Leveraging Passage-level Cumulative Gain for Document Ranking

Zhijing Wu, Jiaxin Mao, Yiqun Liu*, Jingtao Zhan, Yukun Zheng, Min Zhang, and Shaoping Ma
Department of Computer Science and Technology, Institute for Artificial Intelligence,
Beijing National Research Center for Information Science and Technology,
Tsinghua University, Beijing 100084, China
wuzhijing.joyce@gmail.com,yiqunliu@tsinghua.edu.cn

ABSTRACT

Document ranking is one of the most studied but challenging problems in information retrieval (IR) research. A number of existing document ranking models capture relevance signals at the whole document level. Recently, more and more research has begun to address this problem from fine-grained document modeling. Several works leveraged fine-grained passage-level relevance signals in ranking models. However, most of these works focus on context-independent passage-level relevance signals and ignore the context information, which may lead to inaccurate estimation of passage-level relevance. In this paper, we investigate how information gain accumulates with passages when users sequentially read a document. We propose the context-aware Passage-level Cumulative Gain (PCG), which aggregates relevance scores of passages and avoids the need to formally split a document into independent passages. Next, we incorporate the patterns of PCG into a BERT-based sequential model called Passage-level Cumulative Gain Model (PCGM) to predict the PCG sequence. Finally, we apply PCGM to the document ranking task. Experimental results on two public *ad hoc* retrieval benchmark datasets show that PCGM outperforms most existing ranking models and also indicates the effectiveness of PCG signals. We believe that this work contributes to improving ranking performance and providing more explainability for document ranking.

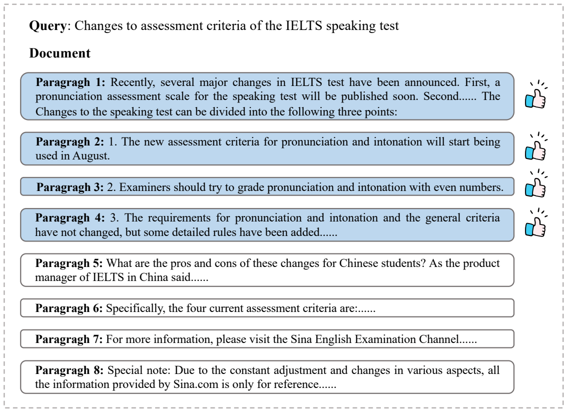
CCS CONCEPTS

• **Information systems → Information retrieval**; *Web search engines*; *Retrieval models and ranking*.

KEYWORDS

Passage-level cumulative gain, document ranking, neural network

ACM Reference Format:
Zhijing Wu, Jiaxin Mao, Yiqun Liu*, Jingtao Zhan, Yukun Zheng, Min Zhang, and Shaoping Ma. 2020. Leveraging Passage-level Cumulative Gain for Document Ranking. In *Proceedings of The Web Conference 2020 (WWW '20), April 20–24, 2020, Taipei, Taiwan.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3366423.3380305



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......

Figure 1: An example of high-gain documents to the query "Changes to assessment criteria of the IELTS speaking test". The document is relevant and meets the information needs in the query although only the first four paragraphs within it are relevant.

1 INTRODUCTION

Document ranking is one of the main challenges in information retrieval research. Given a query and a set of documents, document ranking aims to assign a relevance score for each query-document pair and then rank the documents in descending order according to the scores. For many existing ranking methods, no matter whether they are unsupervised (e.g. BM25 [34] and language models [32, 42]) or supervised (e.g. learning to rank [1, 22] or deep ranking [27]), they usually capture relevance signals at the whole document level.

When humans judge the relevance of documents (e.g. the assessment in TREC *ad hoc* task [3]), a document is typically considered relevant if any part of the document contains useful information: according to the *Scope Hypothesis* [34], the relevant parts could be in any position of a long document. This is in contrast to a sub-document-level (or passage-level) relevance judgment process. Figure 1 shows an example document retrieved for the query "Changes to assessment criteria of the IELTS speaking test". We can see that the document is composed of 8 paragraphs and only 4 of them are relevant to the query. It was assessed to be a "high-gain document" in our user study (see Section 3 for more details), which can totally satisfy users' information needs. However, it may be difficult to capture these local relevance signals if we only focus on whole-document-level features.

To solve this problem and help ranking models capture local signals, several works propose to estimate document relevance

WWW '20, April 20–24, 2020, Taipei, Taiwan

ACM ISBN 978-1-4503-7023-3/20/04.
https://doi.org/10.1145/3366423.3380305

based on fine-grained passage-level relevance signals. In these works [2, 17, 39], documents are split into passages based on textual discourse units (discourse passage), 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 were also proposed. Several researchers employed maximum, minimum or weighted summation functions, and tried to understand the relationship between passage-level relevance and document-level relevance by comparing these results [4, 19, 38]. Other methods turned to deep neural networks to automatically learn the relationship [7, 29]. These efforts led to improved ranking performance by introducing fine-grained relevance signals.

Meanwhile, most of these works make a simple assumption that the content of passages are independent from each other. With this context-independent assumption, passage-level relevance signals can be estimated separately. However, this assumption does not hold in many circumstances. For example, the second and third paragraphs of the example document in Figure 1 are too short for algorithms to accurately estimate their relevance. 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 query. Ignoring context information may lead to inaccurate estimation of passage-level relevance, and a better solution should take this information into consideration.

Different from these existing works, we try to estimate the passage-level cumulative gain (PCG) for document-level relevance estimation, rather than context-independent passage-level relevance signals. Taking the document shown in Figure 1 as an example, users' information needs can only be partially met if they only read a single one of the relevant paragraphs. The information needs are fully met only once they read all four relevant paragraphs. Therefore, we focus on how the information gain (i.e., useful information for the query) accumulates passage by passage when users read a document from top to bottom (here we assume that users will follow the sequential order while reading an article, which accords with findings in user reading behavior [20]). With this framework, we avoid the problem 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 [14, 15] and multiple session-level [16]. In this work, we investigate cumulative gain at the passage level, with the aim of capturing context-aware fine-grained relevance signals. Then we model the sequence of PCG with deep recurrent neural networks and leverage it for document ranking. To summarize, we investigate the following research questions:

- **RQ1:** How does the passage-level information gain accumulate during a user's information seeking process?
- **RQ2:** Can we effectively predict the sequence of passage-level cumulative gain based on the raw text of queries and documents?
- **RQ3:** Can the passage-level cumulative gain be applied to improve the performance of document ranking models?

To shed light on these research questions, we collect the annotations of PCG through a lab study on an existing *ad hoc* retrieval dataset, TianGong-PDR [38]. Based on the dataset, we firstly investigate the patterns of PCG to answer RQ1 by analyzing the PCG sequence and the transition of PCG. Then we define PCG prediction as a sequence prediction task and propose a new Passage-level Cumulative Gain Model (PCGM), which employs BERT [5] to learn initial representations for query-passage pairs and incorporates the observed patterns into an LSTM [11] to effectively predict PCG sequences. Finally, we leverage this model to estimate a relevance score for the whole document. We further test the ranking performance of PCGM over another public document ranking test set, NTCIR-14 Web Chinese test collection [26]. To summarize, the main contributions are as follows:

- To our best knowledge, we construct the first *ad hoc* retrieval dataset [1] which contains passage-level cumulative gain annotations.
- We provide a thorough analysis of the patterns by which passage-level information gain accumulates. This helps us better understand how information gain is perceived when users seek useful information in a document for a certain query intent.
- We show that the sequence of PCG can be effectively predicted by incorporating the observed PCG patterns into a deep neural network.
- We employ the PCG sequence into document ranking models and show its effectiveness in improving ranking performance on both the TianGong-PDR dataset and the NTCIR-14 Web Chinese test collection.

The remainder of this paper is organized as follows. We review related work in Section 2. Then we describe the passage-level cumulative gain and analyze the patterns of PCG to address RQ1 in Section 3. Section 4 describes the proposed model PCGM. The experimental setup and results of PCG prediction and document ranking are presented in Section 5 and 6 respectively. Section 7 concludes this work and suggests directions for future research.

2 RELATED WORK

In this section, we briefly review the related works on passage-level relevance and document ranking models.

2.1 Passage-level Relevance

Callan [2] proposed that with the increase of documents' length, it is natural to consider the fine-grained relevance, such as passage-level relevance, in ranking tasks. Several works have investigated the fine-grained passage-level relevance signals. It was found able to help better understand the relevance judgment process and be further used to improve the performance of document ranking [30, 38]. There were several methods to split a doucment into passages in previous works. Callan [2] 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 [38]. Semantic passages are derived from documents based on the subject or content of the

[1] The data is now available at http://www.thuir.cn/group/~YQLiu/

text [2]. Window passages have not taken the logical structure or semantic information of documents into consideration, but consist of a fixed number of words [39].

After splitting documents into passages, relevance signals are obtained from these passages, which is called passage-level relevance, then the passage-level relevance can be utilized to generate document-level relevance scores [4, 19, 23, 38]. For example, Wu et al. [38] recently used a four-grade relevance scale to annotate each passage of a document. Then they employed maximum, minimum or weighted summation functions to estimate the document-level relevance, which helps better understand the relationship between passage-level relevance and document-level relevance.

In this work, we take a paragraph as a passage and investigate the context-aware passage-level cumulative gain (PCG).

2.2 Document Ranking Models

A large number of models for document ranking have been proposed, including probability models (e.g., BM25 [34]), feature-based learning to rank models [1, 13, 22], and neural ranking models [27]. Neural ranking models have been shown to be effective at automatically learning ranking scores from raw text of queries and documents. Here we mainly review the development of neural ranking models in recent years.

Hu et al. [12] proposed CNN-based ARC-I and ARC-II for matching two sentences. The former gets the representation of the query and document, then compares the two representations to predict the ranking score. The latter first conducts the interaction between the matrixes of the query and document, then predict the ranking score. Deep Relevance Matching Model (DRMM) [8] use matching histogram mapping as the input, and combines a feed forward matching network and a term gating network to consider query term importance. MatchPyramid [28] models text matching as the problem of image recognition by using convolution approaches. Position-Aware Convolutional Recurrent Relevance Matching (PACRR) [13] uses convolutional layers to capture both term matching and positional information based on the query-document interactions. Kernel-based Neural Ranking Model (KNRM) [40] uses a kernel-pooling technique to extract multi-level soft match features between the query and document. All of these models capture matching signals at the whole document level.

Recently, some works try to address the document ranking problem by incorporating fine-grained passage-level matching signals. Pang et al. [30] proposed DeepRank, which first detects relevant locates, then determines local relevance, and finally aggregates local relevance to get the document-level ranking score. Fan et al. [7] followed the same idea and proposed Hierarchical Neural Matching Model (HiNT). Li et al. [21] proposed Reading Inspired Model (RIM) based on the inspiration of users' reading behavior patterns, which first captures the sentence-level relevance signals and then modeling the document-level relevance according to reading heuristics from human. BERT [5] is an effective pre-trained language model trained on large-scale, open-domain corpus, and can be used to obtain the representation of texts. Based on BERT, Dai and Callan [4] took the maximum, first, and summation of matching scores of query-passage pairs as document-level ranking scores, and show the effectiveness of BERT on the document ranking task.

Table 1: Statistics of our dataset.

#Query	#Document	#Passage	#PCG annotation
70	1,050	11,512	34,536

In this work, we use BERT to learn the passage representation and incorporate the historical PCG information into a RNN model to predict the following PCG sequence and document relevance.

3 PASSAGE-LEVEL CUMULATIVE GAIN

In this section, we first describe the procedure of passage-level cumulative gain annotations. Then we conduct a thorough analysis of the patterns of passage-level cumulative gain to address RQ1.

3.1 Definition

We start by defining passage-level cumulative gain (PCG). Given a query and a document, considering that users usually follow the sequential order while reading an article [20], 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 $q = \{q_1, q_2, ..., q_m\}$ and a document $d = \{p_1, p_2, ..., p_n\}$, where q_i is the i-th term in the query and p_i is the i-th passage in the document, the PCG labels of d can be described as a sequence $G_d = \{g_1, g_2, ..., g_n\}$, where g_i $(1 \leq i \leq n)$ 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 (i.e., g^d). In this work, we take one paragraph as one passage, following Wu et al. [38], and use a four-grade PCG judgment scale (i.e., $g_i \in \{0, 1, 2, 3\}, 1 \leq i \leq n$).

3.2 Data Collection

3.2.1 *Task and Participants.*

To investigate PCG in document ranking, we first collect the PCG annotations for a recent and public *ad hoc* retrieval dataset, TianGong-PDR[2] [38]. 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.[3] There are 15 documents for each query and 564 words per document on average. In this work, we conduct a lab-based study to collect the PCG annotations for this dataset.

By posting posters around the campus and social networks, we recruited 45 participants in this study, 19 males and 26 females. They are all undergraduate and graduate students from a university. Their ages are 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. Each participant was paid about $15 as compensation.

3.2.2 *Procedure.*

Each annotation task involves a query-document pair. In each task, participants need to read the passages within the document one by one and annotate the PCG sequence for the whole document. We ask participants to carefully read the instructions and

[2] http://www.thuir.cn/data-pdr/

[3] http://thuctc.thunlp.org/

Table 2: The distributions of four-grade document-level cumulative gain (DLCG) and passage-level cumulative gain (PCG). The Avg. #P and Avg. #W mean the average number of passages and words within documents, respectively.

Document-level				Passage-level	
Type	Proportion	Avg. #P	Avg. #W	Type	Proportion
$DLCG = 0$	0.390	10.8	536	$PCG = 0$	0.527
$DLCG = 1$	0.208	10.5	548	$PCG = 1$	0.230
$DLCG = 2$	0.187	10.9	585	$PCG = 2$	0.136
$DLCG = 3$	0.215	11.8	605	$PCG = 3$	0.107
All	1	11.0	562	All	1

then guide them to finish two training tasks to help them quickly get familiar with the experimental system and instructions. In the beginning of each task, the system will show the search query, search intent description, and the first passage in the document to the participant, who then needs to annotate the PCG degree of the first passage, after reading it. Next, both the first and second passages will be presented and the participant gives the PCG degree 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 PCG for the query-document pair from the participant.

There are totally 70 queries and 15 documents for each query in the dataset. For each query, each participant needs to annotate one document. Therefore, each participant needs to complete 70 annotation tasks (i.e., 70 query-document pairs) in total. All tasks are shown to participants in random order. It takes about one and a half hours to finish these 70 tasks. Each query-document pair is annotated by three different participants.

3.2.3 PCG Annotation Instructions. We use a four-grade PCG annotation in the study, which reflects the degree of gain. The instructions for the four-grade PCG annotation are as follows:

- **No gain (0)**: There is no useful information for the information needs behind the query in the content you have read.
- **Low gain (1)**: Based on the content you have read, the information needs behind the query can be slightly satisfied.
- **Moderate gain (2)**: Based on the content you have read, the information needs behind the query can be fairly satisfied.
- **High gain (3)**: Based on the content you have read, the information needs behind the query can be totally satisfied.

3.2.4 Collected Dataset. Statistics of our dataset are shown in Table 1. There are 70 queries, and 1,050 documents consisting of 11,512 passages in the dataset. For each document, we obtain three sequences of PCG from three different annotators. Therefore, we have totally 34,536 PCG annotations. After collecting the PCG annotations, we first use Krippendorff's α [10] for ordinal data to measure the inter-person agreement of PCG annotations. The value is 0.625, which indicates a moderate agreement level. Table 2 shows the distributions of DLCG and PCG annotations. About 21.5% of documents in the dataset can fully satisfy the information needs. The high-gain documents contain more passages and more words than other kinds of documents on average. Figure 2 shows fractions of PCG annotations in the no-gain (0), low-gain (1), moderate-gain (2), and high-gain (3) documents respectively. In no-gain documents, all of the PCG annotations are zero. We found that the PCG degree with the largest proportion in the other three kinds of documents matches their DLCG degree.

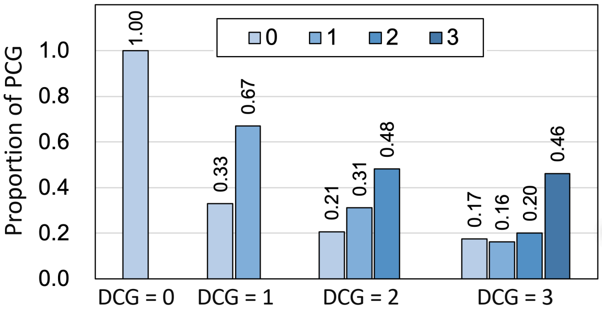


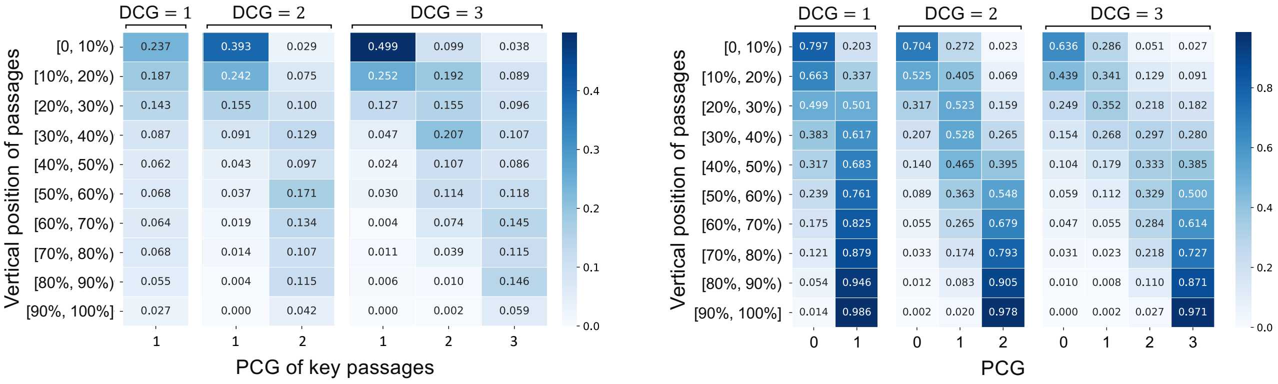
Figure 2: The joint distribution of document-level cumulative gain (DLCG) and passage-level cumulative gain (PCG).

3.3 Patterns of Passage-level Cumulative Gain

To answer RQ1, we analyze how the passage-level information gain accumulates when users are seeking useful information in a document based on the collected data.

We first look into how the PCG 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 PCG annotation for the first $i - 1$ passages, are shown in Figure 4. For example, the "0.905" in the bottom left corner of Figure 4 indicates that when g_{i-1} is zero, the probability for $g_i = 0$ is 0.905. The probabilities that g_{i-1} is greater than g_i are all zero, which shows that the PCG sequence of a document in our collected data is always a non-decreasing sequence. In other words, the useful information captured by users accumulates as they read more passages. This may be because that documents in the TianGong-PDR dataset are news articles. They are well written, structured, and that remains trustworthy throughout. 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 and loses all trust in the content and reduces the PCG grades. Probabilities on the diagonal line are 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 PCG increases from g_{i-1} to g_i, the increment is most likely to be one.

We define the passage where the PCG annotation is different from the previous one as the *key passage*. Since the PCG 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 PCG annotations increase at key passages. There are three kinds of key passages: low-gain key passages ($PCG = 1$), moderate-gain key passages ($PCG = 2$), and high-gain key passages ($PCG = 3$). We split passages within a document into ten parts according to their vertical positions and analyze the distribution of vertical positions of key passages. Figure 3(a) shows distributions of key passages in low-gain, moderate-gain, and high-gain documents. We did 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, the "0.499" in the first row means that in high-gain documents, 49.9% of low-gain key passages are located in the top 10% part of documents. Similarly, the "0.099" in the first row means that in high-gain documents, 9.9% of moderate-gain key passages are located in the top 10% part of documents.



(a)

(b)

Figure 3: The distributions of (a) key passages and (b) PCG annotations at different vertical positions in low-gain (*DLCG* = 1), moderate-gain (*DLCG* = 2), and high-gain (*DLCG* = 3) documents.

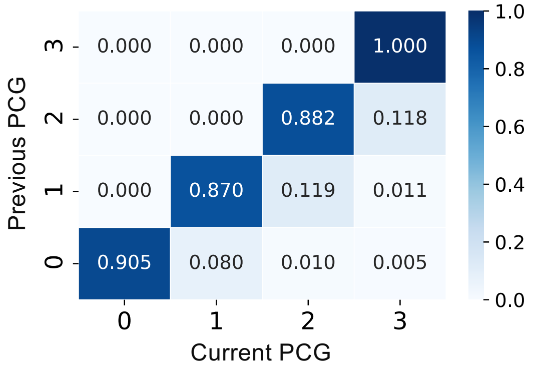


Figure 4: The transition probabilities of passage-level cumulative gain (PCG).

We find that the proportions of low-gain key passages tend to decay as the vertical position increases, which indicates that users usually obtain some useful information at the beginning of documents, except no-gain documents. The higher the cumulative gain of the document, the higher the vertical position of low-gain key passages and moderate-gain passages. When looking into the vertical position where the value of PCG 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.

Figure 3(b) shows distributions of PCG at different vertical positions. The "0.797" in the top left corner means that in 0%~10% part of low-gain documents, the probability that PCG equals zero is 0.797. We observe that in low-gain documents, the probability that PCG 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. This indicates that as the value of DLCG increases, more passages need to be read to judge an accurate DLCG.

3.4 Summary

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

4 PASSAGE-LEVEL CUMULATIVE GAIN MODEL

We have shown in Section 3 that the PCG sequence is non-decreasing and the current PCG is related to the previous one. In this section, we propose a Passage-level Cumulative Gain Model (PCGM) to leverage context-aware sequence information to address the PCG sequence prediction task, and further leverage the last value of the PCG sequence for document ranking.

The framework of PCGM is illustrated in Figure 5. It consists of three major components: passage encoder, sequential encoder, and output layer. The passage encoder aims to learn the semantic representations from both the query and the passage. The output of the passage encoder is then fed into the sequential encoder to generate a context-aware passage representation, which is finally fed into the output layer to predict the PCG sequence. When predicting the PCG, the previous PCG is also taken as an input for the sequential encoder and output layer. The details are described as follows.

4.1 Passage Encoder

We continue to use the notation introduced in Section 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`[4] to obtain a representation for each passage. As shown in Figure 5, we use the output embedding of the first token as the representation

[4] https://github.com/google-research/bert/blob/master/multilingual.md

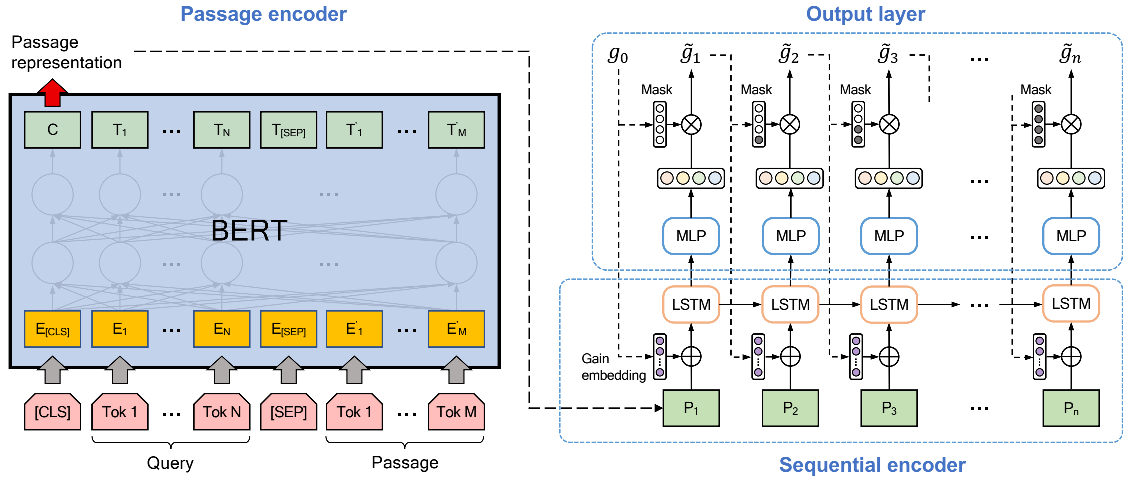


Figure 5: The model architecture of PCGM.

for the entire query-passage pair:

$$P_i = \text{BERT}(q, p_i) \tag{1}$$

4.2 Sequential Encoder

According to the definition of PCG, 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 showed that the PCG sequence is non-decreasing and g_i is related to g_{i-1} in Section 3.3. So g_{i-1} should also be taken as an input when modeling the i-th passage. We use a Gain Embedding Layer to get the corresponding embedding E_{i-1} for $g_i - 1$. The initial PCG g_0 is set to 0. Then we concatenate the i-th passage representation P_i and the previous gain embedding E_{i-1} as the input of an LSTM cell. We use V_i to represent 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 V_i as a context-aware passage representation.

$$E_{i-1} = \text{GainEmbedding}(g_{i-1}) \tag{2}$$

$$U_i = [P_i, E_{i-1}] \tag{3}$$

$$V_1, V_2, ...V_n = \text{LSTM}(U_1, U_2, ..., U_n) \tag{4}$$

4.3 Output Layer

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

$$V_i' = \tanh(W_v V_i + b_v) \tag{5}$$

$$V_i'' = \text{dropout}(V_i') \tag{6}$$

$$O_i = W_o V_i'' + b_o \tag{7}$$

where $W_v \in \mathbb{R}^{|V_i'| \times |V_i|}$, $b_v \in \mathbb{R}^{|V_i'|}$ and $W_o \in \mathbb{R}^{|O_i| \times |V_i''|}$, $b_o \in \mathbb{R}^{|O_i|}$, $|O_i| = 4$. We adopt a gain mask to keep the predicted PCG sequence monotonically incremental as we found in Section 3.3. The mask vector $Mask_i$ is a four-dimensional binary vector with respect to 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} is possible to be predicted. We adopt an element-wise product between $Mask_i$ and O_i. Finally in the output layer, we use $softmax$ to obtain the predicted probabilities P_i for the four grades of PCG.

$$Mask_i = [m_0^i, m_1^i, m_2^i, m_3^i] \tag{8}$$

$$m_j^i = \begin{cases} 0 & j < g_{i-1} \\ 1 & j \geqslant g_{i-1} \end{cases} \tag{9}$$

$$P_i = \text{softmax}(Mask_i \odot O_i) \tag{10}$$

$$P_i = [P(g_i = 0), P(g_i = 1),\ P(g_i = 2),\ P(g_i = 3)] \tag{11}$$

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

$$\mathcal{L}_\theta = -\frac{1}{n}\sum_{i=1}^{n} \log(P(g_i)) + \beta||\Delta_\theta||^2 \tag{12}$$

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, β is the weight for L2 normalization.

To summarize, PCGM is a BERT-based sequential model, which incorporates the context-aware sequence information, including both passages' textual information and PCG signals. In the following sections, we investigate the effectiveness of PCGM, as well as the effect of the gain embedding and gain mask.

5 PASSAGE-LEVEL CUMULATIVE GAIN PREDICTION

In this section, we answer **RQ2**: can we effectively predict the sequence of passage-level cumulative gain? To address this research question, we use the PCGM introduced in Section 4 to predict PCG sequences of documents in TianGong-PDR dataset and compare the performance with a number of baseline models. Further, We analyze the effect of different components in PCGM, i.e., the gain embedding and gain mask.

5.1 Experimental Settings

We adopt two baseline methods for comparison, including a feature-based traditional machine learning model GBDT and a feature-based deep learning model LSTM. These two baselines are based on extracted features. We extract eight learning-to-rank features for each passage according to Qin and Liu [33], including the passage length (the number of words in the passage), the average TF, IDF, and $TF \times IDF$ values of query terms in the passage, scores of BM25 and three language models with the query. Each passage is represented by an eight-dimensional vector.

For the LSTM baseline, we fed the sequence of passage vectors within a document into an LSTM network. Then 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. Considering that the GBDT can not capture the context information, when predicting the i-th PCG of a document, the input features consist of two parts. The first part contains the eight learning-to-rank 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 as a four-category classification task and finally get a four-dimensional probability vector for each passage.

In addition, we also implement ablation experiments, which remove both the gain embedding and 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). We compare these three sub-models to further investigate the effectiveness of the gain embedding and gain mask.

We use the TianGong-PDR dataset with passage-level cumulative gain annotations as our dataset. Details about the dataset are described in Section 3.2. We divide the dataset into 5 sets and conduct five-fold cross-validation. In each fold, we use four sets as the training set and one set as the test set. Early stopping with the patience of 10 epochs is adopted during the training process on each fold. The parameters are optimized using the Adam [18] with a batch size of 32, a learning rate of 0.001, and a dropout rate of 0.1. For the LSTM model, 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, 150 respectively. The previous PCG used for the gain embedding and gain mask is the real label of the previous passage.

Three metrics are used to evaluate the prediction performance: the Log-Likelihood (LL), the Pearson Correlation Coefficient (PCC), and the accuracy. We use the expectation of probabilities of four PCG grades as the predicted ranking score when calculating the PCC. For the calculation of accuracy, we use the *argmax* of probabilities of four PCG grades as the predicted ranking score.

5.2 Results and Analysis

In this section, we compare the sequence prediction performance of our proposed PCGM and baseline models. We investigate PCGM on different documents and passages (i.e., documents of different lengths, and passages with different PCG labels) to better understand the model performance.

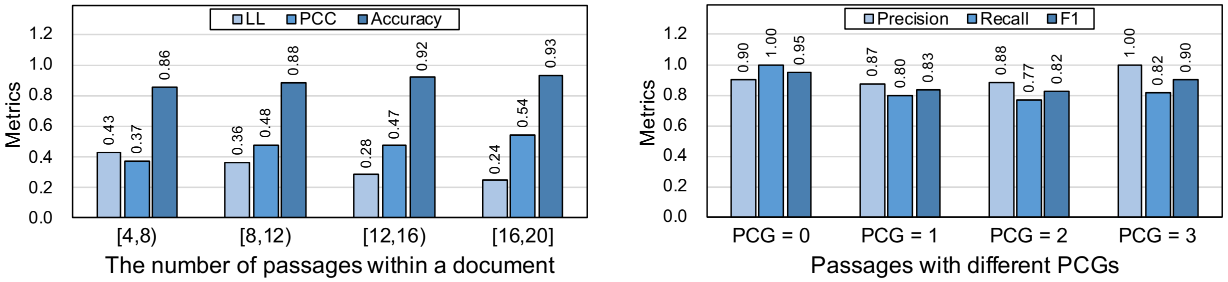
Table 3: PCG prediction performance of different methods over the TianGong-PDR dataset. "*/†" denotes that compared to PCGM/LSTM (the best baseline), the performance difference is statistically significant using Tukey's HSD test.

Model	LL	PCC	Accuracy
GBDT	1.2604*	0.3817*	0.4906*
LSTM	1.0854*	0.4327	0.5272*
PCGM w/o Embed and Mask	1.0303*†	0.4318	0.5483*†
PCGM w/o Embed	**0.3362†**	**0.4606**	**0.8926†**
PCGM w/o Mask	0.3402†	0.4592	**0.8926†**
PCGM	0.3386†	0.4596	**0.8926†**

5.2.1 Overall Results. Overall performance is shown in Table 3. For the two baselines, the LSTM performs better than the 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 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 proposed PCGM performs better than the GBDT and LSTM baselines significantly on the LL and accuracy, demonstrating that by using the BERT and sequence model, we can effectively predict PCG sequences.

5.2.2 Model Ablation. We remove the gain embedding and gain mask from PCGM, both or one at a time, and observe the impact on the performance compared to 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 to 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 that the gain mask generated from the real previous PCG label is sufficient. Adding an extra gain embedding increases the model complexity, and makes it harder.

5.2.3 Experimental analysis. We further analyze the performance of PCGM over documents of different lengths and passages with different PCG labels, as shown in Figure 6. We use the number of passages within a document as the length of the document, 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 PCG grades as the predicted PCG grade, and analyze the precision, recall, and F1-score over passages with different PCG labels. The results show that no-gain passages can be totally 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)

Figure 6: PCG prediction performance of PCGM on (a) documents of different lengths and (b) passages with different PCGs.

6 DOCUMENT RANKING

In this section, we aim to answer **RQ3**: can the passage-level cumulative gain be applied to improve the performance of document ranking models? To address this research question, we apply PCGM to predict the ranking scores of documents over 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 a number of advanced baseline models.

6.1 Evaluation Metric

We evaluate all ranking models using three metrics: nDCG [14], Q-measure [35] and nERR [36]. To examine the ranking performance of models at different ranking positions, we also 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 the Tukey's HSD test to examine the statistical significance of performance differences between different models.

6.2 Experiment 1: Ranking on TianGong-PDR

In this experiment, we examine the ranking performance of PCGM over the TianGong-PDR dataset. Since we already conducted five-fold cross-validation in the PCG sequence prediction task in Section 5, we apply the saved PCG models of five folds into this document ranking task. We take the predicted PCG of the last passage within a document as the predicted ranking score for this document, and take the DLCG label (i.e., the last value of the PCG sequence) as the ground truth. Then we evaluate the ranking performance of PCGM.

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

- **BM25** [34]: This model is a classical and efficient document-level ranking model. Although a number of neural ranking models have been proposed, BM25 is still a challenging baseline to beat [21, 24]. So we include it in this experiment.

For the document-level neural baseline, we choose:

- **BERT-Doc** [5]: BERT first models the interactions between two pieces of text with the attention mechanism and then predicts their relationship, which can be classified as the interaction-based neural ranking model [8]. We use the pre-trained Chinese BERT and fine-tune its output layer for predicting document-level ranking scores.

For the passage-level baseline, we adopt three BERT-based ranking models according to Dai and Callan [4]: BERT-MaxP, BERT-FirstP and BERT-SumP. In all these three models, we first fine-tune the pre-trained BERT model based on the relevance annotations for passages in the TianGong-PDR dataset, and then use it to predict the relevance of each passage in the test set independently. Finally, we obtain the document-level relevance by using different assumptions in the three models:

- **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 calculated by summing all predicted passage-level scores.

In addition, we use the three sub-models of PCGM, `PCGM w/o Embed`, `PCGM w/o Mask`, and `PCGM w/o Embed and Mask` in this experiment to examine the effect of the gain embedding and gain mask.

6.2.2 Experimental settings. We use a public and effective implementation of BERT[5] based on PyTorch [31]. For BERT-based models, the query description and one passage/the entire document are concatenated as the input. During fine-tuning, the document will be truncated to 512 words if its length exceeded 512 words (the maximum input length). For the document-level BERT, only the last linear layer is trained to avoid overfitting. For the same reason, only the last encoder and the output layer are trained in the passage-level BERTs. We use Mean squared error (MSE) as the loss function in the BERT-based baselines. The parameters are optimized by Adam [18] with a batch size of 32 and initial learning rate of 5e-5 as same as [5]. The fine-tuning processes converge after 3 epochs and 7 epochs for document-level and passage-level BERT-based baselines separately.

For PCGM and its sub-models, we directly use the models trained and saved in the PCG sequence prediction (Section 5). Note that during the training process, the previous PCG used for the gain

[5] https://github.com/huggingface/transformers

Table 4: Ranking performance of different ranking models over TianGong-PDR dataset. The differences among models are not statistically significant using Tukey's HSD test).

Model	nDCG@1	nDCG@3	nDCG@5	nDCG@10	nDCG@15	Q-measure	ERR
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**

embedding and gain mask is the real label of the previous passage, while in the test process of this document ranking task, we use the PCG predicted by PCGM at the previous step. Specifically, we use a random sampling method to obtain a predicted PCG 100 times, and we use the mean of each PCG label's probabilities as the predicted probability for this PCG label. Finally, we use the expectation of probabilities of four PCG grades as the predicted ranking score.

6.2.3 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, showing the capability of the pre-trained BERT. Among the three passage-level BERT-based baselines, we can see that BERT-SumP performs 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.

6.2.4 Model Ablation. When comparing among PCGM and its submodels, we can see that the design of gain embedding and gain mask is effective and can help PCGM to achieve a 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 BERT-Doc, showing that our design of the gain embedding and gain mask is effective to take advantage of PCG in the document ranking task.

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

With the best performance of PCGM in the Experiment 1, we examine whether PCGM trained with PCG annotations will still be effective in other public document ranking datasets. In Experiment 2, we use baseline models trained on the Sogou-QCL dataset [44]. Since there is no PCG label in the Sogou-QCL dataset, we can't train PCGM on this dataset and so directly use the saved PCGM models from Section 5. All these models are evaluated over a public test set for a document ranking task, NTCIR-14 Web Chinese test collection.[6]

6.3.1 Baselines. To compare with PCGM, we use BM25 and the BERT-based baseline models introduced in Experiment 1 as baselines. In addition, a number of recently proposed neural ranking models are adopted, including the ARC-I, ARC-II, DRMM, MatchPyramid, PACRR, KNRM, DeepRank, HiNT, and RIM, which have been introduced in Section 2.

6.3.2 Dataset. Sogou-QCL [44] is a large-scale public benchmark dataset for document ranking, which consists of 537,366 queries, 5,480,860 documents and various kinds of click model-based relevance labels, such as PSCM [37], UBM [6] and so on. In the NTCIR-14 Web Chinese task, Zheng et al. [43] won the first place by using Sogou-QCL to train their own document-level neural ranking models. Therefore, in Experiment 2, we follow them and use Sogou-QCL to additionally train the neural ranking baselines introduced in Section 6.3.1. For those baselines, 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 other queries. We use the PSCM-based relevance label as the supervision in the training processes.

For the NTCIR-14 Web Chinese test collection [26], its documents are the top-ranked documents from a large-scale Chinese Web corpus, Sogou-T [25], by using BM25. 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, which were collected through high-quality crowdsourcing.

6.3.3 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, 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 [2] to split the documents of NTCIR-14 Web Chinese test collection into multiple passages.

Since the training details of PCGM and BERT-based baselines have been introduced in Experiment 1, we only describe the experimental settings of several new ranking baselines in Experiment 2. For all neural ranking baselines introduced in Section 6.3.1, we adopt the implementation from Li et al. [21], which is implemented

[6] http://www.thuir.cn/ntcirwww2/

Table 5: Ranking performance of different ranking models over NTCIR-14 Web Chinese test collection. "*" denotes that compared to PCGM, the performance difference is statistically significant using Tukey's HSD test.

Model	nDCG@1	nDCG@3	nDCG@5	nDCG@10	nDCG@15	Q-measure	ERR
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**

by PyTorch based on Matchzoo [9]. We adopt a pointwise loss function, Mean Squared Error (MSE) for the RIM and a pairwise hinge loss function for other baseline models. We apply Adadelta [41] as the optimizer during the training process with a bacth size of 80 and initial learning rate of 0.1. We use early stop strategy with a patience of 10 epochs to get the best models over the test set.

6.3.4 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, ten document-level ranking models, three passage-level ranking models and our PCG model. 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 [21]. 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 can 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. We consider this is because BERT-Doc can only process the first 512 words of a document at most, 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%. Due to the small query size in NTCIR-14 Web Chinese test collection (only 79 quries), these improvements over most of the baselines are not statistically significant. The experimental results not only show the effectiveness of PCGM but also shows that the PCG annotations are valuable.

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

7 CONCLUSION

In this paper, we investigated how the information gain of users accumulates through passages within a document. To the best of our knowledge, this is the first work to propose passage-level cumulative gain (PCG) and study how to apply it to document ranking tasks. First, we defined PCG and collected PCG annotations for a public document-ranking dataset named TianGong-PDR through a lab-based user study. Analysis of the collected annotations demonstrated that the PCG sequence of a document is always monotonically increasing in the documents of TianGong-PDR. Based on the findings of PCCG patterns, we proposed a BERT-based sequential model PCGM for modeling PCG, which uses an LSTM after a pre-trained BERT with a gain embedding layer and a gain mask mechanism. We first evaluated PCGM in the PCG sequence prediction task, and showed the effectiveness of the gain embedding and gain mask. Then we applied PCGM in a document ranking task and showed that PCGM outperforms multiple advanced ranking baselines over two public datasets. We conducted a model ablation test and a comparison with BERT-based baselines to better understand PCGM. This work provides a new method for the document ranking problem by leveraging the PCG and improves the performance of document ranking.

In the future, we plan to study other approaches to leverage the PCG sequence for document ranking, for example, proposing new methods to utilize the last PCG annotation instead of directly adding a gain mask. We also plan to take the query type into consideration to better understand how users perceive PCG under different queries. We believe that a deeper understanding of the PCG can further help improve the document ranking performance.

ACKNOWLEDGMENTS

This work is supported by the National Key Research and Development Program of China (2018YFC0831700), Natural Science Foundation of China (Grant No. 61622208, 61732008, 61532011) and Beijing Academy of Artificial Intelligence (BAAI). We thank Prof. Falk Scholer at RMIT University for helpful comments.

REFERENCES

[1] Christopher JC Burges. 2010. From ranknet to lambdarank to lambdamart: An overview. *Learning* 11, 23-581 (2010), 81.

[2] James P. Callan. 1994. Passage-level Evidence in Document Retrieval. In *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (Dublin, Ireland) *(SIGIR '94)*. Springer-Verlag New York, Inc., New York, NY, USA, 302–310. http://dl.acm.org/citation.cfm?id=188490.188589

[3] Charles LA Clarke, Falk Scholer, and Ian Soboroff. 2005. The TREC 2005 Terabyte Track.. In *TREC*.

[4] Zhuyun Dai and Jamie Callan. 2019. Deeper Text Understanding for IR with Contextual Neural Language Modeling. In *Proceedings of the 42Nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'19)*.

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[6] Georges E. Dupret and Benjamin Piwowarski. 2008. A User Browsing Model to Predict Search Engine Click Data from Past Observations.. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (Singapore, Singapore) *(SIGIR '08)*. ACM, New York, NY, USA, 331–338. https://doi.org/10.1145/1390334.1390392

[7] Yixing Fan, Jiafeng Guo, Yanyan Lan, Jun Xu, Chengxiang Zhai, and Xueqi Cheng. 2018. Modeling Diverse Relevance Patterns in Ad-hoc Retrieval. In *Proceedings of the 41st International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '18)*.

[8] Jiafeng Guo, Yixing Fan, Qingyao Ai, and W. Bruce Croft. 2016. A Deep Relevance Matching Model for Ad-hoc Retrieval. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management (CIKM '16)*.

[9] Jiafeng Guo, Yixing Fan, Xiang Ji, and Xueqi Cheng. 2019. MatchZoo: A Learning, Practicing, and Developing System for Neural Text Matching. In *Proceedings of the 42Nd International ACM SIGIR Conference on Research and Development in Information Retrieval* (Paris, France) *(SIGIR'19)*. ACM, New York, NY, USA, 1297–1300. https://doi.org/10.1145/3331184.3331403

[10] Andrew F. Hayes and Klaus Krippendorff. 2007. Answering the Call for a Standard Reliability Measure for Coding Data. *Communication Methods and Measures* 1, 1 (2007), 77–89. https://doi.org/10.1080/19312450709336664

[11] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[12] Baotian Hu, Zhengdong Lu, Hang Li, and Qingcai Chen. 2014. Convolutional neural network architectures for matching natural language sentences. In *Advances in neural information processing systems*. 2042–2050.

[13] Kai Hui, Andrew Yates, Klaus Berberich, and Gerard de Melo. 2017. PACRR: A position-aware neural IR model for relevance matching. *arXiv preprint arXiv:1704.03940* (2017).

[14] Kalervo Järvelin and Jaana Kekäläinen. 2000. IR Evaluation Methods for Retrieving Highly Relevant Documents. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '00)*.

[15] Kalervo Järvelin and Jaana Kekäläinen. 2002. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems (TOIS)* 20, 4 (2002), 422–446.

[16] Kalervo Järvelin, Susan L Price, Lois ML Delcambre, and Marianne Lykke Nielsen. 2008. Discounted cumulated gain based evaluation of multiple-query IR sessions. In *European Conference on Information Retrieval*. Springer, 4–15.

[17] Marcin Kaszkiel and Justin Zobel. 2001. Effective ranking with arbitrary passages. *Journal of the American Society for Information Science and Technology* 52, 4 (2001), 344–364.

[18] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[19] K Kong, R Luk, K Ho, and F Chung. 2004. Passage-based retrieval using parameterized fuzzy set operators. In *ACM SIGIR Workshop on Mathematical/Formal. Methods for Information Retrieval*.

[20] Xiangsheng Li, Yiqun Liu, Jiaxin Mao, Zexue He, Min Zhang, and Shaoping Ma. 2018. Understanding Reading Attention Distribution During Relevance Judgement. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management (CIKM '18)*.

[21] Xiangsheng Li, Jiaxin Mao, Chao Wang, Yiqun Liu, Min Zhang, and Shaoping Ma. 2019. Teach Machine How to Read: Reading Behavior Inspired Relevance Estimation. In *Proceedings of the 42Nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'19)*.

[22] Tie-Yan Liu. 2009. *Learning to Rank for Information Retrieval*. Now Publishers Inc., Hanover, MA, USA.

[23] Xiaoyong Liu and W Bruce Croft. 2002. Passage retrieval based on language models. In *Proceedings of the eleventh international conference on Information and knowledge management*. ACM, 375–382.

[24] Cheng Luo, Tetsuya Sakai, Yiqun Liu, Zhicheng Dou, Chenyan Xiong, and Jingfang Xu. 2017. Overview of the ntcir-13 we want web task. *Proc. NTCIR-13* (2017).

[25] Cheng Luo, Yukun Zheng, Yiqun Liu, Xiaochuan Wang, Jingfang Xu, Min Zhang, and Shaoping Ma. 2017. SogouT-16: A New Web Corpus to Embrace IR Research. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Shinjuku, Tokyo, Japan) *(SIGIR '17)*. ACM, New York, NY, USA, 1233–1236. https://doi.org/10.1145/3077136.3080694

[26] Jiaxin Mao, Tetsuya Sakai, Cheng Luo, Peng Xiao, Yiqun Liu, and Zhicheng Dou. 2019. Overview of the ntcir-14 we want web task. In *Proceedings of the 14th NTCIR Conference on Evaluation of Information Access Technologies*.

[27] Liang Pang, Yanyan Lan, Jiafeng Guo, Jun Xu, and Xueqi Cheng. 2017. A deep investigation of deep ir models. *arXiv preprint arXiv:1707.07700* (2017).

[28] Liang Pang, Yanyan Lan, Jiafeng Guo, Jun Xu, Shengxian Wan, and Xueqi Cheng. 2016. Text matching as image recognition. In *Thirtieth AAAI Conference on Artificial Intelligence*.

[29] Liang Pang, Yanyan Lan, Jiafeng Guo, Jun Xu, Jingfang Xu, and Xueqi Cheng. 2017. DeepRank: A New Deep Architecture for Relevance Ranking in Information Retrieval. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management (CIKM '17)*.

[30] Liang Pang, Yanyan Lan, Jiafeng Guo, Jun Xu, Jingfang Xu, and Xueqi Cheng. 2017. Deeprank: A new deep architecture for relevance ranking in information retrieval. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*. ACM, 257–266.

[31] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in pytorch. (2017).

[32] Jay Michael Ponte and W Bruce Croft. 1998. *A language modeling approach to information retrieval*. Ph.D. Dissertation. University of Massachusetts at Amherst.

[33] Tao Qin and Tie-Yan Liu. 2013. Introducing LETOR 4.0 Datasets. *CoRR* abs/1306.2597 (2013). arXiv:1306.2597 http://arxiv.org/abs/1306.2597

[34] Stephen E Robertson and Steve Walker. 1994. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *Proceedings of the 17th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'94)*. Springer, 232–241.

[35] Tetsuya Sakai. 2004. New Performance Metrics Based on Multigrade Relevance: Their Application to Question Answering.. In *NTCIR*.

[36] Tetsuya Sakai and Ruihua Song. 2011. Evaluating Diversified Search Results Using Per-intent Graded Relevance. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Beijing, China) *(SIGIR '11)*. ACM, New York, NY, USA, 1043–1052. https://doi.org/10.1145/2009916.2010055

[37] Chao Wang, Yiqun Liu, Meng Wang, Ke Zhou, Jian-yun Nie, and Shaoping Ma. 2015. Incorporating Non-sequential Behavior into Click Models. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Santiago, Chile) *(SIGIR '15)*. ACM, New York, NY, USA, 283–292. https://doi.org/10.1145/2766462.2767712

[38] Zhijing Wu, Jiaxin Mao, Yiqun Liu, Min Zhang, and Shaoping Ma. 2019. Investigating Passage-level Relevance and Its Role in Document-level Relevance Judgment. In *Proceedings of the 42Nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'19)*.

[39] Wensi Xi, Richard Xu-Rong, Christopher SG Khoo, and Ee-Peng Lim. 2001. Incorporating window-based passage-level evidence in document retrieval. *Journal of information science* 27, 2 (2001), 73–80.

[40] Chenyan Xiong, Zhuyun Dai, Jamie Callan, Zhiyuan Liu, and Russell Power. 2017. End-to-end neural ad-hoc ranking with kernel pooling. In *Proceedings of the 40th International ACM SIGIR conference on research and development in information retrieval*. ACM, 55–64.

[41] Matthew D Zeiler. 2012. ADADELTA: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701* (2012).

[42] Chengxiang Zhai and John Lafferty. 2017. A study of smoothing methods for language models applied to ad hoc information retrieval. In *ACM SIGIR Forum*, Vol. 51. ACM, 268–276.

[43] Yukun Zheng, Zhumin Chu, Xiangsheng Li, Jiaxin Mao, Yiqun Liu, Min Zhang, and Shaoping Ma. [n.d.]. THUIR at the NTCIR-14 WWW-2 Task. ([n. d.]).

[44] Yukun Zheng, Zhen Fan, Yiqun Liu, Cheng Luo, Min Zhang, and Shaoping Ma. 2018. Sogou-QCL: A New Dataset with Click Relevance Label. In *The 41st International ACM SIGIR Conference on Research &#38; Development in Information Retrieval* (Ann Arbor, MI, USA) *(SIGIR '18)*. ACM, New York, NY, USA, 1117–1120. https://doi.org/10.1145/3209978.3210092