information

Article

Multi-Layer Contextual Passage Term Embedding for Ad-Hoc Retrieval

Weihong Cai *,†, Zijun Hu † , Yalan Luo †, Daoyuan Liang, Yifan Feng and Jiaxin Chen

College of Engineering, Shantou University, Shantou 515063, China; 20zjhu1@stu.edu.cn (Z.H.); 18ylluo@stu.edu.cn (Y.L.); 18dyliang@stu.edu.cn (D.L.); 20yffeng@stu.edu.cn (Y.F.); 20jxchen4@stu.edu.cn (J.C.)
* Correspondence: whcai@stu.edu.cn
† These authors contributed equally to this work.

Abstract: Nowadays, pre-trained language models such as Bidirectional Encoder Representations from Transformer (BERT) are becoming a basic building block in Information Retrieval tasks. Nevertheless, there are several limitations when applying BERT to the query-document matching task: (1) relevance assessments are applicable at the document-level, and the tokens of documents often exceed the maximum input length of BERT; (2) applying BERT to long documents leads to a great consumption of memory usage and run time, owing to the computational cost of the interactions between tokens. This paper explores a novel multi-layer contextual passage architecture that leverage text summarization extraction to generate passage-level evidence for the pre-selected document passage thus brought new possibilities for the long document relevance task. Experiments were conducted on two standard ad-hoc retrieval collections from the Text Retrieval Conference (TREC) 2004 Robust Track (Robust04) and ClueWeb09 with two different characteristics individually. Experimental results show that our approach can significantly outperform the strong baselines and even compared with the same BERT-based models, the precision of our methods as well as state-of-the-art neural ranking models.

Keywords: relevance matching; text summarization extraction; passage-level evidence

Citation: Cai, W.; Hu, Z.; Luo, Y.; Liang, D.; Feng, Y.; Chen, J. Multi-Layer Contextual Passage Term Embedding for Ad-Hoc Retrieval. *Information* **2022**, *13*, 221. https://doi.org/10.3390/info13050221

Academic Editor: Norbert Fuhr

Received: 20 March 2022
Accepted: 19 April 2022
Published: 25 April 2022

1. Introduction

Query document relevance matching plays a dominant role in information retrieval. Typically, given a query and a list of candidate documents, a ranking function is applied to produce scores that represent the relevance of each query and document pair. Traditional ad hoc retrieval task utilizes statistical features such as BM25 term weighting [1] and document scoring method as the ranking function [2]. Although the BM25 model achieves an improvement in the calculation of inverse document frequency (IDF) weights by controlling the scale of both term frequencies and document length, thus becoming the benchmark retrieval model of TREC; however, it fails when applying to documents which are very long [3]. The essence of the traditional information retrieval model is the exact matching between queries and documents, which ignored the semantics of the words. So as to solve the problem of word ambiguity, early information retrieval research focused on the use of traditional machine learning methods to solve such problems, such as topic model [4] and term proximity information [5]. However, the computation of handcraft features is time-consuming and the integrity would not have been enforced. In recent years, pre-trained language models have led to promising results in information retrieval (IR) tasks. Typically, models such as word2vec [6] have been widely used in neural IR. However, word co-occurrence is only a shallow bag-of-words model, which cannot avoid the ambiguity of word items. Subsequently, some more powerful language models such as Long Short-Term Memory (LSTM), Transformer, and Generative Pre-Training (GPT) were used to improve the performance of information retrieval tasks. Nevertheless, these models were trained unidirectionally, the self-attention mechanism will only focus on the previous n-grams, which will directly cause the representations learned by the model to be incomplete. More Recently, BERT [7] due to its bidirectional training mechanism, addressed the issue mentioned above. Different from traditional unidirectional models, BERT takes the entire text as input and is trained bidirectional with two novel strategies: "masked language model" (MLM) and "next sentence prediction" task, which takes the word dependencies and sentence structures into account. Several works proposed utilizing the prevalent model BERT for the query-document relevance task, for instance, Yang et al. [8] first apply BERT to infer individual sentences within a document, and then integrate the sentence scores into document scores. Dai et al. [9] proposed BERTIR model which takes the query and document as the input and utilizes BERT to achieve a better text understanding of the query and document text content. Although the BERT-based models have outperformed most previous neural IR models, whereas in fact there are several limitations when applying BERT to long documents. First of all, the representation of a long document can not capture the whole meaning of the text. Furthermore, applying neural networks to long documents leads to a great consumption of memory usage and run time, owing to the computational cost of the interactions between tokens. Because of these problems, A general-purpose, long-document-friendly model becomes more desirable. This paper explores a method that leverage text summarization extraction to generate passage-level evidence for the pre-segmented document thus brought the new possibility for the long document relevance task. Our approach is evaluated on two standard ad hoc-retrieval datasets from the TREC 2004 Robust Track (Robust04) and ClueWeb09 with two different characteristics individually. The major contribution of this paper include:

- We proposed a novel multi-layer contextual document pre-processing method. Firstly, we leverage text summarization extraction to capture passage-level evidence of long document, then we concatenate the extracted sentences as local contextual information to the pre-segmented passages. In addition, we take the title of the document as the global information, these components together construct the context-aware passage-level embedding.
- We utilized the Maximal Margin Relevance (MMR) algorithm to implement passage-level summarization extraction, thus giving birth to the local contextual information, compared with leveraging original text as local evidence, the prediction performance has been improved a lot.
- We provided a practical approach for the long document to be trained in the neural networks, which addresses the previously mentioned long document constraints and can be commonly used in other neural IR models.

2. Related Works

Neural networks methods have been successfully applied in producing query-document relevance scores. Without loss of generality, we divide existing approaches into two mainstream directions, including matching features methods and pre-trained language model methods. We will briefly review these researches as follows.

2.1. Matching Features Methods

Matching features method, by its name, captures diverse matching features, including exact match signals and passage-level signals.

2.1.1. Exact Match Signals

Examples for exact match signals include DSSM [10], DRMM [11], and Enhanced DRMM [12]. Guo et al. [11] first state the difference between semantic matching and relevance matching. They design a deep relevance matching model, namely DRMM, which takes the local interactions between each pair of words among query and document as input. Combining histogram mapping, a feed-forward network with a term gating network to produce the matching score. Though DRMM was shown to outperform most strong IR baselines at that time, it ignores the contextual information and the word order of each word and can not be trained end-to-end. Thus, Enhanced DRMM [12] was proposed to address the limitations which occur in DRMM. Inspired by PACRR's [13] convolutional n-gram matching features, Enhanced DRMM utilizes PACRR-like convolutional n-gram features for the context modeling which enables end-to-end training and contains the contextual information naturally. In spite of Enhanced DRMM has shown they outperform BM25-based baselines, DRMM and PACRR, they report a best AP score of 0.272, which is still far from the best-known score of 0.3686 by Cormack [14] on the dataset from the Robust Track at TREC 2004 (Robust04). They conclude there is still a great space for improvements.

2.1.2. Passage-Level Signals

Different from the most traditional retrieval methods which assess relevance based on document-level signals. In passage-level methods, documents were split into several passages in advance. Pang et al. [15] proposed DeepRank, which simulates the human judgment process by the following process: detecting relevant locations, then calculate the local relevance, finally aggregates the local relevances to generate the document-level relevance label. After BERT [7] has been estimated effective in question answering, Yang et al. [8] first apply BERT to infer sentences individually and then aggregates the sentences score to produce the document's score. The research work was followed by Yilmaz [16], they expand the former work by leveraging cross domain passage-level relevance to fine-tune BERT models to gain a better generality of the word representation. The studies mentioned above follow the same way to preprocess the document as segments individually, then aggregates the passage-level relevance as the document's score. However, the contextual information among passages were ignored.

2.2. Pre-Trained Language Model Methods

Recently, the application of pre-trained language models to IR have outperformed all previous works. Early neural IR models using word embedding like Word2Vec [6] to represent text. The word representations are learned from the surrounding context in a large corpus and customizes for search tasks. However, word co-occurrence is only a shallow bag-of-words model, the ambiguity of the natural words is unavoidable. Traditional unidirectional models usually use a left-to-right or right-to-left architecture during pre-training. The major limitation is that for every token, it can only attended to its previous words thus the contextual information cannot be intact. Devlin et al. proposed BERT [7] a deep bidirectional training model to address the issue that exists in unidirectional models mentioned above. Specifically, BERT proposed two novel unsupervised prediction tasks: MLM, which mask some tokens randomly, and the objective is to predict the masked token based on its context. Furthermore, they introduce the "next sentence prediction" task to pre-train the text pair, thus the representation trained by BERT takes the word dependencies and sentence structures into consideration. BERT has shown great success in natural language processing (NLP) task, and become a prevalent pre-trained language model in IR field. In addition to [8] we mentioned above, BERT has been widely used in ad-hoc retrieval, examples like BERTIR [9], ColBERT [16]. BERTIR leverage BERT to provide a deeper text understanding of the query and document, then combined with search knowledge to enhance its performance in related search tasks. Though BERTIR has shown prominent effectiveness in applying BERT to IR, the computational cost can not be ignored. Khattab et al. [17] proposed ColBERT which attachs late interaction to enrich the expressiveness. Simultaneously it utilizes the pre-compute document representations offline and parallelizes the rest documents online to speed up the query process.

3. Materials and Methods

3.1. Architecture

Although BERT has shown its effectiveness in IR tasks, according to [9], applying BERT to long documents leads to increasing memory usage and run time due to the complexity in interacting every pair of tokens. In this work, we propose a novel multi-layer contextual passage retrieval method based on BERT, which utilizes summary extraction to generate contextual passage-level evidence, thus provides a passage-level solution for the application of neural ranking models to query-document matching tasks. The architecture of the model is depicted in Figure 1.

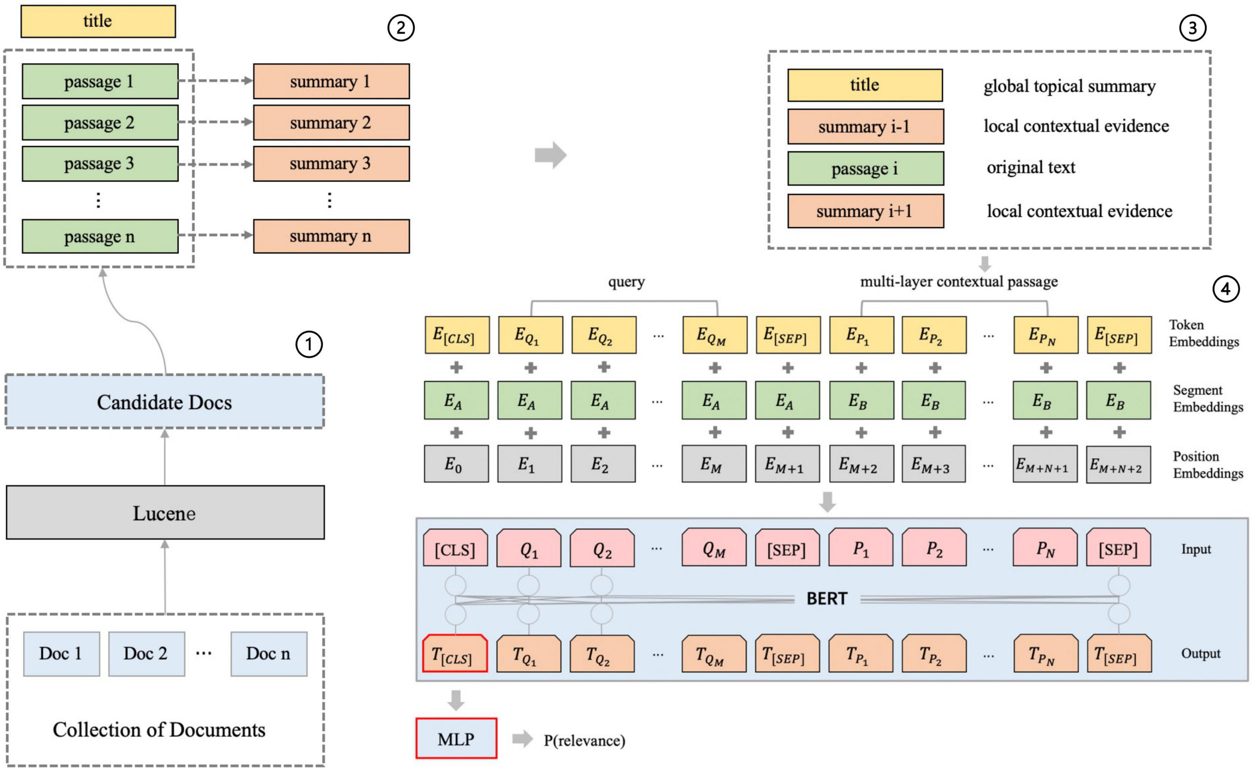
title
passage 1
passage 2
passage 3
passage n
summary 1
summary 2
summary 3
summary n
①
②
③
④
Candidate Docs
Lucene
Doc 1
Doc 2
···
Doc n
Collection of Documents
title
global topical summary
summary i-1
local contextual evidence
passage i
original text
summary i+1
local contextual evidence
query
multi-layer contextual passage
$E_{[CLS]}$ E_{Q_1} E_{Q_2} ··· E_{Q_M} $E_{[SEP]}$ E_{P_1} E_{P_2} ··· E_{P_N} $E_{[SEP]}$
Token Embeddings
E_A E_A E_A ··· E_A E_A E_B E_B ··· E_B E_B
Segment Embeddings
E_0 E_1 E_2 ··· E_M E_{M+1} E_{M+2} E_{M+3} ··· E_{M+N+1} E_{M+N+2}
Position Embeddings
[CLS] Q_1 Q_2 ··· Q_M [SEP] P_1 P_2 ··· P_N [SEP]
Input
BERT
$T_{[CLS]}$ T_{Q_1} T_{Q_2} ··· T_{Q_M} $T_{[SEP]}$ T_{P_1} T_{P_2} ··· T_{P_N} $T_{[SEP]}$
Output
MLP
P(relevance)

Figure 1. Overall model architecture. ①: Pre-segmenting and Text Preprocessing ②: Passage-level Summary Extraction ③: Generation of Multi-layer Contextual Passage ④: The Input of BERT.

As shown in the figure, the process can be divided into two steps. Firstly, for the length limitation of the documents, long documents can be pre-segmented into passages to ensure not exceed the restriction of maximum sequence length. Furthermore, the passage-level evidence is available by using text summary extraction, thus can be regarded as the local contextual information, which is attached with the title as the global topical summary, together comprising the multi-layer contextual passage. Secondly, the two text sequences of the query and the processed passage can be taken as the input of BERT. In addition, our approach uses the pre-trained word embeddings provided by BERT as the general representation of words. After the training and fine-tuning process, the output embedding of the first token [CLS] is used as the representation of the entire input textual sequence. It can be fed into multi-layer perceptron (MLP) to obtain the similarity prediction between the query and the paragraph.

3.2. Generation of Multi-Layer Contextual Passage

3.2.1. The Process of Summarization

Text summarization technology is a method that compresses long text sequences into a more concise form while ensuring that its core content can be conveyed. This paper uses the Maximum Margin Relevance (MMR) algorithm to extract text summaries and uses the TF-IDF weighting mechanism to measure the similarity between sentence vectors.

As shown in Figure 2, the summarization consists of three steps:

- **Pre-segmenting and text preprocessing.** This paper first pre-segment the long document to meet the BERT's limitation of the text sequence length of the input sequence not exceeding 512 tokens. Then the basic text preprocesses such as tokenization, word stemming and removal of stop words are performed on each passage. Stem extraction is the process of removing affixes to get roots and stop word removal is the removal of meaningless words.

- **Sentence Scoring.** In this paper, the Term Frequency–Inverse Document Frequency (TF-IDF) weight mechanism is used to evaluate the importance of terms in the document, the importance of sentences is evaluated according to the TF-IDF value of terms in the sentence, and the similarity between two sentence vectors is measured at the same time.
- **Summary Extraction.** This paper uses the Maximum Margin Relevance (MMR) algorithm to select important sentences from each passage as a passage-level summary. While ensuring that the extracted sentences and passage topics have high relevance, it eliminates the redundancy of extracted sentences and increases the diversity of results.

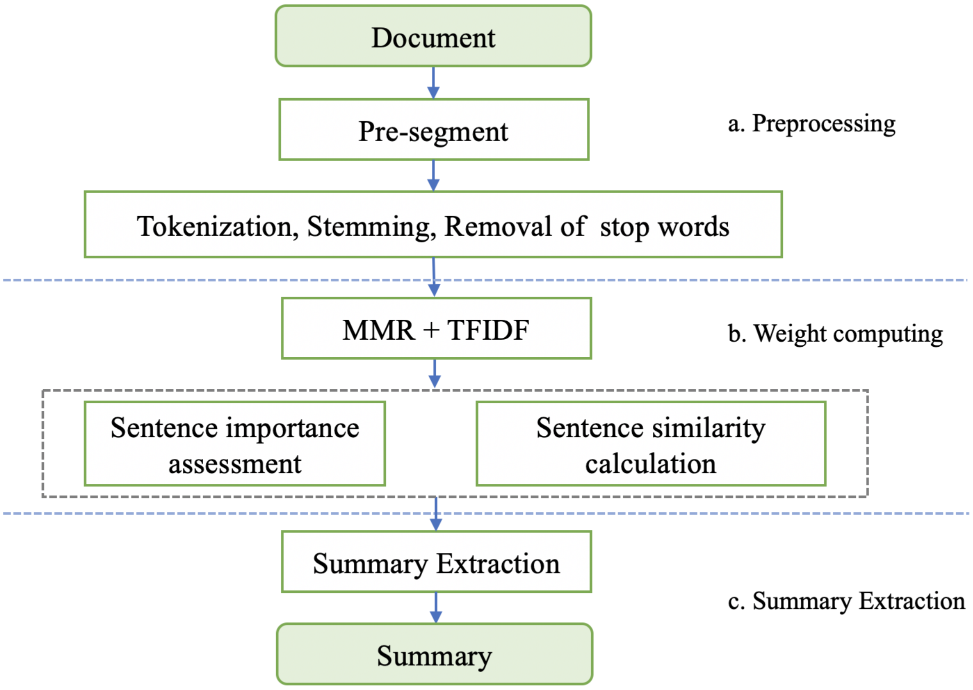

Figure 2. The process of summarization.

3.2.2. Sentence Scoring Using TF-IDF Value

Scoring sentences is a forword-step for selecting sentences within a document. In this work, we use TF-IDF weight mechanism to achieve term importance assessment, sentence scoring, and the calculation of the similarity between sentence vectors.

- **Document classification.** The ordinary TF-IDF weight mechanism does not take into account that there may be unbalanced distribution of different types of articles in the document set, which may weaken the representativeness of TF-IDF values. In order to reduce this influence, we add a process. We use clustering algorithm to process the document set, the collection of documents is processed into smaller collections of documents of several categories, and the uneven distribution of different types in the document collection is reduced after processing. TF-IDF values will be calculated separately in the classified document collection.
- **Term importance assessment.** Term Frequency (TF) is a method to evaluate the importance of a word in a passage. The importance of a term depends on the number of times the term appears in the paragraph. After basic text preprocessing, such

as removing stop words and stemming, is performed on the data, the method of calculating the TF value of the term w is as follows:

$$tf_{w,p} = \frac{n_{w,p}}{\sum_{u \in \{w_p\}} n_{u,p}} \tag{1}$$

where $n_{w,p}$ denotes the number of times a word appears in a passage and w_p denotes a collection which contains each word in passage p, thus $\sum_{u \in \{w_p\}} n_{u,p}$ represents the numbers of words in passage p. While computing TF, all terms are considered equally important, thus we need to introduce the IDF to diminish the weight of frequent words. We assume the number of passages in the collection can be denoted as n, in the meanwhile n_w denotes the number of passages which contains the word w, hence the IDF value of word w can be defined as:

$$idf_{w,p} = \log(\frac{n}{n_w}) \tag{2}$$

Hence, the IDF value of a rare word can be relatively higher than a frequent word. Finally, we can combine the calculation of term frequency and inverse document frequency to produce an aggregate weight for each term in each document. The TF-IDF weighting scheme assigns weight to word w in a certain passage p is given below:

$$tf\text{-}idf_{w,p} = tf_{w,p} \times idf_{w,p} \tag{3}$$

The TF-IDF value of all terms in a passage represents the importance of the term within the passage. Based on the ranking of TF-IDF value, we can take the top-n words constructs the query corresponding to the passage. The query will be applied to select the first sentence with the highest relevance score for the passage.

- **Sentence Scoring.** For each sentence s in the passage, we compute its TF-IDF score by summing the weights of its words:

$$Score(S) = tf\text{-}idf_s = \frac{1}{|s|} \sum_{w \in s} tf\text{-}idf(w) \tag{4}$$

where $|s|$ denotes the length of the sentence s.

- **Sentence Similarity Calculation.** The TF-IDF mechanism makes it possible to map text sentences in the vector space. The semantic similarity between two sentences can be measured by calculating the distance between the two sentence vectors. There are many ways to measure the similarity between vectors, such as cosine similarity, Euclidean distance, or the use of neural network models for judgment. This paper uses cosine similarity to evaluate the sentence similarity of two sentence vectors S_1 and S_2. The calculation method is as follows:

$$Similarity(S_1, S_2) = \frac{\sum_{i=1}^{n}(S_{1i} \times S_{2i})}{\sqrt{\sum_{i=1}^{n}(S_{1i})} \times \sqrt{\sum_{i=1}^{n}(S_{2i})}} \tag{5}$$

3.2.3. Summary Extraction with Maximal Marginal Relevance Algorithm

The summary extraction usually needs to meet two requirements: on the one hand, the extracted content needs to be highly related to the original passage; on the other hand, it is necessary to eliminate information redundancy as much as possible and increase the diversity of summary. In order to satisfy these two requirements, this paper uses the MMR algorithm to extract sentences from passages, the process in detail is shown in Figure 3.

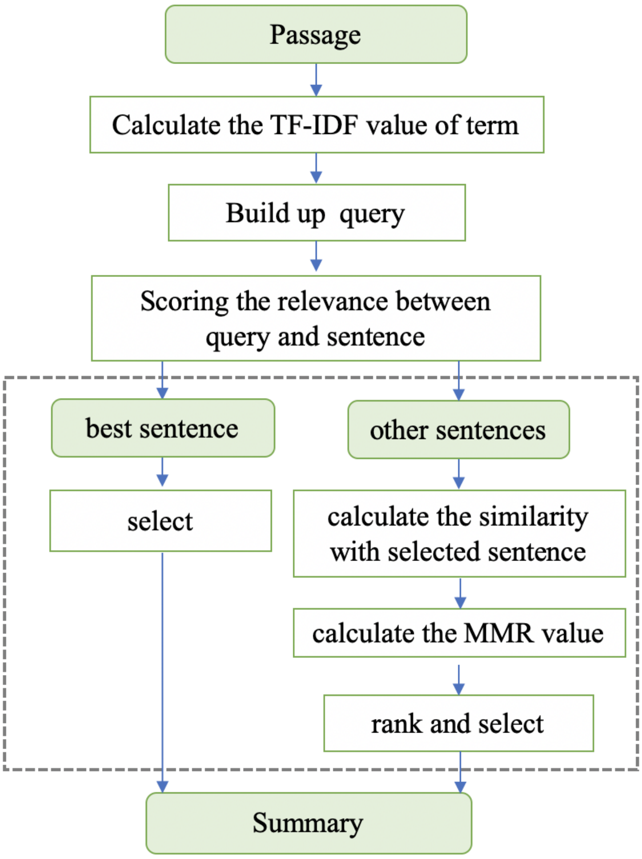
Passage
Calculate the TF-IDF value of term
Build up query
Scoring the relevance between query and sentence
best sentence
other sentences
select
calculate the similarity with selected sentence
calculate the MMR value
rank and select
Summary

Figure 3. The process of summary extraction.

The Maximal Marginal Relevance (MMR) algorithm is used to reduce the redundancy meanwhile retaining relevance to the query when extracting sentence. We calculate the similarity among query and sentences to remain the relevance, then we calculate the similarity between sentences within a document and remove the similar sentences to diminish the redundancy. The summary is constructed from a list of ranked sentences, at each iteration, the algorithm works as follows:

$$MMR = \arg\max_{s_i \in S}[\lambda sim(Q, s_i) - (1-\lambda)\max_{s_j \in C} sim(s_i, s_j)] \tag{6}$$

where Q denotes query, S denotes the set of candidate sentences within a document and C denotes a collection of selected sentences which was extracted to form a summary. Hence the former item in the square brackets $sim(Q, s_i)$ represents the similarity of a certain sentence s_i to the query, the later one $sim(s_i, s_j)$ measures the similarity between s_i and s_j. λ values between 0 to 1, where 0 indicates maximum relevance on the contrary 1 indicates maximum diversity.

Assume that for each passage the goal is to select n sentences, the selection process is shown in Algorithm 1.

Algorithm 1 MMR Algorithm

```
Input: query $Q$, the collection of documents $D$
Output: the collection of selected sentences $summary$
Initialize: summary = ∅
 for each doc in $D$ do
   $passages = gen_passages(doc)$
   for each passage in $passages$ do
     $sents = preprowords_andwordFreq(passage)$
     $bestSentence = best_sentence(sents, query, IDF)$
     for each sentence $s_i$ in $sents$ do
       $updatevariablesummary = [bestSentence]$
       $removebestSentencefromsents$
       $Sim1 = sentenceSim(Q, s_i, IDF)$
       for each sentence $s_j$ in $sents$ do
         $Sim2 = sentenceSim(s_i, s_j, IDF)$
         $MMRScore(sent) = \arg\max[\lambda Sim1 - (1 - \lambda) \max Sim2]$
         $maxxer = max(MMRScore)$
         $summary.append(maxxer)$
       end for
     end for
   end for
 end for
 return $C$
```

3.2.4. Generation of Multi-Layer Contextual Passage

According to the human reading experience, the understanding of a single passage consisted of the following three types of information:

- **The document's title.** The title of a document describes briefly the central subject of a document, which can be regarded as a global information definition of a document.
- **The context clue.** The context clues are hints that help a reader to understand the meanings of new or unfamiliar passages. It can be regarded as a local evidence which is beneficial for achieving a better reading comprehension of a paragraph.
- **The passage body.** The methods mentioned above are just auxiliary strategies to help us better leverage the contextual information, eventually, we have to focus on the passage itself to dig intthe topic.

3.3. BERT-Based Query Document Relevance Retrieval with Multi-Layer Contextual Passage

This work utilizes BERT architecture as our base model which was pre-trained on a large corpus and can be fine-tuned for the downstream task, the detail described by Devlin et al. [7]. It is pre-trained with two unsupervised tasks namely MLM and Next Sentence Prediction which jointly pre-trains text pair representation. Multiple text segments can be encoded using two special tokens ([CLS] and [SEP]), while [CLS] is a special token used for classification task, and [SEP] token separates two segments. In this work, the input representation of BERT is shown in Figure 4.

From the above figure, the input of BERT is constructed of the summing of the following three parts:

- **Token embedding** Let us consider a query $Q = \{Q_1, Q_2, ..., Q_M\}$ and a passage $P = \{P_1, P_2, ..., P_N\}$ which consists of the concatenated multi-layer contextual structure.The model takes the query Q and the passage P as the segment pairs to construct the input sequence of BERT : $S = [[CLS], Q, [SEP], P, [SEP]]$, the tokens of the sequence are embedded into embeddings.
- **Segment embedding** The query and passage pairs are separated with the [SEP] token, we add a query Q embedding to every token in the query and a passage P embedding to every token in the passage.

- **Position embedding** We use the positional embeddings to enable BERT to capture the word order of the input representation which contributes to the ability of learning the sequential characters.

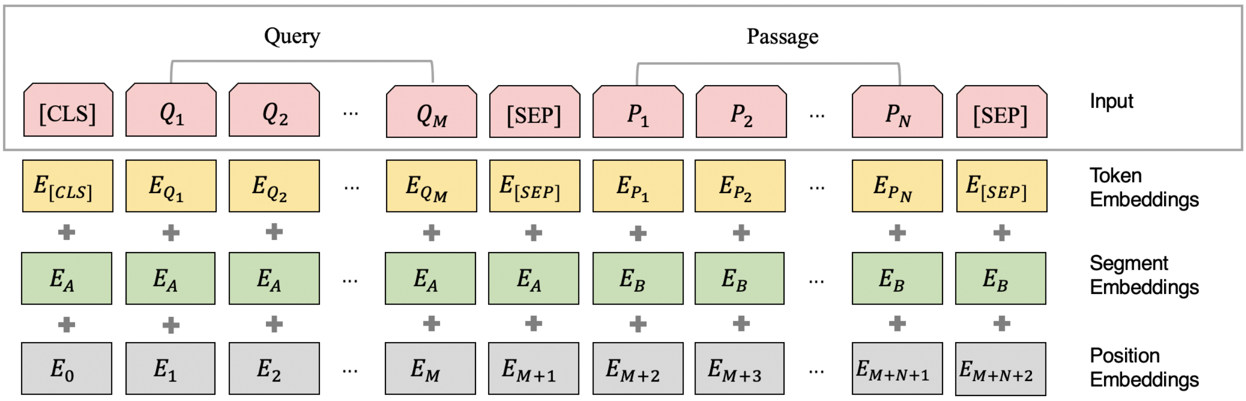

Figure 4. BERT input representation.

After processing the input of the model, this paper utilizes the sentence pair classification task to obtain the relevance score between each passage and query. The relevance evaluation process of query paragraphs is shown in Figure 5.

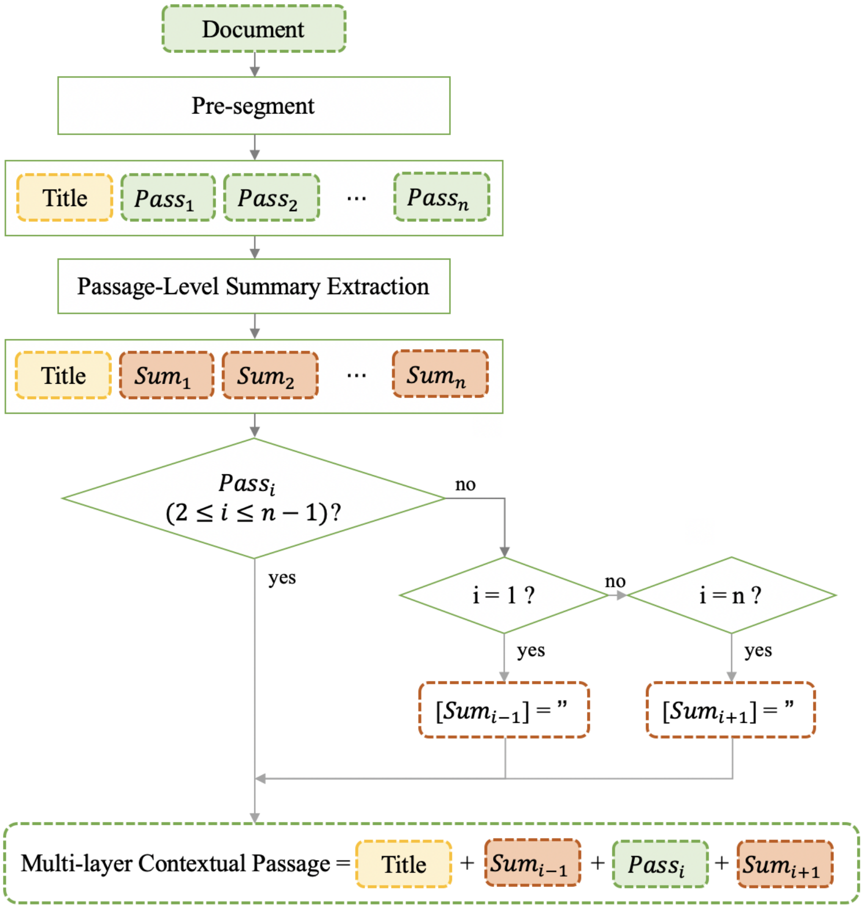
Document
Pre-segment
Title
$Pass_1$
$Pass_2$
$Pass_n$
Passage-Level Summary Extraction
Title
Sum_1
Sum_2
Sum_n
$Pass_i$ $(2 \leq i \leq n-1)$?
no
yes
i = 1 ?
no
i = n ?
yes
yes
$[Sum_{i-1}] = ''$
$[Sum_{i+1}] = ''$
Multi-layer Contextual Passage = Title + Sum_{i-1} + $Pass_i$ + Sum_{i+1}

Figure 5. Multi-level context paragraph structure generation flowchart.

As shown in the figure above, the final hidden state C_i of the category character [CLS] usually plays two roles: (1) it can be used as the embedded representation of the input text sequence; (2) C_i can be fed into Multi Layer Perceptron (MLP) and with softmax as the activation function, the final classification result can be used to calculate the relevance score of query and passage.

In this paper, we propose a novel approach to simulate the human reading strategies mentioned above, the generation process is shown in Figure 6.

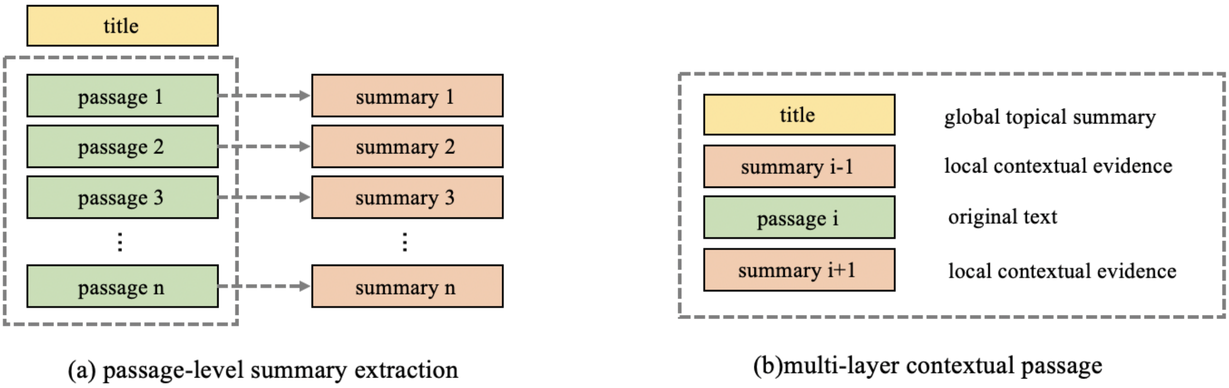

Figure 6. Generation of multi-layer contextual passage.

As shown in Figure 6, we firstly split a document into several independent passages, when the document title is available, the title was considered as the global topical summary added to every single passage. In addition, a textual summary extraction model is applied to generate the local clues by combining TF-IDF and Maximal Marginal Relevance (MMR). Eventually, the raw text of passage body attached with the contextual information was taken as the input of a neural language model to produce a query-passage matching score to further predict a query-document relevance score. Specifically, For passage i, how to construct its multi-layer contextual passage is shown in Figure 7.

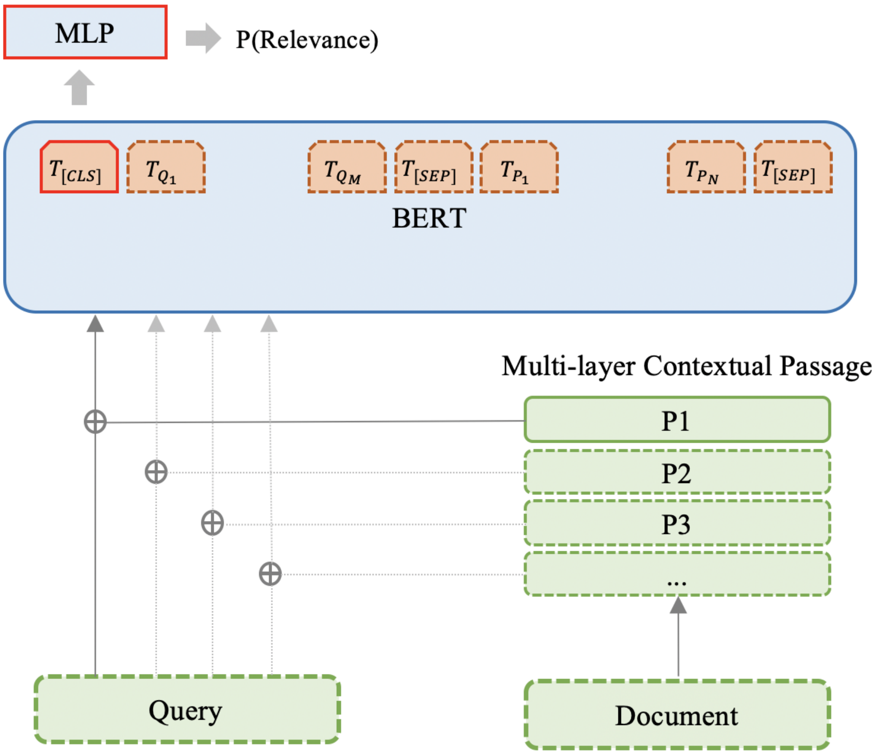

Figure 7. Prediction by BERT.

Assuming that a document can be divided into n passages, for a certain passage $pass_i$, when i in a range from 2 to $n-1$, the multi-layer contextual passage of $pass_i$ is the concatenation of the title, the summary of $i-1$th passage, original text of $pass_i$ as the summary of $i+1$th passage. In the other case, if i equals 1, which means the passage is the first passage within the document, hence its Sum_{i-1} is an empty string. In the same way, if i equals $n-1$, which means the passage is the last passage of the document, and its Sum_{i+1} is an empty string either. An example of text content is shown in Table 1.

Table 1. Example of Concatenated Multi-layer Contextual Passage (docid: clueweb09-en008-57-21952 passid:passage-2).

id	clueweb09-en0008-57-21952
position	2
Title	Computer Keyboards Reviews, Buy Wireless Computer Keyboards, Best Deals of Keyboards
Sum_{i-1}	innovative approach to keyboard design, the OLED-based Optimus Maximus keyboard is best considered an expensive novelty. Its $1600 price tag keeps it out of the hands of the average consumer, and we also question the
$Pass_i$	practical benefit of using 113 customizable OLED screens as an input device. There is something undoubtedly unique and appealing about the degree to which the Optimus Maximus gives you complete control over its keys appearance. However, even for gamers, designers, and others who tend to demand more from their input hardware, the Optimus Maximus offers insufficient utility to justify its high price. Read Optimus Maximus: Reviews, Deals, Specifications, Videos, and Prices Add Comments Solar Powered Computer Keyboard and Mouse Read
Sum_{i+1}	More Weird Computer Keyboards, Wireless Keyboard Tired of changing batteries of your wireless keyboard and mouse? Probably it is the time to change. If you ask me what is the best option, I would definitely suggest this KYE Systems Slim Star

4. Results

In this section, we conduct experiments on two TREC collections and utilize the standard TREC evaluation tool to demonstrate the effectiveness of our proposed model.

4.1. Datasets

We demonstrate the efficiency of our approach on two widely-used ad hoc retrieval datasets, they are the news corpus Robust04 which is provided by TREC (Text Retrieval Conference), and the webpage corpus Clueweb09.

Robust04 is a news corpus provided by 2004 Robust track [7] for the reason to improve the performance of retrieval. It contains 249 queries and 528,155 documents which are formed as a ranked list corresponding to the queries. We conduct our experiment with two kinds of queries: (1) **Title.** a version of short keyword query text. (2) **Description.** a type of long descriptive natural language which can be regarded as the expansion of the title. The query text sample in Robust04 is shown in Table 2. This section may be divided by subheadings. It should provide a concise and precise description of the experimental results, their interpretation as well as the experimental conclusions that can be drawn.

Table 2. Example of query text in Robust04 (Topic 693).

Title	**Newspapers Electronic Media**
Description	What has been the effect of the electronic media on the newspaper industry?
Narrative	Relevant documents must explicitly attribute effects to the electronic media: information about declining readership is irrelevant unless it attributes the cause to the electronic media.

ClueWeb09-B is a webpage corpus provided by Carnegie Mellon University to support Information Retrieval and Natural Language Processing research. The dataset contains a web crawl of 50 million English pages and 200 queries with title and description. The query text in ClueWeb09 can be divided into three parts: title, description, and subtopics. **Title** is a condensed representation of the query statement composed of keywords, and **Description** is the complete description of the query statement. **Subtopics** usually involves multiple topics related to the query. Specific example of query text in ClueWeb is shown in Table 3.

Table 3. Example of query text in ClueWeb09 (Topic 27).

Title	**Starbucks**
Description	Find information about the coffee company Starbucks.
Narrative	[“Take me to the Starbucks homepage.”, “What is the balance on my Starbucks gift card?”, “Find the menu from Starbucks, with prices.”, “Find calorie counts and other nutritional information about Starbucks products.”, “Find recipes from Starbucks, either for making or using Starbucks products.”, “I’m looking for locations of Starbucks stores worldwide.”]

4.2. Baselines

We adopt three types of baseline methods for comparison, including traditional information retrieval baselines and early neural ranking model baselines and BERT-based baselines.

4.2.1. Traditional Retrieval Baselines

For traditional information retrieval, we compare two classical methods, includes QL (query likelihood) model [17] and BM25 model [1].

QL [17]: Query likelihood model is an information retrieval model based on language modeling using Dirichlet smoothing method. The model utilizes maximum likelihood estimation (MLE) and unigram language model as pre-hypotheses, for calculating the probability of generating a query given a document language model. QL performs best in retrieval models based on language modeling.

BM25 [1]: The BM25 approach is the representative of probabilistic retrieval models and one of the most effective models among traditional information retrieval models. This paper uses the open source tool Anserini and Parrot algorithm as the implementation of the algorithm.

4.2.2. Early Neural Ranking Baselines

For early neural ranking models, we compare with strong baselines including DSSM_D [17] and DRMM [18].

DSSM_D [18]: DSSM_D is the state-of-the-art deep matching model used in web search tasks. We evaluate the model based on <query, document> pairs where the document is the full text from the datasets, we denote this model as DSSM_D.

DRMM [11]: DRMM model is a deep relevance matching model, they first propose that the ad-hoc retrieval task can be formalized as a relevance matching problem between the two text sequences of query and document. The paper introduced different types of histogram mapping functions and term gating functions, we chose the best performed one DRMM (LCH+IDF) as our baseline for comparison.

4.2.3. BERT-Based Retrieval Baselines

In order to verify the accuracy of the query document based on the multi-level paragraph structure, we also compare with other BERT-based neural ranking models namely BERTIR.

BERTIR-MaxP/BERTIR-SumP [7]: BERTIR is a method that takes advantage of BERT to enhance the language understanding of query and document. The BERTIR model uses a sliding window method to predict the relevance score between query and passages. In the passage-level training method, the document score will be denoted as BERTIR-MaxP and BERTIR-SumP based on the best passage score or the sum of all passages within a document.

4.3. Experimental Setups

This work utilizes an off-the-shelf BERT$_{Base}$ architecture from Devlin at [2] with the configuration file and vocabularies provided by the official. The BERT$_{Base}$ model contains 12 layers of Transformer, 768 hidden state units, and the number of heads based on the multi-head attention mechanism is 12, the total number of parameters in the model reaches 110 M. In addition, this article uses the uncased type to ignore the the letters case.

We finetune the model on the datasets described above and set the maximum sequence length of 256 tokens. We train our model using cross-entropy loss function for every epoch with a batch size of 16 and then predict with the batch size of 32. We assign the initial learning rate for Adam of 1×10^{-5} to optimize and set the warmup proportion of 0.1 during the training process to perform linear learning.

This paper adopts the 5-fold cross validation method to verify the performance of our method. Firstly, for each dataset, we divided it into five parts. Then, we choose four folds are used to train and fine-tune the BERT model, and then the model predicts the relevance score on the remaining fold data, finally, we repeated the appeal step five times, and each time we selected a different folds data as training data. We use the official TREC-eval evaluation toolkit to calculate the MAP value and nDCG value of each fold, the final result is the average value of all folds.

4.4. Performance Comparison

In order to verify the effectiveness of our method in this paper, this subsection will specify the performance results of the model on the two data sets and compare them with related research work.

4.4.1. Comparison on Robust04

The Robust04 dataset is a news corpus which provided by TREC and has been a standard dataset supporting information retrieval research.

This article compares three types of strong benchmark models for information retrieval, namely: (1) representatives of traditional information retrieval models, including query likelihood models (QL) and BM25 Model; (2) strong baseline of early neural ranking models such as DSSM and DRMM; (3) BERT-based models like BERTIR which enhance retrieval performance by pre-training and fine-tuning BERT. The comparison result is as follows:

As shown in Table 4, the model we proposed has outperformed traditional information retrieval models (QL and BM25), the state-of-the-art neural networks in recent years (DSSM and DRMM) and other BERT-based models (BERTIR-MaxP and BERTIR-SumP) on Robust04 dataset.

Table 4. Comparison on Robust04.

	Title		**Description**	
Model	**nDCG@20**	**P@20**	**nDCG@20**	**P@20**
QL	0.415	0.369	0.391	0.334
BM25	0.418	0.370	0.399	0.337
DSSM$_D$	0.201	0.171	0.169	0.145
DRMM (LCH+IDF)	0.431	0.382	0.437	0.371
BERTIR-MaxP	0.469	0.408	0.529	0.439
BERTIR-SumP	0.467	0.402	0.524	0.443
OurMethod-MaxP	**0.489**	**0.419**	**0.537**	0.451

In terms of specific evaluation metrics, the performance of the DRMM model far exceeds that of DSSM and traditional information retrieval models. Compared with DRMM, using title queries brings a 13.45% improvement on nDCG@20 metric, and a 9.68% improvement on P@20 metric; using the description queries brings a 22.88% improvement and 21.56% improvement on nDCG@20 and P@20, respectively.

The main reason is that the DRMM model uses pre-trained Word2Vec as the word embedding representation, which is far less effective than the BERT-based information retrieval model. The experimental results also further illustrate that the contextual word embeddings is more effective than the bag-of-words model in information retrieval tasks.

In addition, compared to the previous best-performing model BERTIR-MaxP, which is also based on BERT. Compared with BERTIR-MaxP, using title queries brings a 4.26% improvement on nDCG@20 metric and a 2.70% improvement on P@20 metric. While using description queries, this paper brings a 1.51% improvement on nDCG@20 and 2.73% improvement on P@20. The comparison of the experimental results also shows that the summary generated by the summary extraction can provide more comprehensive contextual passage information than the original text in the sliding window mode. The final accuracy improvement also shows the effectiveness of multi-layer contextual passages.

4.4.2. Comparison on ClueWeb09

ClueWeb09-B is a large-scale web data set, the themes of which are from TREC Web Tracks 2009, 2010, and 2011. The data set contains 200 queries and 50 million web pages, and the vocabulary has a scale of 38 million. This paper chooses strong baseline models that have performed well on this dataset in recent years, which can also be divided into three categories: traditional information retrieval models, neural ranking models, and BERT-based information retrieval models. These models have been introduced on the Robust04 dataset, so they will not be described again. The comparison results are shown in Table 5.

From the above table, we can figure out that the method we proposed has outperformed the other neural ranking models by using title queries. In addition, on the description queries, the results are similar to the best description query baseline (BERTIR-SumP). The specific evaluation metrics can be viewed from the Title field and the Description field, respectively:

- Title. In the Title field of the ClueWeb09 Cat B dataset, the MAP value of the model method in this paper reaches 0.189 and the value of nDCG@20 reaches 0.327, which exceeds all previous related work in this field. With the help of multi-layer contextual passage, our method brings a 12.37% improvement on nDCG@20 and a 16.67% improvement on MAP, respectively, over the best performed baseline (BERTIR-SumP). The comparison further proves the effectiveness of our method.
- Description. In the Description field of the ClueWeb09 Cat B dataset, the MAP value of the model method in this paper reaches 0.145 and the nDCG@20 value reaches 0.255, which is basically the same as the previous best model BERTIR-SumP retrieval effect and exceeds all other previous benchmark models.

Table 5. Comparison on Clueweb09-B.

	Title		**Description**	
Model	**nDCG@20**	**MAP**	**nDCG@20**	**MAP**
QL	0.224	0.100	0.283	0.075
BM25	0.225	0.101	0.196	0.080
DSSM$_D$	0.099	0.039	0.078	0.034
DRMM (LCH+IDF)	0.258	0.113	0.227	0.087
BERTIR-MaxP	0.287	0.161	0.261	0.144
BERTIR-SumP	0.291	0.162	0.266	0.147
OurMethod	**0.327**	**0.189**	**0.260**	0.148

5. Discussion

In the Description field of the ClueWeb09 Cat B data set, the retrieval effect is not significantly improved compared to BERTIR-SumP, We have analyzed a variety of possible factors.

The main reasons are as follows: on the one hand, ClueWeb09 is a web page dataset, and its text content contains a lot of web page data information, such as tables, navigation bars, and other discontinuous texts. Therefore, the extracted passage level summary is affected by the quality of the dataset. Hence it cannot accurately express the subject of the contextual paragraph. Therefore, compared with BERTIR-SumP, which uses the original text in the sliding window as the context, the contextual clues are not significant. On the other hand, the Description field is a longer natural language text description than the Title field, so it contains more information than the Title field, which dilutes the weights of the really important keywords in the query text. It is more difficult to predict query document relevance accurately on webpage datasets.

6. Conclusions

This paper has carried out research work on how to build relevance matching between long documents and query. In view of the limitation of the neural ranking model applied to long document tasks, this paper proposes a multi-layer contextual passage structure based on BERT, which mainly utilizes summarization to provide better cobtextual information. We conduct our experiment on two standard information retrieval datasets, the experimental results verify that there are several advantages in our method: (1) This work is based on pre-training and fine-tuning BERT, which can bring a large improvement compared with the traditional neural ranking model that uses word2vec as word embedding. BERT can take syntactic structure and word dependency into account, hence it can better capture contextual semantics, thereby improving the accuracy of information retrieval tasks. (2) This article creatively proposes to use text summarization extraction technology to generate passage-level evidence. On the one hand, this method transforms the query document relevance matching task into relevance matching between query and passages, which solves the problem of the maximum input sequence length when the neural ranking model is applied to long documents. On the other hand, the sentences extracted by the text summary extraction technology can be regarded as a refined representation of the contextual passages. Compared with the sliding window method, our approach can provide more concise and accurate contextual information. Although the model method proposed in this paper has outstanding performance in the task of query document relevance matching, there are still some expandable directions for subsequent research work. Such as this paper utilize the combination of TF-IDF and MMR algorithm to select important sentence within a document, the sequence work can use the other summary generation methods to provide a summary with higher quality.

Author Contributions: Conceptualization, Y.L.; methodology, Y.L.; software, Z.H. and D.L.; validation, Y.L., D.L. and Z.H.; formal analysis, Y.L.; investigation, Y.L. and Z.H.; resources, W.C.; data curation, Y.L. and Z.H; writing—original draft preparation, Y.L.; writing—review and editing, Z.H. and Y.F.; visualization, J.C.; supervision, W.C.; project administration, W.C.; funding acquisition, W.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Science and Technology Planning Project of Guang dong Province, grant number 2016B010124012, 2019B010116001.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: We would like to thank the authors of all datasets used in this paper for making the data available to the community.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Robertson, S.; Zaragoza, H.; Taylor, M. Simple BM25 extension to multiple weighted fields. In Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management, Washington, DC, USA, 8–13 November 2004; pp. 42–49.
2. Rousseau, F.; Vazirgiannis, M. Composition of TF normalizations: New insights on scoring functions for ad hoc IR. In Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval, Dublin, Ireland, 28 July–1 August 2013; pp. 917–920.
3. Lv, Y.; Zhai, C.X. When documents are very long, BM25 fails! In Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, Beijing, China, 24–28 July 2011; pp. 1103–1104.
4. Jian, F.; Huang, J.X.; Zhao, J. A simple enhancement for ad-hoc information retrieval via topic modelling. In Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval, Pisa, Italy, 17–21 July 2016; pp. 733–736.
5. Pan, M.; Zhang, Y.; Zhu, Q. An adaptive term proximity based rocchio's model for clinical decision support retrieval. *BMC Med. Inform. Decis. Mak.* **2019**, *19*, 251. [CrossRef] [PubMed]
6. Mikolov, T.; Sutskever, I.; Chen, K. Distributed representations of words and phrases and their compositionality. In Proceedings of the Conference on Neural Information Processing Systems, Lake Tahoe, NV/CA, USA, 5–10 December 2013.
7. Devlin, J.; Chang, M.W.; Lee, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv* **2018**, arXiv:1810.04805.
8. Yang, W.; Zhang, H.; Lin, J. Simple applications of BERT for ad hoc document retrieval. *arXiv* **2019**, arXiv:1903.10972.
9. Dai, Z.; Callan, J. Deeper text understanding for ir with contextual neural language modeling. In Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, Paris, France, 21–25 July 2019; pp. 985–988.
10. Huang, P.S.; He, X.; Gao, J. Learning deep structured semantic models for web search using clickthrough data. In Proceedings of the 22nd ACM International Conference on Information and Knowledge Management, San Francisco, CA, USA, 27 October–1 November 2013; pp. 2333–2338.
11. Guo, J.; Fan, Y.; Ai, Q. A Deep Relevance Matching Model for Ad-hoc Retrieval. In Proceedings of the Conference on Information and Knowledge Management, Venice, Italy, 24–28 April 2016; pp. 55–64.
12. Mcdonald, R.; Brokos, G.I.; Androutsopoulos, I. Deep Relevance Ranking Using Enhanced Document-Query Interactions. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, 31 October–4 November 2018.
13. Hui, K.; Yates, A.; Berberich, K.; Melo, G.D. PACRR: A Position-Aware Neural IR Model for Relevance Matching. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, Copenhagen, Denmark, 7–11 September 2017.
14. Cormack, G.; Clarke, C.; Büttcher, S. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval, Boston, MA, USA, 19–23 July 2009.
15. Pang, L.; Lan, Y.; Guo, J.; Xu, J.; Xu, J.; Cheng, X. DeepRank: A New Deep Architecture for Relevance Ranking in Information Retrieval. In Proceedings of the CIKM, Singapore, 6–10 November 2017.
16. Yilmaz, Z.A.; Yang, W.; Zhang, H.; Lin, J. Cross-Domain Modeling of Sentence-Level Evidence for Document Retrieval. In Proceedings of the EMNLP/IJCNLP, Hong Kong, China, 3–7 November 2019.
17. Zhai, C.; Lafferty, J. A study of smoothing methods for language models applied to Ad Hoc information retrieval. In Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval—SIGIR'01, New Orleans, LA, USA, 9–12 September 2001.
18. Khattab, O.; Zaharia, M. ColBERT: Efficient and Effective Passage Search via Contextualized Late Interaction over BERT. *arXiv* **2020**, arXiv:2004.12832.