B-PROP: Bootstrapped Pre-training with Representative Words Prediction for Ad-hoc Retrieval

Xinyu Ma, Jiafeng Guo, Ruqing Zhang, Yixing Fan, Yingyan Li and Xueqi Cheng
CAS Key Lab of Network Data Science and Technology, Institute of Computing Technology,
Chinese Academy of Sciences, Beijing, China
University of Chinese Academy of Sciences, Beijing, China
{maxinyu17g,guojiafeng,zhangruqing,fanxdixing,liyingyan19s,cxq}@ict.ac.cn

ABSTRACT

Pre-training and fine-tuning have achieved remarkable success in many downstream natural language processing (NLP) tasks. Recently, pre-training methods tailored for information retrieval (IR) have also been explored, and the latest success is the PROP method which has reached new SOTA on a variety of ad-hoc retrieval benchmarks. The basic idea of PROP is to construct the representative words prediction (ROP) task for pre-training inspired by the query likelihood model. Despite its exciting performance, the effectiveness of PROP might be bounded by the classical unigram language model adopted in the ROP task construction process. To tackle this problem, we propose a bootstrapped pre-training method (namely B-PROP) based on BERT for ad-hoc retrieval. The key idea is to use the powerful contextual language model BERT to replace the classical unigram language model for the ROP task construction, and re-train BERT itself towards the tailored objective for IR. Specifically, we introduce a novel contrastive method, inspired by the divergence-from-randomness idea, to leverage BERT’s self-attention mechanism to sample representative words from the document. By further fine-tuning on downstream ad-hoc retrieval tasks, our method achieves significant improvements over PROP and other baselines, and further pushes forward the SOTA on a variety of ad-hoc retrieval tasks.

CCS CONCEPTS

• Information systems → Information retrieval.

KEYWORDS

Bootstrapped Pre-training; Pre-trained Language Model; Ad-hoc Retrieval

1 INTRODUCTION

Recently, we have witnessed the bloom of the ‘pre-training and fine-tuning’ paradigm in the natural language processing (NLP) field [4, 12, 29, 38], which firstly learns universal language representations from large-scale unlabeled text corpora and then fine-tunes the pre-trained models on downstream tasks. The success of pre-trained models in the NLP field has also attracted much attention in the IR community. Researchers have applied the popular models, e.g., ELMo [27], BERT [12] and T5 [29], for ad-hoc document ranking, and shown that they can also benefit the search tasks especially when the training data are limited [10, 22–24, 35].

Beyond the direct application of existing pre-trained language models, there have been some pioneer studies on the pre-training objectives tailored for IR [6, 21]. The underlying belief is that using a pre-training objective that more closely resembles the downstream task could lead to better fine-tuning performance [14, 16, 38]. A latest practice in this direction is the pre-training method with representative words prediction (PROP) [21], which has reached new SOTA on a variety of ad-hoc retrieval benchmarks [7, 8, 28] as well as the top performance on the MS MARCO Document Ranking Leaderboard1. The key idea of PROP is to construct the representative words prediction (ROP) task for pre-training inspired by the query likelihood (QL) model [37]. Given an input document, a pair of word sets are sampled according to the document language model, and the word set with higher likelihood is deemed as more representative to the document. A Transformer model is then pre-trained to predict the pairwise preference between the two sets of words, jointly with the Masked Language Model (MLM) objective. As we can see, the success of PROP heavily relies on the quality of the “representative” words sampled from a document. In their work [21], the ROP task is constructed based on the classical multinomial unigram language model with Dirichlet prior smoothing [37]. Although the unigram language model is a popular and strong language model in IR, its limitation is also clear. The term independence assumption adopted by the language model makes it difficult to fully capture the document semantic by ignoring the correlation between words. Consequently, this may affect the ROP sampling process, which is supposed to find representative words with respect to the document semantic.

To tackle this problem, in this paper, we propose to Bootstrapped Pre-training with Representative wOrds Prediction (B-PROP) based on BERT [12] for ad-hoc retrieval. As we know, BERT is a contextual language model learned from large-scale unlabeled text corpora. With a stack of multi-head self-attention architecture, each token in BERT accumulates the information from both left and right

---

1https://microsoft.github.io/msmarco/
Table 1: Words sampled from MS MARCO document D360225 using different BERT-based language models. Darker color indicates higher attention. Green color denotes words that are representative for the document while red color denotes the common words.

| Title: pulmonary fibrosis |
|---------------------------|

Pulmonary fibrosis Synonyms Interstitial pulmonary fibrosis A chest X-ray demonstrating pulmonary fibrosis believed to be due to amiodarone. Speciality Pulmonology Pulmonary fibrosis (literally "scarring of the lungs") is a respiratory disease in which scars are formed in the lung tissues, leading to serious breathing problems. Scar formation, the accumulation of excess fibrous connective tissue (the process called fibrosis), leads to thickening of the walls, and causes reduced oxygen supply in the blood. As a consequence patients suffer from perpetual shortness of breath. [1] In some patients the specific cause of the disease can be diagnosed, but in others the probable cause cannot be determined, a condition called idiopathic pulmonary fibrosis. There is no known cure for the scars and damage in the lung due to pulmonary fibrosis. [2] Contents [ hide ] 1 Signs and symptoms 2 Cause 3 Pathogenesis 4 Diagnosis 5 Treatment 6 Prognosis 7 Epidemiology 8 References 9 External links 10 Symptoms of pulmonary fibrosis are mainly: [3]...

| Vanilla Attention-based Term Distribution: pulmonary fibrosis interstitial ... |
|-----------------------------|
| Contrastive Term Distribution: fibrosis pulmonary interstitial idiopathic lung chest disease causes patients x-ray scars ... |

context to enrich its representation and the special classification token [CLS] is an aggregate of the entire sequence representation, which represents the comprehensive understanding of the input sequence [12]. The BERT representation has shown its superior performance over a large variety of language understanding tasks [18, 30, 34]. Therefore, the key idea of B-PROP is to leverage the powerful contextual language model BERT to replace the classical unigram language model for the ROP task construction, and re-train BERT itself towards the tailored objective for IR.

However, it is non-trivial to apply BERT for the ROP task construction. At first thought, an intuitive solution is to directly sample representative words according to BERT’s [CLS]-Token attention, which is equivalent to sampling from a term distribution by summing up and re-normalizing the vanilla attention weights over distinct terms. Unfortunately, as shown in Table 1, we find that although the vanilla [CLS]-Token attention-based term distribution could generate representative words, it also favors common words, e.g., in, the, of and so on. This is reasonable for BERT since these common words appear frequently and are useful in composing a document, but they are not representative to the document in retrieval. In other words, BERT attempts to encode all the semantic information in a document, so the term distribution obtained from its vanilla attention is a semantic distribution, but not necessarily a representative/informative distribution.

Inspired by the divergence-from-randomness idea [1], we propose a novel contrastive method to leverage BERT’s attention mechanism to sample representative words from the document. The basic idea of divergence-from-randomness says that the informativeness of a term could be computed by measuring the divergence between a term distribution produced by a random process and the actual term distribution in a document. As mentioned above, we can compute the actual term distribution in a document based on BERT’s vanilla [CLS]-Token attention. To approximate the term distribution produced by a random process, we take an expectation over all the term distributions in the collection. We then compute the cross-entropy (i.e., the divergence) between the document term distribution and the random term distribution to obtain the contrastive term distribution. Sampling from such contrastive term distribution, we can now obtain representative words for a document as shown in Table 1. Therefore, in B-PROP, we adopt this contrastive method for sampling to construct the ROP task, and re-train BERT with respect to the PROP learning objective.

We pre-train B-PROP over two kinds of large-scale text corpus respectively following the experimental setting in PROP. One is the English Wikipedia which contains tens of millions of well-formed wiki-articles, and the other is the MS MARCO Document Corpus which contains about 4 million Web documents. We then fine-tune B-PROP on 7 representative downstream ad-hoc retrieval datasets, including Robust04, ClueWeb09-B, Gov2, MQ2007, MQ2008, MS MARCO document ranking task, and TREC 2019 Deep Learning (DL) Track benchmark. The experimental results demonstrated that B-PROP can achieve significant improvements over baselines without pre-training or with other pre-training methods, and further push forward the SOTA on a variety of ad-hoc retrieval tasks.

2 BACKGROUND

Our method adopts the same pre-training task of PROP [21], so we would like to first briefly review this method in this section. The key idea of PROP is inspired by the traditional statistical language model for IR, specifically the query likelihood model [37]. The query likelihood model assumes that the query is generated as a piece of text representative of the “ideal” document. Based on this assumption, the representative words prediction (ROP) task is introduced for pre-training for IR as follows.

Representative Word Sets Sampling. Given an input document, a pair of word sets are sampled according to the document language model. Specifically, to simulate the varied query length in practice [2, 3], PROP first uses a Poisson distribution to sample a positive integer as the size of the word set. Then a pair of word sets with the same size are sampled in parallel from the corpus vocabulary according to the document language model, i.e., a multinomial unigram language model with Dirichlet prior smoothing [37].

Representative Words Prediction (ROP). Given the pair of word sets sampled above, the likelihood of each set is computed according to the query likelihood model and the word set with higher likelihood is deemed as more “representative” of the document. A Transformer model is then pre-trained to predict the pairwise preference between the two sets of words (i.e., ROP).

Specifically, the word set $s$ and the document $d$ are concatenated with special delimiter tokens as a single input sequence, i.e.,
Session 6D: IR Models

Algorithm 1: Representative Word Sets Sampling

Input: Document Collection $D$, Vocabulary $V = \{w_k\}_{k=1}^K$
Original BERT model $M$.

for each document $d \in D$ do
    Extract the vanilla [CLS]-Token attention weight $\alpha_d$ from $M$, w.r.t. Eq. (4)
    Compute the document term distribution $P(w_k|\theta_d)$ based on $\alpha_d$, w.r.t. Eq. (5)
end

Approximate the random term distribution $P(w_k|\theta_{random})$ based on all $P(w_k|\theta_d)$ in $D$, w.r.t. Eq. (6)

for each document $d$ in $D$ do
    Compute the contrastive term distribution $P(w_k|\theta_{contrastive})$ based on $P(w_k|\theta_d)$ and $P(w_k|\theta_{random})$, w.r.t. Eq. (7) and Eq. (8)
    Sample the size of the word set $l$, w.r.t. Eq. (9)
    Sample a pair of word sets $s_1$ and $s_2$ according to $P(w_k|\theta_{contrastive})$, w.r.t. Eq. (10)
end

semantic information in a document as possible, thus becomes less discriminative (i.e., not to differentiate common words which are randomly distributed in the corpus [1]). Inspired by the divergence-from-randomness idea, we introduce a novel contrastive method to leverage BERT’s attention mechanism to sample representative words from a document, and re-train BERT itself towards the PROP learning objective.

3.2 Contrastive Sampling for ROP with BERT

The basic idea of divergence-from-randomness [1] says that the informativeness/representativeness of a term could be computed by measuring the divergence between a term distribution produced by a random process and the document term distribution. Following this idea, we introduce a contrastive method to leverage BERT’s attention mechanism to sample representative words from a document for the ROP task construction.

In the following, we first show how to compute the document term distribution based on BERT’s vanilla [CLS]-Token attention. We then take an expectation over all the term distributions in the collection to approximate the term distribution produced by a random process. Finally, we compute the cross-entropy (i.e., the divergence) between the two distributions, i.e., the document term distribution and the random term distribution, to obtain the contrastive term distribution. The ROP task is then constructed by sampling pairs of representative word sets from the document.

3.2.1 Document Term Distribution. The document term distribution is computed based on BERT’s vanilla [CLS]-Token attention. We first introduce the self-attention mechanism in BERT and then show how to compute the document term distribution.

The multi-head attention in BERT’s self-attention layer [32] allows the model to jointly attend to information from different representation subspaces at different positions. Given an input document $d = (x_1, x_2, \ldots, x_T)$ with $T$ tokens, which starts with a special classification token ([CLS]), the $i$-th attention head for

\[
\text{Algorithm 1: Representative Word Sets Sampling}
\]

\[
\text{Input: Document Collection } D, \text{ Vocabulary } V = \{w_k\}_{k=1}^K \text{ Original BERT model } M.
\]

\[
\text{for each document } d \in D \text{ do}
\begin{align*}
\text{Extract the vanilla [CLS]-Token attention weight } \alpha_d & \text{ from } M, \text{ w.r.t. Eq. (4)} \\
\text{Compute the document term distribution } P(w_k|\theta_d) & \text{ based on } \alpha_d, \text{ w.r.t. Eq. (5)} \\
\end{align*}
\]

\[
\text{end}
\]

\[
\text{Approximate the random term distribution } P(w_k|\theta_{random}) \text{ based on all } P(w_k|\theta_d) \text{ in } D, \text{ w.r.t. Eq. (6)}
\]

\[
\text{for each document } d \text{ in } D \text{ do}
\begin{align*}
\text{Compute the contrastive term distribution } \\
P(w_k|\theta_{contrastive}) & \text{ based on } P(w_k|\theta_d) \text{ and } P(w_k|\theta_{random}), \text{ w.r.t. Eq. (7) and Eq. (8)} \\
\text{Sample the size of the word set } l, \text{ w.r.t. Eq. (9)} \\
\text{Sample a pair of word sets } s_1 \text{ and } s_2 \text{ according to } P(w_k|\theta_{contrastive}), \text{ w.r.t. Eq. (10)}
\end{align*}
\]

\[
\text{end}
\]

[CLS]+s+[SEP]+d+[SEP], and fed into the Transformer. Let a triple $(s_1, s_2, d)$ denote the pair of word sets sampled from the corresponding document $d$. The likelihood score of the pair of word sets are computed as follows,

\[
score = \prod_{w \in s} P(w|\theta_d),
\]

where $\theta_d$ is the multinomial unigram language model with Dirichlet prior smoothing. Suppose set $s_1$ has a higher likelihood score than $s_2$ with respect to $d$, the ROP task can be formulated by a typical pairwise loss, i.e., hinge loss, for the pre-training,

\[
\mathcal{L}_{ROP} = \max(0, 1 - P(s_1|d) + P(s_2|d)),
\]

where the likelihood of each set $P(s|d)$ denotes how representative the word set is to the document. It is obtained by applying a MLP over the final hidden state of [CLS] when pre-training.

Masked Language Modeling (MLM). Similar to BERT, PROP also adopts the MLM [12] objective as one of its pre-training objectives besides the ROP objective, to build good contextual representations for the query and the document. Specifically, the MLM loss $\mathcal{L}_{MLM}$ is defined as:

\[
\mathcal{L}_{MLM} = - \sum_{x \in m(x)} \log p(\hat{x} | x_{m(x)}),
\]

where $x$ denotes the input sentences, $m(x)$ and $x_{m(x)}$ denotes the masked words and the rest words from $x$, respectively.

3 B-PROP

In this section, we describe our bootstrapped pre-training method with representative words prediction (B-PROP) for ad-hoc retrieval in detail.

3.1 Motivation

The core of the ROP pre-training task mentioned above is to sample representative word sets from a document. As demonstrated in [21], a random sampling strategy would lead to much worse performance of the pre-trained model as compared with the document language model-based sampling. This result clearly indicates that the quality of the "representative" word sets is key to the success of the pre-training.

Although the unigram language model with Dirichlet prior smoothing has shown its promising results in PROP, its limitation is also clear. The term independence assumption adopted by the language model makes it difficult to fully capture the document semantic by ignoring the correlation between words, which in turn affect the representativeness of the sampled words. To address this problem, we propose to leverage BERT, a powerful contextual language model, to replace the unigram language model for the ROP task construction. The key assumption here is that a better document language model would lead to higher quality of representative words sampling.

Unfortunately, as shown in the pilot analysis in Table 1, directly sampling from the term distribution based on BERT’s [CLS]-Token attention may not necessarily produce representative words for a document, since the vanilla attention also favors common words with high frequency, e.g., stop words in IR. The underlying reason is that the contextual language model focuses on encoding as much
[CLS] is,
\[
\text{Attention}_i(h_{[CLS]}) = \sum_{t=1}^{T} \text{softmax} \left( \frac{W_i^h h_{[CLS]} + W_i^h h_t}{\sqrt{d/h}} \right) \alpha_i \beta_i,
\]
where \( h_i \) denotes a \( d \) dimensional hidden vector of the \( t \)-th sequence token. \( h_{[CLS]} \) denotes a \( d \) dimensional hidden vector of the [CLS] token. \( h \) is the number of self-attention heads. \( W_i^h, W_i^k \) and \( W_i^q \) are learned matrices of size \( d/h \times d \times d \). Thus, the the outputs of the \( h \) attentions heads for [CLS] are \( \{\text{Attention}_i(h_{[CLS]})\}_{i=1}^h \).

Specifically, we could extract the attention weight \( a_i' \) from the \( i \)-th attention head for [CLS] in BERT’s final layer, i.e.,
\[
a_i' = \text{softmax} \left( \frac{W_i^h h_{[CLS]} + W_i^h h_t}{\sqrt{d/h}} \right).
\]
Then, we average the [CLS]-Token attention weights across \( h \) attention heads, to obtain the final vanilla [CLS]-Token attention weight \( a' \) of each token, i.e.,
\[
a' = \frac{1}{h} \sum_{i=1}^{h} a_i'.
\]

Typically, a term may appear multiple times within the same document. Based on the vanilla [CLS]-Token attention, for each distinct term \( w_k \) in the vocabulary \( V = \{w_k\}_{k=1}^K \), we first add up the final vanilla [CLS]-Token attention weights of the same term over different positions in a document \( d \), i.e.,
\[
\hat{\beta}_{w_k} = \sum_{x_t = w_k} a', x_t \in d.
\]
The document would be dominated by terms with large attention weights, to alleviate this problem, we leverage the term saturation function used as part of BM25 [31]. The distinct term attention weight is thus computed by,
\[
\beta_{w_k} = \frac{\hat{\beta}_{w_k}}{b + \hat{\beta}_{w_k}},
\]
where \( b \) is a hyper-parameter controlling the shape of the saturation curve. Note the distinct term attention weight is set as zero if the document does not contain the corresponding term.

Then, we normalize the distinct term attention weights of all the terms in the vocabulary \( V \), to obtain the document term distribution \( P(w_k|\theta_d) \),
\[
P(w_k|\theta_d) = \frac{\exp(\beta_{w_k})}{\sum_{w_k \in V} \exp(\beta_{w_k})},
\]

3.2.2 Random Term Distribution. To approximate the term distribution produced by a random process, we take an expectation over all the term distributions in the document collection \( D \). The random term distribution \( P(w_k|\theta_{random}) \) is computed by,
\[
P(w_k|\theta_{random}) = \mathbb{E}(w_k|D) = \frac{1}{|D|} \sum_{d \in D} P(w_k|\theta_d),
\]
where \( D \) is the document collection containing \(|D|\) documents.

3.2.3 Constrastive Sampling for ROP. Finally, we compute the cross-entropy (i.e., the divergence) between the document term distribution \( P(w_k|\theta_d) \) and the random term distribution \( P(w_k|\theta_{random}) \) to obtain the contrastive term distribution \( P(w_k|\theta_{contrastive}) \), i.e.,
\[
\hat{y}_{w_k} = -P(w_k|\theta_d) \log P(w_k|\theta_{random}),
\]
\[
P(w_k|\theta_{contrastive}) = \frac{\exp(\hat{y}_{w_k})}{\sum_{w_k \in V} \exp(\hat{y}_{w_k})},
\]
where the softmax function ensures the contrastive term distribution is valid. After obtaining the contrastive term distribution, we could sample pairs of word sets from documents for the ROP task construction. The detailed sampling process is described in Algorithm 1.

Given a document \( d \), we sample a pair of word sets, each as a generated pseudo query, according to the contrastive term distribution. Following the procedures in PROP, we first use a Poisson distribution [2, 3] to sample a positive integer \( l \) as the size of the word set, i.e.,
\[
P(x) = \frac{\lambda^x e^{-\lambda}}{x!}, x = 1, 2, 3, \ldots,
\]
where \( \lambda \) is a hyper-parameter that indicates the expectation of interval.

Then, we sample a pair of word set \( s_1 \) and \( s_2 \) with the same size \( l \) in parallel from the vocabulary \( V \) according to the contrastive term distribution, i.e.,
\[
s = \text{Sample}(V), w_k \sim P(w_k|\theta_{contrastive}).
\]

3.3 Re-Training BERT with ROP and MLM

Based on the above sampled representative word sets, we compute the likelihood score of each word set according to Eq. (1). We then re-train BERT towards the tailored objective for IR, i.e., the ROP objective in Eq. (2), jointly with the MLM objective in Eq. (3), as follows,
\[
\mathcal{L}_{total} = \mathcal{L}_{ROP} + \mathcal{L}_{MLM}.
\]
The pre-trained model is named as B-PROP for short. B-PROP could then be fine-tuned on a variety of downstream ad-hoc retrieval tasks.

4 EXPERIMENTS

In this section, we conduct experiments to demonstrate the effectiveness of B-PROP on a set of benchmark collections.

4.1 Experimental Settings

We first introduce our experimental setting, including datasets, baseline methods, evaluation methodology, and implementation details.

4.1.1 Datasets. Here, we introduce the datasets for pre-training and the downstream tasks.

- Pre-training Corpus. Following PROP, we also leverage two large public document collections, including Wikipedia and MS MARCO Document Corpus, for pre-training. Wikipedia is a widely used corpus which contains tens of millions of well-formed Wiki-articles.
We adopt three types of baselines for comparison,

- Query likelihood model (QL) [37] is one of the best per-

- MS MARCO Document Corpus is another large-scale docu-

- MS MARCO Document Corpus respectively, we obtain two types of pre-training

- Downstream Tasks. We fine-tune B-PROP on 7 representative

- Milton Query Track 2007 (MQ2007) [5] is a large Web collection with over 50M
documents, where the topics are accumulated from TREC Web Tracks 2009, 2010, and 2011.

- Gov2 [7] contains about 150 queries and 25M documents collected from the .gov domain Web pages, where the topics are accumulated from TREC Terabyte Tracks 2004, 2005, and 2006.

- Million Query Track 2007 (MQ2007) [5] is a LETOR [28] benchmark dataset with 1692 queries using the GOV2 collection.

- Million Query Track 2008 (MQ2008) [5] is another LETOR [28] benchmark dataset with 784 queries using the GOV2 collection.

- MS MARCO Document Ranking (MS MARCO) [5] is a large-scale benchmark dataset for web document retrieval, with about 0.37 million training queries.

- TREC 2019 Deep Learning Track (TREC DL) [9] replaces the test set in MS MARCO with a novel set produced by TREC with more comprehensive labeling.

4.1.2 Baselines. We adopt three types of baselines for comparison, including traditional retrieval models, popular neural IR models and advanced pre-trained models.

- Traditional Retrieval Models include:
  - Query likelihood model (QL) [37] is one of the best performing language models based on Dirichlet smoothing.
  - BM25 [31] is another highly effective retrieval model that represents the classical probabilistic retrieval model.

- Neural IR Models include:
  - DRMM [15] is a deep relevance matching model which employs a joint deep architecture at the query term level for relevance matching.
  - Conv-KNRM [11] is a popular Convolutional Kernel-based Neural Ranking Model which models n-gram soft matches for ad-hoc retrieval.

- Pre-trained Models include:
  - BERT [12] is designed to pre-train deep bidirectional representations from large-scale unlabeled text with MLM and Next Sentence Prediction (NSP).
  - TransformerICT pre-trains with Inverse Cloze Task (ICT) [6] which is specifically designed for passage retrieval in QA scenario. ICT aims to teach the model to predict the removed sentence given a context text. We pre-train the Transformer model on Wikipedia with both ICT and MLM objectives.
  - PROP is a recently proposed pre-training model tailored for IR. By pre-training PROP on Wikipedia and MS MARCO Document Ranking dataset respectively, there are two types of PROP models denoted as PROPWiki and PROPMARCO, respectively.

4.1.3 Evaluation Methodology. For 5 small datasets, we conduct 5-fold cross-validation to minimize over-fitting without reducing the number of learning instances. We take the topic “titles” as queries for 5 small datasets and queries are randomly divided into 5 folds. The parameters for each model are tuned on 4-of-5 folds. The final fold in each case is used to evaluate the optimal parameters. This process is repeated 5 times, once for each fold. The final result is obtained by averaging the performance on each tested fold. Following [10, 15, 21, 22], we report the normalized discounted cumulative gain at rank 20 (nDCG@20), and precision at rank 20 (P@20) for Robust04, ClueWeb09-B and Gov2. Following [13, 26], we report the P@10 and nDCG@10 for MQ2007 and MQ2008.

For two large-scale ad-hoc retrieval collections, the top 100 documents are compared using the Mean Reciprocal Rank at 100 (MRR@100) for MS MARCO and nDCG@10 for TREC DL, which is suggested in the official instructions. We also evaluate MRR@10 and nDCG@100 for MS MARCO and TREC DL, respectively. For MS MARCO, we report the performance results on the dev set since the MS MARCO Document Ranking leaderboard limits the frequency of the submission. For TREC DL, we report results on the test set.

4.1.4 Implementation Details. Here, we describe the implementation details of B-PROP and baselines, pre-training procedures, and fine-tuning procedures.

- Model Implementations. For pre-trained models (i.e., BERT, TransformerICT, PROP and B-PROP), we use the popular Transformers library\(^2\) for implementations. For TransformerICT and PROP, parameters are initialized by the BERT’s checkpoint released by Google\(^3\). For B-PROP, since BERT applies WordPiece tokenization which could split an entire word into several subwords, we take the [CLS]-Token attention weight of the first subword as that of the entire word. The parameter b in the saturation function is set as 0.01.

- Pre-training Procedures. For MLM, following BERT, we randomly select 15% words in the input sequence, and the selected words are (1) 80% of time replaced by a mask token [MASK], (2) 10% of time replaced by a random token, (3) 10% of time unchanged. Note that the representative words sampled from a document are not considered to be masked for PROP and B-PROP. The input sequence are truncated to a maximum of 512 for all pre-trained models. We use Adam optimizer [17] with a linear warm-up technique over the first 10% steps to learn B-PROP’s

\(^2\)[https://github.com/huggingface/transformers

\(^3\)[https://github.com/google-research/bert

### Table 2: Statistics of the ad-hoc retrieval datasets

| Dataset        | Genre     | #Queries | #Documents |
|----------------|-----------|----------|------------|
| Robust04       | news      | 250      | 0.5M       |
| ClueWeb09-B    | web pages | 150      | 50M        |
| Gov2           | .gov pages| 150      | 25M        |
| MQ2007         | .gov pages| 1,692    | 25M        |
| MQ2008         | .gov pages| 784      | 25M        |
| MS MARCO       | web pages | 0.37M    | 3.2M       |
| TREC DL        | web pages | 0.37M    | 3.2M       |
In this section, we first look at the performance comparisons between B-PROP and the baselines on the 5 small datasets. As shown in Table 3, we can observe that: (1) Compared with traditional retrieval models and neural IR models, the pre-trained model BERT can improve the performance significantly across all the tasks especially on datasets with very few training instances. For example, the relative improvements of BERT over BM25 and DRMM are about 28% and 20% respectively in terms of nDCG@20 on ClueWeb09-B. These results show that pre-training on large-scale text corpus could learn useful text representations to benefit ranking tasks. (2) TransformerICT obtains similar performance with the BERT model on all 5 datasets, indicating that the ICT objective designed for passage retrieval in QA scenario cannot bring additional benefits to ad-hoc document retrieval as compared with BERT. (3) PROP achieves the best performance among all the baseline methods, demonstrating that using a pre-training objective that more closely resembles the downstream task could lead to better fine-tuning performance. For example, the relative improvements of PROP over BERT and TransformerICT are about 12% and 10% respectively in terms of nDCG@20 on ClueWeb09-B.

When we look at B-PROP, we find that: (1) Our B-PROP model outperforms the best baseline PROP significantly. The better results of our models over PROP demonstrate the effectiveness of leveraging the powerful contextual language model BERT to replace the classical unigram language model for the ROP task construction. (2) Among the two variants of our model, B-PROP\textsubscript{Wiki} performs better than B-PROP\textsubscript{MARCO} on Robust04, while B-PROP\textsubscript{MARCO} performs better than B-PROP\textsubscript{Wiki} on ClueWeb09-B and Gov2. By further analyzing the datasets, we find that the news articles in Robust04 are more similar to well-formed Wiki-articles in Wikipedia, while the Web pages in ClueWeb09-B and Gov2 are more similar to the Web corpora with a similar domain to downstream tasks. Therefore, the reason might be that pre-training on the text corpora with a similar domain to downstream tasks could lead to better fine-tuning performance.

Then, we look at the performance comparisons between B-PROP and the baselines on 2 large-scale datasets. As shown in Table 4, we can see that: (1) The performance trend of different types of models on these two large-scale datasets is quite consistent with that on the previous small datasets. Note that neural IR models perform poorly on these 2 datasets indicating that they are under-fitting due to lack of training data. (2) B-PROP outperforms the best baseline PROP on all 5 datasets, demonstrating the effectiveness of leveraging the powerful contextual language model BERT to replace the classical unigram language model for the ROP task construction. (3) PROP achieves the best performance among all the baseline methods, demonstrating that using a pre-training objective that more closely resembles the downstream task could lead to better fine-tuning performance. For example, the relative improvements of PROP over BERT and TransformerICT are about 12% and 10% respectively in terms of nDCG@20 on ClueWeb09-B.

When we look at B-PROP, we find that: (1) Our B-PROP model outperforms the best baseline PROP significantly.
to the small model size. (2) The improvements of B-PROP\textsubscript{Wiki} over BERT and PROP\textsubscript{Wiki} are around 5% and 3% respectively in terms of MRR@100 on MS MARCO under both the rerank and fullrank settings. The results indicate that B-PROP can benefit downstream ad-hoc retrieval tasks not only on small-sized training data, but also on large-scale training data. (3) Among the two variants of our models, B-PROP\textsubscript{MARCO} performs better than B-PROP\textsubscript{Wiki} in terms of all the evaluation metrics on all the two datasets. This further confirms the importance of the consistency between the fine-tuning document corpus and the pre-training document corpus.

Finally, by comparing the performance of B-PROP on all datasets, we find that the improvements of B-PROP over the baselines on small-sized datasets are significantly higher than that on large-scale datasets. For example, the improvement of B-PROP\textsubscript{MARCO} over BERT is about 20% on ClueWeb09-B in terms of nDCG@20, while the improvement of B-PROP\textsubscript{MARCO} over BERT is about 6% in terms of MRR@100 on MS MARCO under the fullrank subtask. The reason might be that the large-scale labeled data is sufficient to train an accurate enough model, and thus they may gain little benefit from the pre-trained models. However, it is often time-consuming and expensive to annotate high-quality data for IR. Therefore, it is of great value to explore the pre-training objectives tailored for ad-hoc retrieval since most real-world situations in IR have only limited data or poor-quality data.

### 4.3 Impact of Contrastive Method

Since we propose a contrastive method to sample representative words for the ROP task construction, we here study the effect of the contrastive method. Specifically, we conduct experiments to compare the performance of B-PROP\textsubscript{MACRO} by directly sampling representative words from the document term distribution (i.e., Eq. (5)) and the contrastive term distribution (i.e., Eq. (7)) in the ROP task. Due to the limited space, we only report the nDCG@20 results on Robust04 and Gov2, and the nDCG@10 results on MQ2007 and TREC DL.

![Figure 1: Comparison of pre-training B-PROP\textsubscript{MACRO} with and without contrastive methods. We report nDCG@20 for Robust04 and Gov2, and nDCG@10 for MQ2007 and TREC DL.](image_url)

TREC DL. As shown in Figure 1, we can find that the performance of B-PROP\textsubscript{MACRO} by sampling representative words from the contrastive term distribution is better than that from the document term distribution. As discussed above, we have found that the document term distribution based on BERT’s vanilla [CLS]-Token attention favors common words, which may not necessarily produce representative words. These results further confirm the effectiveness of the contrastive method by computing cross-entropy between the document term distribution and the random term distribution to obtain a representative/informative term distribution.

### 4.4 Impact of Saturation Function

A saturation function (i.e., Eq. 5) borrowed from BM25 is applied to help obtain a good document term distribution. The underlying belief is any one term’s contribution to the document score cannot exceed a saturation value, no matter how large it has the vanilla [CLS]-Token attention weight. Here, we study how such saturation
We find that pre-training steps may effect the performance of different downstream tasks with the similar/same domain, and it may also result in overfitting on the source domain.

### 4.6 Comparison of Transfer Learning from Supervised Data

Here, we compare pre-training on unlabeled corpus (i.e., documents) with transfer learning from supervised data (i.e., query-document pairs). We first train BERT on MS MARCO with labeled query-document pairs and then fine-tune this model on the Robust04 and Gov2 datasets. We denote this model as BERT\textsubscript{MSMARCO}. For

| Model       | Robust04 (nDCG@20) | Robust04 (P@20) | Gov2 (nDCG@20) | Gov2 (P@20) |
|-------------|---------------------|-----------------|----------------|--------------|
| BERT        | 0.459*              | 0.389*          | 0.495*         | 0.586*       |
| BERT\textsubscript{MSMARCO} | 0.462*              | 0.391*          | 0.512*         | 0.590*       |
| B-PROP\textsubscript{MACRO} | \textbf{0.510}     | \textbf{0.429}  | \textbf{0.552} | \textbf{0.606} |

*B-PROP\textsubscript{MACRO} and BERT, we directly fine-tune the model on the Robust04 and Gov2 datasets.

As shown in Table 6, we can see that: (1) BERT\textsubscript{MSMARCO} improves the performance over BERT with only a small margin. The reason might be that the learning objective of BERT\textsubscript{MSMARCO} on the MS MARCO is the same as the final ranking objective on the Robust04 and Gov2. Such transfer learning methods only work if the initial and target problems are similar enough. While the knowledge in the MS MARCO doesn’t seem to be effectively transferred to the Robust04 and Gov2 due to the difference of the domain, query type, and matching pattern. (2) By pre-training towards the MLM and ROP objective, B-PROP\textsubscript{MACRO} achieves the best performance, which again demonstrate its effectiveness. This indicates that pre-training is a more effective and generalizable method to benefit the downstream tasks.

### 4.7 Zero- and Low-Resource Setting

In real-world practice, it is often time-consuming and difficult to collect a large number of relevance labels in IR. To simulate the low-resource IR setting, we pick a limited number of queries from 5 small ad-hoc retrieval datasets to fine-tune the corresponding best performing B-PROP. For 5 small datasets, we fine-tune B-PROP with the batch size as 8, and choose the best learning rate in the range of [1e-5, 2e-5] on the validation set without warm-up.

As shown in Figure 3, we report the nDCG@20 results for the Robust04, ClueWeb09-B and Gov2, and the nDCG@10 results for the MQ2007 and MQ2008. We can see that: (1) B-PROP outperforms

Figure 2: The performance of B-PROP\textsubscript{MACRO} on different downstream tasks over pre-training steps.

Table 5: Comparison of B-PROP\textsubscript{MACRO} pre-training with and without applying the saturation function (SAT), and using SAT with different b. Two-tailed t-tests demonstrate the degradation with respect to (w/SAT, k=0.01) method are statistically significant (− indicates p ≤ 0.05).

| Model       | Robust04 (nDCG@20) | Robust04 (nDCG@20) | Robust04 (MRR@100) | Robust04 (MRR@100) |
|-------------|---------------------|---------------------|---------------------|---------------------|
| w/o SAT     | 0.469*              | 0.308*              | 0.424*              | 0.424*              |
| w/ SAT, b=0.05 | 0.486*              | 0.319*              | 0.432*              | 0.432*              |
| w/ SAT, b=0.01 | 0.510               | 0.353               | 0.441               | 0.441               |
| w/ SAT, b=0.001 | 0.475*              | 0.313*              | 0.426*              | 0.426*              |
PROP significantly on all the datasets using the same number of limited supervised data. (2) B-PROP fine-tuned on limited supervised data can achieve comparable performance with BERT fine-tuned on the full supervised datasets. For example, B-PROP fine-tuned with 10 queries could outperform BERT fine-tuned on the full supervised datasets on Robust04, ClueWeb09-B and Gov2. (3) Under the zero resource setting, for example, B-PROP could outperform BM25 on ClueWeb09-B and Gov2.

5 RELATED WORK

In this section, we briefly review two lines of related work, i.e., pre-trained language models and pre-training objectives tailored for IR.

5.1 Pre-trained Language Models

Pre-training language representation models on large-scale unlabeled text corpora and then fine-tuning on downstream tasks have achieved remarkable success in NLP field [4, 12, 29, 36, 38]. A notable example is BERT [12], a bidirectional transformer-based language model that pre-trains with MLM and NSP to learn both contextual text representations and coherence relationship between two pieces of text. By adding a simple classification layer on top of BERT, BERT could adapt to various classification/regression tasks and outperform many task-specific architectures. The success of pre-trained models in NLP has also attracted a lot of attention in the IR community. Nogueira et al. [23–25], Dai et al. [10] and MacAvaney et al. [22] explored to incorporate these pre-trained language models (e.g., BERT) to search tasks. The common approach is to feed the query-document pair into BERT and compute the relevance score using an MLP layer over the final state of the [CLS] token. They found that BERT could significantly outperform those neural ranking models on ad-hoc retrieval tasks.

5.2 Pre-training Objectives Tailored for IR

Beyond the direct application of existing pre-trained language models, there have been some pioneer studies on the pre-training objectives tailored for IR. Besides ICT [6, 19], Chang et al. [6] proposed another two pre-training tasks for passage retrieval in QA scenario: One is the Body First Selection (BFS) where the query is a random sentence in the first section of a Wikipedia page, and the document is a passage from another page where there is a hyperlink link to the page of the query. The underlying idea is to resemble the relevance relationship between natural language questions and answer passages. However, some tasks even depend on certain special document structures, e.g., hyperlink. Recently, Ma et al. [21] proposed a novel pre-training objective tailored for ad-hoc retrieval, i.e., representation words prediction (ROP), and achieved new SOTA on a variety of ad-hoc retrieval tasks. As described in Section 2, PROP might be bounded by the classical language model adopted in the ROP task construction.

6 CONCLUSION

In this paper, we proposed B-PROP to address the problem of PROP where the effectiveness might be bound by the statistical document language model for the ROP task construction. The key idea of B-PROP is to leverage the powerful contextual language model BERT to replace the unigram language model for the ROP task construction and re-train BERT towards the tailored objective for IR. Inspired by the divergence-from-randomness idea, we introduced a contrastive method to measure the divergence between the document term distribution and the random term distribution produced by a random process to compute the informativeness/representativeness of a term. Therefore, in B-PROP, we adopted this contrastive method for sampling to construct the ROP task, and re-train BERT towards the PROP learning objective. Experimental results demonstrated the effectiveness of B-PROP on the downstream ad-hoc retrieval tasks. In the future work, we would like to adopt ROP objective for the first-stage retrieval in IR.

ACKNOWLEDGMENTS

This work was supported by Beijing Academy of Artificial Intelligence (BAAI) under Grants No. BAAI2019ZD0306, and funded by the National Natural Science Foundation of China (NSFC) under Grants No. 62006218, 61902381, 61773362, and 61872338, the Youth Innovation Promotion Association CAS under Grants No. 20144310, 2016102, and 2021100, the National Key RD Program of China under Grants No. 2016QY02D0405, the Lenovo-CAS Joint Lab Youth Scientist Project, the K.C.Wong Education Foundation, and the Foundation and Frontier Research Key Program of Chongqing Science and Technology Commission (No. cstc2017jcyjBX0059).
REFERENCES

[1] G. Amati and C. Rijkeberg. 2002. Probabilistic models of information retrieval based on measuring the divergence from randomness. ACM Trans. Inf. Syst. (2002), 357–389.

[2] Avi Arampatzis and Jaap Kamps. 2008. A Study of Query Length. In Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, New York, NY, USA, 811–812.

[3] Leif Azzopardi, Maarten de Rijke, and Kristzian Balog. 2007. Building Simulated Queries for Known-Item Topics: An Analysis Using Six European Languages. In Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, New York, NY, USA, 455–462.

[4] T. Brown and B. Mann et al. 2020. Language Models are Few-Shot Learners. ArXiv abs/2005.14165 (2020).

[5] Daniel Fernando Campos, T. Nguyen, M. Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, L. Deng, and Bhaskar Mitra. 2016. MS MARCO: A Human Generated Machine Reading Comprehension Dataset. ArXiv abs/1611.09268 (2016).

[6] Wei-Cheng Chang, F. Yu, Yin-Wen Chang, Yiming Yang, and S. Kumar. 2019. Pre-training Tasks for Embedding-based Large-scale Retrieval. International Conference on Learning Representations (2019).

[7] C. Clarke, Nick Craswell, and I. Soboroff. 2004. Overview of the TREC 2004 Terabyte Track. In TREC.

[8] C. Clarke, Nick Craswell, and I. Soboroff. 2009. Overview of the TREC 2009 Web Track. In TREC.

[9] Nick Craswell, Bhaskar Mitra, E. Yilmaz, Daniel Fernando Campos, and E. Voorhees. 2020. Overview of the TREC 2019 deep learning task. ArXiv abs/2003.07820 (2020).

[10] Zhuyun Dai, Chenyan Xiong, J. Callan, and Zhuyun Liu. 2018. Convolutional Neural Networks for Soft-Matching N-Grams in Ad-hoc Search. Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining (2018).

[11] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers). Association for Computational Linguistics, New Orleans, Louisiana, 2227–2237.

[12] Matthew Peters, Mark Neumann, Michael Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep Contextualized Word Representations. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers). Association for Computational Linguistics, New Orleans, Louisiana, 3571–3582.

[13] T. Qin and T. Liu. 2013. Introducing LETOR 4.0 Datasets. ArXiv abs/1306.2597 (2013).

[14] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. J. Mach. Learn. Res. 21 (2020), 140:1–140:67.

[15] Pawan Rujipakjun, Jie Zhang, Konstantin Lopryanov, and Percy Liang. 2016. SQuAD: 100, 000+ Questions for Machine Comprehension of Text. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing.

[16] S. Robertson and H. Zaragoza. 2009. The Probabilistic Relevance Framework: BM25 and Beyond. Found. Trends Inf. Retr. 3 (2009), 333–389.

[17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Ethan Ilievski, Mateusz Wojna, micun, and Illia Sutskever. 2017. Attention is All You Need. In Proceedings of the 31st Conference on Neural Information Processing Systems (Long Beach, California, USA). Curran Associates Inc., Red Hook, NY, USA, 6000–6010.

[18] E. Voorhees. 2004. Overview of the TREC 2004 Robust Task.

[19] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2018. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. In BlackboxNLP@EMNLP.

[20] Wei Yang, Haotian Zhang, and Jimmy Lin. 2020. Simple Applications of BERT for Ad Hoc Document Retrieval. ArXiv abs/2003.07820 (2020).

[21] Z. Yang, Zhiqiang Dai, Yiming Yang, J. Carbonell, R. Salakhutdinov, and Quoc V. Le. 2019. XLNet: Generalized Autoregressive Pretraining for Language Understanding. In Proceedings of the 10th International Conference on Learning Representations, Long Beach, CA, USA, May 7-9, 2019, Conference Track Proceedings, Joshua Bengio and Yann LeCun (Eds.).

[22] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Karunya Kawakami, and Chris Dyer. 2016. Neural Architectures for Named Entity Recognition. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Association for Computational Linguistics, San Diego, California, 260–270.

[23] Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent Retrieval for Weakly Supervised Open Domain Question Answering. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, 6086–6096.

[24] Jimmy Lin, Rodrigo Nogueira, and A. Yates. 2020. Pretrained Transformers for Text Ranking: BERT and Beyond. ArXiv abs/2010.06467 (2020).

[25] Shane MacAvaney, Andrew Yates, Arman Cohen, and Nash Goharian. 2019. CEDR: Contextualized Embeddings for Document Ranking. Proceedings of the 42nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (2019).

[26] Rodrigo Nogueira and Kyunghyun Cho. 2019. Passage Re-ranking with BERT. ArXiv abs/1901.04085 (2019).

[27] Rodrigo Nogueira, Zhiyong Jiang, and Jimmy Lin. 2020. Document Ranking with a Pretrained Sequence-to-Sequence Model. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing.

[28] Rodrigo Nogueira, Wei Yang, Jimmy Lin, and Kyunghyun Cho. 2019. Document Expansion by Query Prediction. ArXiv abs/1904.08375 (2019).

[29] Liang Pang, Yanyan Lan, J. Guo, Jun Xu, J. Xu, and X. Cheng. 2017. DeepRank: A New Deep Architecture for Relevance Ranking in Information Retrieval. Proceedings of the 2017 ACM Conference on Information and Knowledge Management (2017).

[30] Wei Yang, Haotian Zhang, and Jimmy Lin. 2020. Simple Applications of BERT for Ad Hoc Document Retrieval. ArXiv abs/2003.07820 (2020).

[31] S. Robertson and H. Zaragoza. 2009. The Probabilistic Relevance Framework: BM25 and Beyond. Found. Trends Inf. Retr. 3 (2009), 333–389.

[32] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Ethan Ilievski, Mateusz Wojna, micun, and Illia Sutskever. 2017. Attention is All You Need. In Proceedings of the 31st Conference on Neural Information Processing Systems (Long Beach, California, USA). Curran Associates Inc., Red Hook, NY, USA, 6000–6010.

[33] E. Voorhees. 2004. Overview of the TREC 2004 Robust Task.

[34] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2018. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. In BlackboxNLP@EMNLP.

[35] Wei Yang, Haotian Zhang, and Jimmy Lin. 2019. Simple Applications of BERT for Ad Hoc Document Retrieval. ArXiv abs/1903.10972 (2019).

[36] Z. Yang, Zhiqiang Dai, Yiming Yang, J. Carbonell, R. Salakhutdinov, and Quoc V. Le. 2019. XLNet: Generalized Autoregressive Pretraining for Language Understanding. In NeurIPS.

[37] ChengXiang Zhai. 2008. Statistical Language Models for Information Retrieval: A Critical Review. Found. Trends Inf. Retr. 2 (2008), 137–213.

[38] Jingqiang Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. 2020. PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Hal Daume Ill and Aarti Singh (Eds.). 11328–11339.