NTCIR-14 Web Chinese test collection are extracted from raw Web pages, where there is no paragraph information. They are usually not well-organized and contain many independent but short texts. Therefore,

[11] http://www.thuir.cn/ntcirwww2/, May 2022.

to test the passage-level baselines, we set a sliding window with a size of 200 Chinese characters and an overlap of 50 Chinese characters according to Callan[8] to split the documents of NTCIR-14 Web Chinese test collection into multiple passages.

The training details of PCGM and BERT-based baselines are the same as those in experiment 1. For all the neural ranking baselines, we adopt the same implementation and training strategy with Li *et al.*[33], which is implemented by PyTorch based on Matchzoo[45]. We use Sogou-QCL[46] to train the neural ranking baselines, which is a large-scale public benchmark dataset for document ranking and consists of 537 366 queries, 5 480 860 documents, and various kinds of click model-based relevance labels, such as PSCM[47], UBM[48] and so on. In the NTCIR-14 Web Chinese task, Zheng *et al.*[49] won the first place by using Sogou-QCL to train their own document-level neural ranking models. We randomly split the Sogou-QCL into two parts for training and validation separately. The validation set contains 200 queries, and the training set contains the other queries. We use the PSCM-based relevance label as the supervision in the training processes. We adopt a pointwise loss function, Mean Squared Error (MSE), for the RIM and a pairwise hinge loss function for the other baseline models. We apply Adadelta[50] as the optimizer during the training process with a batch size of 80 and an initial learning rate of 0.1. We use an early stop strategy with patience of 10 epochs to get the best models over the test set.

Performance Comparison. We report the performance of ranking models over the NTCIR-14 Web Chinese test collection in Table 5. There are four types of models: BM25 for the probabilistic ranking model, 10 document-level ranking models, three passage-level ranking models, and our PCGM. We can see that BM25 performs rather well and outperforms most document-level neural ranking baselines, except BERT-Doc, on several metrics, which is consistent with Li *et al.*[33]. Among all the document-level neural ranking baselines, BERT-Doc performs the best, indicating the effectiveness of the pre-trained language model in the document ranking task. When comparing the three passage-level BERT-based baseline models, we see that BERT-MaxP performs the best and outperforms all BM25 and document-level neural ranking models. We find that both BERT-MaxP and BERT-SumP outperform BERT-Doc. This may be because BERT-Doc can only process the first 512 words of a document, which loses a lot of document information. PCGM achieves the best performance among all the metrics. Our results show that its improvements in nDCG@1, nDCG@5, and nDCG@15 over BERT-Doc are 12.1%, 15.3%, and 11.8% respectively. Due to the small query size in the NTCIR-14 Web Chinese test collection (only 79 queries), these improvements over most of the baselines are not statistically significant. The experimental results not only show the effectiveness of PCGM but also

Table 5. Ranking Performance of Different Ranking Models over NTCIR-14 Web Chinese Test Collection[23]

Model	nDCG					Q-Measure	ERR
	@1	@3	@5	@10	@15		
BM25	0.432	0.443	0.438	0.471	0.490	0.423	0.575
ARC-I	0.397	0.400*	0.427	0.451	0.461*	0.413	0.541
ARC-II	0.422	0.425	0.433	0.445*	0.473	0.424	0.562
DRMM	0.357	0.413	0.430	0.467	0.486	0.434	0.555
MatchPyramid	0.388	0.374*	0.374*	0.415*	0.433*	0.375*	0.519*
PACRR	0.403	0.459	0.455	0.469	0.483	0.427	0.556
KNRM	0.458	0.435	0.447	0.468	0.493	0.427	0.562
DeepRank	0.443	0.437	0.447	0.461	0.489	0.443	0.559
HiNT	0.397	0.380*	0.399*	0.421*	0.449*	0.393	0.534
RIM	0.475	0.458	0.464	0.467	0.478	0.428	0.577
BERT-Doc	0.462	0.464	0.472*	0.497	0.516	0.449	0.613
BERT-MaxP	0.505	0.505	0.515	0.539	0.557	0.498	0.637
BERT-FirstP	0.431	0.462	0.476	0.508	0.531	0.469	0.593
BERT-SumP	0.485	0.486	0.486	0.498	0.521	0.462	0.621
PCGM	**0.518**	**0.538**	**0.544**	**0.562**	**0.577**	**0.515**	**0.661**

Note: "*" denotes that compared with PCGM, the performance difference is statistically significant using Tukey's HSD test. The best results in each group are marked in bold.

show that the DPCG annotations are valuable.

To summarize experiments 1 and 2, experimental results show that PCGM can effectively learn from fine-grained DPCG signals with the design of the gain embedding and the gain mask, which helps PCGM outperform all the baseline models on both the TianGong-PDR dataset and the NTCIR-14 Web Chinese test collection.

8 Marginal Relevance Estimation for Document Ranking

In Section 6 and Section 7, we show the effectiveness of PCGM in cumulative gain prediction and document ranking tasks respectively. PCGM, which simulates the process of seeking useful information within a document, outperforms multiple advanced ranking models. By analyzing the QPCG sequence, we find that in a query session, when the useful information in a document is duplicated with the documents users have read, the increment of the user's gain is less than the document's DLCG level. However, we do not consider the cross-document effect when ranking the documents. The ranking list is ordered by declining predicted relevance or DLCG scores to the query in Section 7. In this section, we aim to investigate how to estimate a more accurate marginal relevance score[19], which considers both relevance and novelty. Carbonell and Goldstein[19] proposed the classical Maximal Marginal Relevance (MMR) method for this task. They measured the relevance and novelty scores of a document independently and used a linear combination of these two scores as the marginal relevance score. Different from them, we try to leverage PCGM to estimate the marginal relevance and conduct preference tests to show the effectiveness of PCGM.

8.1 Problem Definition and Method

Given a query q and a set of documents D, we aim to construct a ranking list RL, which ranks the document with higher relevance and higher novelty at the top position. Following the basic idea of MMR, we select documents one by one from D to put in RL by maximizing the marginal relevance.

$$\arg\max_{d_i \in RL\backslash D}\{f(d_i|q,RL)\}, \tag{1}$$

where $RL\backslash D$ is the set of yet unselected documents in D, and f is the score function which scores d_i considering both the query and already selected documents. Carbonell and Goldstein[19] used a linear combination as the score function:

$$\begin{aligned}&f_{\mathrm{MMR}}(d_i|q,RL)\\ &=\lambda Sim_1(d_i,q)-(1-\lambda)\max_{d_j\in RL} Sim_2(d_i,d_j),\end{aligned}$$

where Sim_1 and Sim_2 are two similarity functions, and λ is a parameter, whose value is in the interval $[0,1]$. It computes a standard relevance-based ranking list when $\lambda = 1$, and computes a diversity-based ranking list when $\lambda = 0$. This method has been proved effective in document ranking and summarization tasks. However, the explicit combination of relevance and similarity scores may be too simple to model the inner relationship. In this paper, we use PCGM introduced in Section 5 to estimate the marginal relevance (MR):

$$\begin{aligned}MR_{d_i} &= \mathbb{E}(PCGM(q,[RL;d_i]))-\\ &\quad\ \mathbb{E}(PCGM(q,RL)),\\ PCGM(q,RL) &= [P(g_N=0),P(g_N=1),\\ &\quad\ P(g_N=2),P(g_N=3)],\end{aligned}$$

where N is the total number of passages in RL, g_N is the predicted gain of the last passage, $\mathbb{E}$ means getting the expectation of predicted probability vectors, and $[;]$ means the concatenation. Since $PCGM(q,RL)$ is a constant value when RL is fixed, (1) can be simplified as:

$$\arg\max_{d_i\in RL\backslash D}\{\mathbb{E}(PCGM(q,[RL;d_i]))\}.$$

That is to say, in each ranking step, we concatenate every document d_i in $RL\backslash D$ with RL and use PCGM to calculate a new expectation of gain. Then, we select the document which achieves the greatest gain expectation and add it to the ranking list. The step repeats until we obtain a ranking list with a fixed length. In this paper, we start by ranking the top two documents (i.e., the length of the ranking list is 2).

8.2 Dataset

We use the TianGong-PDR dataset to construct the training data and the testing data. For the training set, we sample 700 document pairs (d_1,d_2) (10 for each query) from TianGong-PDR. Each pair is regarded as a ranking list $[d_1,d_2]$ whose length is 2. To train the PCGM, we first collect QPCG annotations for each ranking list. The instruction and the procedure are the same as those in Subsection 3.2.2. Ranking lists belonging to the same query are randomly allocated into one of the 10 groups without repetition. Therefore, each group contains 70 ranking lists. We recruit

30 participants. Each participant needs to annotate one group. Three different participants annotate each group. It takes them around three hours to finish 70 tasks. Each is paid around \$30 as compensation. The value of Hayes and Krippendorff's α for all annotations is 0.689. It indicates a moderate agreement level. Details about the training data are shown in Fig.11. We can see that documents' relevance at the first and the second position has a similar distribution. The training data covers all possible QLCG cases.

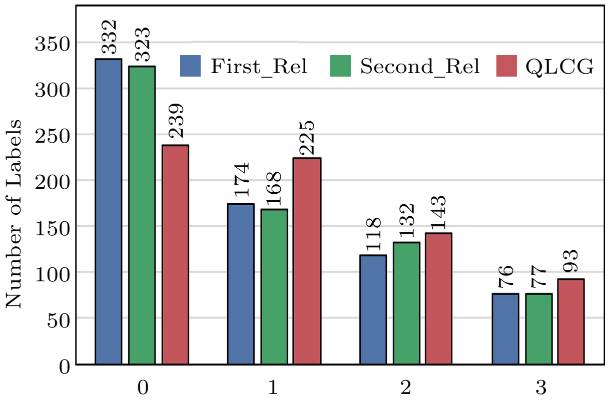


Fig.11. Distributions of four-grade relevance of the first/second documents (i.e., First_Rel/Second_Rel) in the ranking list and the four-grade QLCG of the ranking.

To test the performance of our model, we construct a testing set containing users' preference annotations and investigate whether users prefer the document with a higher predicted score. We sample 210 pairs of ranking lists from the TianGong-PDR dataset following three rules. 1) Because it is difficult to judge preference on two irrelevant documents, we only reserve the documents whose relevance scores are no less than 2. 2) The first documents in the ranking lists of one pair are the same. 3) The relevance scores of the second document in the ranking lists of one pair are the same. The sampling process can be described as follows: 1) constructing a candidate documents set D with all of the documents whose relevance scores are no less than 2; 2) sampling a document d from D as the first document of the ranking list; 3) sampling two documents with same relevance scores from $\{d\}\backslash D$. Finally, we obtain a ranking list pair $\{[d_1,d_2],[d_1,d_3]\}$, where $Rel_{d_2}=Rel_{d_3}$.

Then we collect users' preference annotations on the testing set. All pairs in the testing set are randomly allocated into three groups without repetition. We recruit 15 participants, and each participant needs to annotate one group. Five participants annotate each group. For each ranking list pair $\{[d_1,d_2],[d_1,d_3]\}$, we first show the query and d_1 to participants and ask them to read it carefully. Then we ask them to write the useful information they find briefly to ensure they read the document carefully. After that, we show d_2 and d_3 on the screen side by side. Participants should read them and decide which one they prefer. Finally, they are asked to give a preference annotation according to a five-level preference criterion from -2 (i.e., Left+2, indicating that the left one is much better than the right one) to 2 (i.e., Right+2, indicating that the right one is much better than the left one). To avoid position bias, we randomly put two ranking lists on the left or the right. We use the summation of the five annotations to determine the final preference judgments. We define three types of preference.

Same. There is no significant preference between the two ranking lists. The absolute value of the summation of five preference judgments is smaller than 2.

Weak Preference. Users have a weak preference between the two ranking lists. The absolute value of the summation of five preference judgments is equal to or larger than 2 and smaller than 5.

Strong Preference. Users have a strong preference between the two ranking lists. The absolute value of the summation of five preference judgments is equal to or larger than 5.

The distribution of preference types is shown in Fig. 12. Users prefer one ranking list compared with the other one in 159 pairs (75.7% of all the 210 pairs). Users have a strong preference in 36 pairs. It shows that even the relevance scores of two documents are the same, users have a preference between these two documents in most cases. It is not reasonable to only consider the relevance when ranking documents.

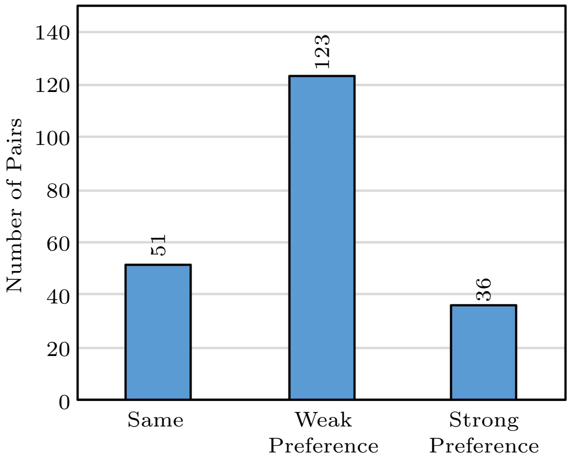


Fig.12. Distribution of preference annotations.

8.3 Experimental Settings and Results

We train the PCGM on the training data with QPCG labels using five-fold cross-validation and the early stopping strategy with patience of 10. The labeled data is randomly divided into five sets, ensuring that the data belonging to one query is in the same set. In each fold, we use four sets as the training set and one as the validation set. The maximum sequence length is 40. Other settings are the same as those introduced in Subsection 6.1. We also run the classical MMR model as baselines. We use five-fold cross-validation to train the parameter λ. We use BERT to get query-aware passage embedding. Then we use the element-wise summation of passage embeddings as the document embedding. Finally, we use (1 − the standardized Euclidean distance) as the similarity score of two documents.

We only focus on the predicted results of the 159 pairs where users have a significant preference. Ranking list A is predicted to be preferred compared with ranking list B when the expectation of PCGM's output of A is greater than that of B. The accuracy of predicted results is shown in Fig.13. We report the experimental results on the data with weak preference, the data with strong preference, and all the 159 pairs, respectively. We also report the prediction performance of relevant signals (i.e., BM25 scores) and MMR. "0.711" in Fig.13 means that in 71.1% of pairs, predicted results are consistent with users' preference annotations. We find the followings. 1) The relevance and MMR models perform slightly worse on strong preference data than weak preference data. However, our PCGM performs better on the strong preference data. 2) PCGM performs the best among these methods on both the subsets and all data. It shows the advantages of PCGM on the marginal relevance estimation task.

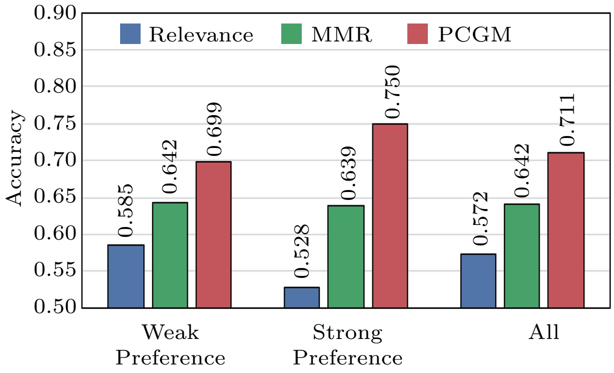


Fig.13. Performance of relevance, MMR, and PCGM methods on the data with weak preference, the data with strong preference, and all data.

9 Conclusions

In this paper, we investigated how users' information gain accumulates through passages within a document and within a query session. To the best of our knowledge, this is the first work to propose passage cumulative gain (PCG) and study how to apply it to document ranking tasks. We first defined document-level PCG (DPCG) and query-level PCG (QPCG). We collected DPCG and QPCG annotations through multiple lab-based user studies for a document-ranking dataset TianGong-PDR and an exploratory search dataset SearchSuccess. Analysis of the annotations showed that the DPCG sequence of a document is always non-decreasing. We also found that the increment of QPCG after users read a document is less than its DLCG label, and it is common that QPCG stops increasing before the query session ends. Based on the findings of DPCG and QPCG patterns, we proposed a BERT-based sequential model PCGM for modeling PCG sequences. Experimental results showed the effectiveness of the PCGM with the gain embedding and the gain mask on the PCG sequence prediction task. When applying PCGM to the document ranking related tasks, we find that PCGM outperforms multiple advanced ranking baselines. The marginal relevance scores predicted by PCGM are highly consistent with users' preferences. This work provided a new method for document ranking by leveraging the DPCG and QPCG sequences and improved the performance of document ranking.

There are also several potential limitations to this work. We assume that passages in documents are read sequentially and completely according to the previous research on reading behavior analysis. However, this assumption does not hold in some situations because users may have different reading habits. In the future, we plan to model the skipping behavior and the stop reading behavior to address this limitation. Since our focus in this work is to verify whether our proposed passage cumulative gain model (PCGM) structure is effective in PCG sequence predicting and document ranking, we directly use the most popular original BERT model rather than the state-of-the-art model to obtain the initial passage embeddings, and then update the embeddings using RNN. We would like to try more effective passage encoders and fine-tune them in future work. When estimating the marginal relevance, we consider only the top two results in the ranking list. In the future, we plan to consider a more practical situation and rank the top 10 results. We also plan to consider query types to better understand how users perceive QPCG under different intents. We believe a deeper understanding of the QPCG sequence can further help improve the document ranking performance.

References

[1] Robertson S E, Walker S. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *Proc. the 17th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*, Jul. 1994, pp.232-241. DOI: 10.1007/978-1-4471-2099-5_24.

[2] Ponte J M. A language modeling approach to information retrieval [Ph.D. Thesis]. University of Massachusetts, 1998.

[3] Zhai C, Lafferty J. A study of smoothing methods for language models applied to ad hoc information retrieval. *ACM SIGIR Forum*, 2017, 51(2): 268-276. DOI: 10.1145/3130348.3130377.

[4] Burges C J. From RankNet to LambdaRank to LambdaMART: An overview. Technical Report, MSR-TR-2010-82, Microsoft, 2010. https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/MSR-TR-2010-82.pdf, Apr. 2022.

[5] Liu T. Learning to Rank for Information Retrieval. Springer, 2011. DOI: 10.1007/978-3-642-14267-3.

[6] Pang L, Lan Y, Guo J, Xu J, Cheng X. A deep investigation of deep IR models. arXiv:1707.07700, 2017. https://arxiv.org/abs/1707.07700, May 2022.

[7] Clarke C L, Scholer F, Soboroff I. The TREC 2005 terabyte track. In *Proc. the 14th Text Retrieval Conference*, Nov. 2005.

[8] Callan J P. Passage-level evidence in document retrieval. In *Proc. the 17th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*, July 1994, pp.302-310. DOI: 10.1007/978-1-4471-2099-5_31.

[9] Kaszkiel M, Zobel J. Effective ranking with arbitrary passages. *Journal of the American Society for Information Science and Technology*, 2001, 52(4): 344-364. DOI: 10.1002/1532-2890(2000)9999:9999¡::AID-ASI1075¿3.0.CO;2-%23.

[10] Xi W, Xu R R, Khoo C S, Lim E P. Incorporating window-based passage-level evidence in document retrieval. *Journal of Information Science*, 2001, 27(2): 73-80. DOI: 10.1177/016555150102700202.

[11] Dai Z, Callan J. Deeper text understanding for IR with contextual neural language modeling. In *Proc. the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, Jul. 2019, pp.985-988. DOI: 10.1145/3331184.3331303.

[12] Wu Z, Mao J, Liu Y, Zhang M, Ma S. Investigating passage-level relevance and its role in document-level relevance judgment. In *Proc. the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, Jul. 2019, pp.605-614. DOI: 10.1145/3331184.3331233.

[13] Fan Y, Guo J, Lan Y, Xu J, Zhai C, Cheng X. Modeling diverse relevance patterns in ad-hoc retrieval. In *Proc. the 41st International ACM SIGIR Conference on Research and Development in Information Retrieval*, Jul. 2018, pp.375-384. DOI: 10.1145/3209978.3209980.

[14] Pang L, Lan Y, Guo J, Xu J, Xu J, Cheng X. DeepRank: A new deep architecture for relevance ranking in information retrieval. In *Proc. the 2017 ACM Conference on Information and Knowledge Management*, Nov. 2017, pp.257-266. DOI: 10.1145/3132847.3132914.

[15] Li X, Liu Y, Mao J, He Z, Zhang M, Ma S. Understanding reading attention distribution during relevance judgement. In *Proc. the 27th ACM International Conference on Information and Knowledge Management*, Oct. 2018, pp.733-742. DOI: 10.1145/3269206.3271764.

[16] Järvelin K, Kekäläinen J. IR evaluation methods for retrieving highly relevant documents. In *Proc. the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Jul. 2000, pp.41-48. DOI: 10.1145/345508.345545.

[17] Järvelin K, Kekäläinen J. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems*, 2002, 20(4): 422-446. DOI: 10.1145/582415.582418.

[18] Järvelin K, Price S L, Delcambre L M, Nielsen M L. Discounted cumulated gain based evaluation of multiple-query IR sessions. In *Proc. the 30th European Conference on Information Retrieval Research*, March 30-April 3, 2008, pp.4-15. DOI: 10.1007/978-3-540-78646-7_4.

[19] Carbonell J, Goldstein J. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proc. the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Aug. 1998, pp.335-336. DOI: 10.1145/290941.291025.

[20] Liu M, Liu Y, Mao J, Luo C, Zhang M, Ma S. "Satisfaction with failure" or "unsatisfied success": Investigating the relationship between search success and user satisfaction. In *Proc. the 2018 World Wide Web Conference*, Apr. 2018, pp.1533-1542. DOI: 10.1145/3178876.3186065.

[21] Devlin J, Chang M, Lee K, Toutanova K. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proc. the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Jun. 2019, pp.4171-4186. DOI: 10.18653/v1/N19-1423.

[22] Hochreiter S, Schmidhuber J. Long short-term memory. *Neural Computation*, 1997, 9(8): 1735-1780. DOI: 10.1162/neco.1997.9.8.1735.

[23] Wu Z, Mao J, Liu Y, Zhan J, Zheng Y, Zhang M, Ma S. Leveraging passage-level cumulative gain for document ranking. In *Proc. the Web Conference 2020*, Apr. 2020, pp.2421-2431. DOI: 10.1145/3366423.3380305.

[24] Liu X, Croft W B. Passage retrieval based on language models. In *Proc. the 2002 ACM CIKM International Conference on Information and Knowledge Management*, Nov. 2002, pp.375-382. DOI: 10.1145/584792.584854.

[25] Wu Z, Mao J, Liu Y, Zhang M, Ma S. Investigating reading behavior in fine-grained relevance judgment. In *Proc. the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, Jul. 2020, pp.1889-1892. DOI: 10.1145/3397271.3401305.

[26] Hearst M A, Plaunt C. Subtopic structuring for full-length document access. In *Proc. the 16th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*, June 27-July 1, 1993, pp.59-68. DOI: 10.1145/160688.160695.

[27] Salton G, Allan J, Buckley C. Approaches to passage retrieval in full text information systems. In *Proc. the 16th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*, June 27-July 1, 1993, pp.49-58. DOI: 10.1145/160688.160693.

[28] Hui K, Yates A, Berberich K, De Melo G. PACCR: A position-aware neural IR model for relevance matching. In *Proc. the 2017 Conference on Empirical Methods in Natural Language Processing*, Sept. 2017, pp.1049-1058. DOI: 10.18653/v1/D17-1110.

[29] Hu B, Lu Z, Li H, Chen Q. Convolutional neural network architectures for matching natural language sentences. In *Proc. the 27th International Conference on Neural Information Processing Systems*, Dec. 2014, pp.2042-2050.

[30] Guo J, Fan Y, Ai Q, Croft W B. A deep relevance matching model for ad-hoc retrieval. In *Proc. the 25th ACM International Conference on Information and Knowledge Management*, Oct. 2016, pp.55-64. DOI: 10.1145/2983323.2983769.

[31] Pang L, Lan Y, Guo J, Xu J, Wan S, Cheng X. Text matching as image recognition. In *Proc. the 30th AAAI Conference on Artificial Intelligence*, Feb. 2016, pp.2793-2799.

[32] Xiong C, Dai Z, Callan J, Liu Z, Power R. End-to-end neural ad-hoc ranking with kernel pooling. In *Proc. the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, Aug. 2017, pp.55-64. DOI: 10.1145/3077136.3080809.

[33] Li X, Mao J, Wang C, Liu Y, Zhang M, Ma S. Teach machine how to read: Reading behavior inspired relevance estimation. In *Proc. the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, Jul. 2019, pp.795-804. DOI: 10.1145/3331184.3331205.

[34] Robertson S E. The probability ranking principle in IR. In *Readings in Information Retrieval*, Jones K S, Willett P (eds.), Morgan Kaufmann Publishers Inc., 1997, pp.281-286.

[35] Goffman W. A searching procedure for information retrieval. *Information Storage and Retrieval*, 1964, 2: 73-78. DOI: 10.1016/0020-0271(64)90006-3.

[36] Fuhr N. A probability ranking principle for interactive information retrieval. *Information Retrieval*, 2008, 11(3): 251-265. DOI: 10.1007/s10791-008-9045-0.

[37] Zuccon G, Azzopardi L A, Van Rijsbergen K. The quantum probability ranking principle for information retrieval. In *Proc. the 2nd Conference on the Theory of Information Retrieval*, Sept. 2009, pp.232-240. DOI: 10.1007/978-3-642-04417-5_21.

[38] Chen H, Karger D R. Less is more: Probabilistic models for retrieving fewer relevant documents. In *Proc. the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Aug. 2006, pp.429-436. DOI: 10.1145/1148170.1148245.

[39] Hayes A F, Krippendorff K. Answering the call for a standard reliability measure for coding data. *Communication Methods and Measures*, 2007, 1(1): 77-89. DOI: 10.1080/19312450709336664.

[40] Roitero K, Maddalena E, Demartini G, Mizzaro S. On fine-grained relevance scales. In *Proc. the 41st International ACM SIGIR Conference on Research and Development in Information Retrieval*, Jul. 2018, pp.675-684. DOI: 10.1145/3209978.3210052.

[41] Sarkar P, Pillai J S. User expectations of augmented reality experience in Indian school education. In *Proc. the 7th International Conference on Research into Design*, Jan. 2019, pp.745-755. DOI: 10.1007/978-981-13-5977-4_63.

[42] Kingma D P, Ba J. Adam: A method for stochastic optimization. arXiv:1412.6980, 2014. https://arxiv.org/abs/1412.6980, May 2022.

[43] Sakai T, Song R. Evaluating diversified search results using per-intent graded relevance. In *Proc. the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*, Jul. 2011, pp.1043-1052. DOI: 10.1145/2009916.2010055.

[44] Luo C, Zheng Y, Liu Y, Wang X, Xu J, Zhang M, Ma S. SogouT-16: A new web corpus to embrace IR research. In *Proc. the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, Aug. 2017, pp.1233-1236. DOI: 10.1145/3077136.3080694.

[45] Guo J, Fan Y, Ji X, Cheng X. MatchZoo: A learning, practicing, and developing system for neural text matching. In *Proc. the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, Jul. 2019, pp.1297-1300. DOI: 10.1145/3331184.3331403.

[46] Zheng Y, Fan Z, Liu Y, Luo C, Zhang M, Ma S. Sogou-QCL: A new dataset with click relevance label. In *Proc. the 41st International ACM SIGIR Conference on Research and Development in Information Retrieval*, Jul. 2018, pp.1117-1120. DOI: 10.1145/3209978.3210092.

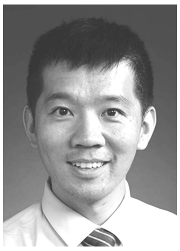
[47] Wang C, Liu Y, Wang M, Zhou K, Nie J, Ma S. Incorporating non-sequential behavior into click models. In *Proc. the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, Aug. 2015, pp.283-292. DOI: 10.1145/2766462.2767712.

[48] Dupret G E, Piwowarski B. A user browsing model to predict search engine click data from past observations. In *Proc. the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Jul. 2008, pp.331-338. DOI: 10.1145/1390334.1390392.

[49] Zheng Y, Chu Z, Li X, Mao J, Liu Y, Zhang M, Ma S. THUIR at the NTCIR-14 WWW-2 task. In *Proc. the 14th International Conference on NII Testbeds and Community for Information Access Research*, Jun. 2019, pp.165-179. DOI: 10.1007/978-3-030-36805-0_13.

[50] Zeiler M D. ADADELTA: An adaptive learning rate method. arXiv:1212.5701, 2012. https://arxiv.org/abs/1212.5701, May 2022.

Zhi-Jing Wu received her B.S. degree in computer science and technology from Tsinghua University, Beijing, in 2017. She is now a Ph.D. candidate in the Department of Computer Science and Technology at Tsinghua University, Beijing. Her research interests include user behavior modeling and document ranking.



Yi-Qun Liu received his B.S. and Ph.D. degrees in computer science and technology from Tsinghua University, Beijing, in 2003 and 2007, respectively. He is now a professor in the Department of Computer Science and Technology at Tsinghua University, Beijing. His research interests include Web search, user behavior analysis, and natural language processing.

Jia-Xin Mao received his B.S and Ph.D. degrees in computer science and technology from Tsinghua University, Beijing, in 2013 and 2018, respectively. He is now an assistant professor at Gaoling School of Artificial Intelligence, Renmin University of China, Beijing. His research interests include information retrieval, Web search, user behavior analysis, and dense retrieval.

Min Zhang received her B.S. and Ph.D. degrees in computer science and technology from Tsinghua University, Beijing, in 1999 and 2003, respectively. She is now an associate professor in the Department of Computer Science and Technology at Tsinghua University, Beijing. Her research interests include Web information retrieval and recommendation, user behavior analysis and profiling, machine learning, and data mining.

Shao-Ping Ma received his B.S., M.S., and Ph.D. degrees in computer science and technology from Tsinghua University, Beijing, in 1982, 1984, and 1997, respectively. He is now a professor in the Department of Computer Science and Technology at Tsinghua University, Beijing. His research interests include information retrieval and natural language processing.

JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY

Volume 37, Number 4, July 2022

Content

Special Section of Xia Peisu Young Scholars Forum 2021

Special Section of MASS 2020–2021

Regular Paper

JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY
《计算机科学技术学报》

Volume 37 Number 4 2022 (Bimonthly, Started in 1986)

Indexed in: **SCIE, Ei, INSPEC, JST, AJ, MR, CA, DBLP**

Edited by:
THE EDITORIAL BOARD OF JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY
Zhi-Wei Xu, Editor-in-Chief, P.O. Box 2704, Beijing 100190, P.R. China
Managing Editor: Feng-Di Shu E-mail: jcst@ict.ac.cn http://jcst.ict.ac.cn Tel.: 86-10-62610746

Copyright ©Institute of Computing Technology, Chinese Academy of Sciences 2022

Sponsored by: Institute of Computing Technology, CAS & China Computer Federation
Supervised by: Chinese Academy of Sciences
Undertaken by: Institute of Computing Technology, CAS
Published by: Science Press, Beijing, China
Printed by: Beijing Baochang Color Printing Co. Ltd
Distributed by:
China: All Local Post Offices
Other Countries: Springer Nature Customer Service Center GmbH, Tiergartenstr. 15, 69121 Heidelberg, Germany
Available Online: https://link.springer.com/journal/11390

ISSN 1000-9000

国内统一刊号: CN11-2296/TP 国内邮发代号: 2-578 RMB ¥160.00