Improving Embedding-based Large-scale Retrieval via Label Enhancement

Peiyang Liu[1,2], **Xi Wang**[2], **Sen Wang**[2], **Wei Ye**[1,*], **Xiangyu Xi**[1,3] and **Shikun Zhang**[1]
[1] National Engineering Research Center for Software Engineering, Peking University, China,
[2] PX Securities, Beijing, China,
[3] Meituan-Dianping Group, Beijing, China
{liupeiyang, wye, zhangsk}@pku.edu.cn,
{wangxi, wangsen}@pxsec.cn,
xixiangyu@meituan.com

Abstract

Current embedding-based large-scale retrieval models are trained with 0-1 hard label that indicates whether a query is relevant to a document, ignoring rich information of the relevance degree. This paper proposes to improve embedding-based retrieval from the perspective of better characterizing the query-document relevance degree by introducing label enhancement (LE) for the first time. To generate label distribution in the retrieval scenario, we design a novel and effective supervised LE method that incorporates prior knowledge from dynamic term weighting methods into contextual embeddings. Our method significantly outperforms four competitive existing retrieval models and its counterparts equipped with two alternative LE techniques by training models with the generated label distribution as auxiliary supervision information. The superiority can be easily observed on English and Chinese large-scale retrieval tasks under both standard and cold-start settings.

1 Introduction

Retrieval systems such as search engines have been a vital tool in helping people access the vast amount of information online. As shown in Figure 1, existing methods for large-scale retrieval will first utilize a less powerful but more efficient retrieval algorithm (*Retriever*) to reduce the potential candidates, and then employ more powerful models (*Ranker*) to re-rank the retrieved documents (Padaki et al., 2020; Mass and Roitman, 2020). We will focus on improving *Retriever* in this paper.

With pre-trained word embeddings (Mikolov et al., 2013b,a; Pennington et al., 2014; Liu et al., 2020) and language models (e.g., BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019)) achieving great success in a wide variety of NLP tasks, researchers have begun to leverage BERT-style models to solve large-scale retrieval problems. These models consider the retrieval phase as a regression task trained with 0-1 hard labels, representing only two types of relevance degrees (relevant or irrelevant) between query-document pairs (Chang et al., 2020; Lu et al., 2020).

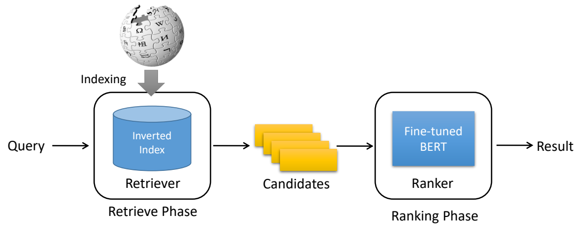


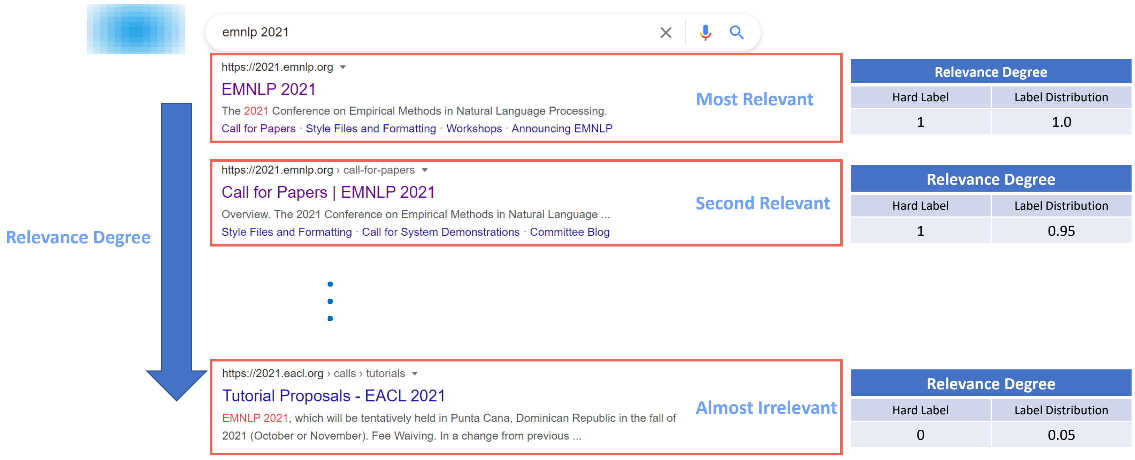
Figure 1: The architecture of classical large-scale retrieval systems.

The relevance degrees between queries and documents, however, can have much more possibilities. For example, we present a query and three actual results retrieved by the Google search engine in Figure 2. Though all three documents are relevant to the query, the relevance degrees can vary significantly if we assign a real-valued number indicating to what extent a query and a document relate. On the other hand, even if a query and a document are marked as irrelevant by the hard label, a weak relevance degree may exist between them. In such scenarios, label distribution (Geng, 2016), which involves the relevance degrees between queries and documents, is a more reasonable description of an instance. The observation inspires us to explore the label distribution to improve existing large-scale *Retriever* models trained with hard labels. We can easily expect the following two novel LE methods for *Retriever* models.

- One straightforward LE method in our scenario is to exploit the semantic relevance between queries and documents based on classic term weighting methods (e.g., TF-IDF (Spärck Jones, 1972, 2004)). The problem with this method is that term weight will be

[*] Corresponding author

Findings of the Association for Computational Linguistics: EMNLP 2021, pages 133–142
November 7–11, 2021. ©2021 Association for Computational Linguistics



emnlp 2021

https://2021.emnlp.org
EMNLP 2021
The 2021 Conference on Empirical Methods in Natural Language Processing.
Call for Papers · Style Files and Formatting · Workshops · Announcing EMNLP

Relevance Degree	
Hard Label	Label Distribution
1	1.0

https://2021.emnlp.org › call-for-papers
Call for Papers | EMNLP 2021
Overview. The 2021 Conference on Empirical Methods in Natural Language ...
Style Files and Formatting · Call for System Demonstrations · Committee Blog

Relevance Degree	
Hard Label	Label Distribution
1	0.95

https://2021.eacl.org › calls › tutorials
Tutorial Proposals - EACL 2021
EMNLP 2021, which will be tentatively held in Punta Cana, Dominican Republic in the fall of 2021 (October or November). Fee Waiving. In a change from previous ...

Relevance Degree	
Hard Label	Label Distribution
0	0.05

Figure 2: Retrieval results of the online search engine. Hard labels can only label these results as relevant or irrelevant, ignore relevance degrees.

static and context-free. For example, given the sentence "EMNLP 2021 is held after ACL 2021, accepted papers will be published in ACL Anthology." the first "ACL" is a conference name, while the second "ACL" is a professional society, they should have different term weights. However, TF-IDF cannot distinguish them and will assign them unreasonable equal term weights.

- Another way to generate label distribution is by training a contextual-embedding-based model with hard labels and then exploiting the prediction scores as label distribution, widely used for knowledge distillation (Hinton et al., 2015) and performance improvement (Zhang et al., 2019). This label distribution, called dark knowledge by Furlanello et al. (2018), is generated implicitly and lacks clear physical interpretation. From this perspective, term weighting methods can bring complementary and more explainable prior knowledge beneficial to the *Retriever* model.

To this end, we choose to generate label distributions based on term weights method in a way that integrates the merits of the two paradigms above. Specially, we employ BERT to generate contextualized text representations and learn to predict term weight for each word with its TF-IDF value as the supervised signal. In this way, we achieve a dynamic term weight scorer, named BERT-Scorer. Based on BERT-Scorer, we can predict each word's contextual term weights in a query and a document. We then generate label distributions for the query-document pairs based on their term weights of overlapped words and finally train *Retriever* models with generated label distributions as auxiliary supervision information.

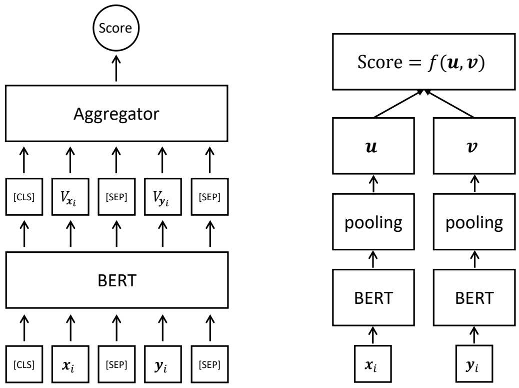
We have conducted extensive experiments on English and Chinese large-scale retrieval tasks under both standard and cold-start settings. Experimental results show that our approach significantly improves state-of-the-art models and has superiority over alternative label enhancement methods.

Our main contributions are as follows:

1. We propose to exploit query-document relevance degree to improve embedding-based *Retriever* models. This work is the first pioneer investigation on leveraging label enhancement to characterize relevance degree and incorporating it into the *Retriever* models to the best of our knowledge.

2. By designing a novel dynamic term-weight scorer that integrates contextual BERT representation and static TF-IDF information, we achieve a novel and effective label enhancement method that automatically generates label distributions for the retrieval tasks.

3. Our method significantly outperforms state-of-the-art models and its counterparts equipped with alternative label enhancement techniques on English and Chinese large-scale retrieval tasks under both standard and cold-start settings.



(a) The cross-attention architecture used by *Rankers*.

(b) The multi-tower architecture used by *Retrievers*.

Figure 3: Architecture of multi-tower models used by *Retriever* and cross-attention models used by *Ranker*.

2 Background and Related Work

2.1 BERT-style Retriever and Ranker

Large-scale retrieval is usually solved in two steps. The retrieval phase (*Retriever*) first reduces the solution space, returning a subset of candidate documents. The ranking phase (*Ranker*) then re-ranks the documents (Chang et al., 2020). Unlike *Ranker* witnessing significant advances recently due to the BERT-style pre-training tasks on cross-attention models (see left side in Figure 3) (Padaki et al., 2020; Mass and Roitman, 2020), the retrieval phase, which is the focus of this paper, remains less well studied.

Existing BERT-style *Rankers* can not be applied to large-scale retrieval problems. Since the prediction function $f(query, doc)$ with BERT is a pre-trained deep bidirectional Transformer model (Vaswani and Shazeer, 2017), we can not afford to apply the prediction process for every possible document given a query. Therefore, BERT-style *Retriever* will employ a multi-tower architecture (see the right side in Figure 3), in which embeddings of documents can be first predicted offline and then fetched to calculate the final relevance score efficiently. For example, we can deploy an inverted index based ANN (approximate near neighbor) search algorithms (Shrivastava and Li, 2014; Guo et al., 2016) to *Retriever*, and employ Faiss library (Johnson et al., 2017) to quantize the vectors and then implemented the efficient embedding search in *Retriever*.

As a representative BERT-style *Retriever*, Reimers and Gurevych (2019) use siamese and triplet network structures based on BERT to derive semantically meaningful sentence embeddings, which can be compared using cosine similarity. Some researchers further improve model performance by introducing external knowledge or data. For example, Chang et al. (2020) build a two-tower Transformer model with more pre-training data, which can significantly outperform the widely used BM-25 algorithm. Lu et al. (2020) distill knowledge from larger BERT into a two-tower architecture network for efficient retrieval. Liu et al. (2021) build a four-tower BERT model that leverages the distances between simple negative and hard negative instances for embedding-based large-scale retrieval.

2.2 Label Distribution and Label Enhancement

The process of generating label distributions from hard labels is defined as label enhancement (LE). LE has achieved remarkable results in many fields, e.g., computer vision (Gao et al., 2020; Xu et al., 2020) and biological information classification (Xu et al., 2019; Lv et al., 2019). Knowledge distillation from the deep learning community (Hinton et al., 2015) is another way to generates label distributions, also known as soft labels. The distillation process mainly refers to using prediction scores (e.g., SoftMax logits) of pre-trained models as auxiliary objectives.

We focus on embedding-based large-scale retrieval problems as the first touch on incorporating label enhancement into this field. It is worth noting that the primary purpose of LE is incorporating the possibility (or uncertainty) into the original hard label to facilitate model performances, rather than generating the ground truth label distribution.

3 The Proposed Approach

Given a training set $D = \{(\langle x_i, y_i \rangle, l_i) | 1 \le i \le N\}$ with N instances, the hard label $l_i \in \{0, 1\}$ denotes whether a query x_i and document y_i are relevant or not. Our proposed LE method can automatically generate label distributions d_i for each query-document pair $\langle x_i, y_i \rangle$, which is further introduced to assist retrieval tasks. The details are demonstrated in the following subsections.

3.1 Initial Term Weights

Given a positive training instance $(\langle x_i, y_i \rangle, l_i = 1)$, where x_i contains n tokens $\{w_1, w_2, ..., w_n\}$, proper term weights should reflect whether a term

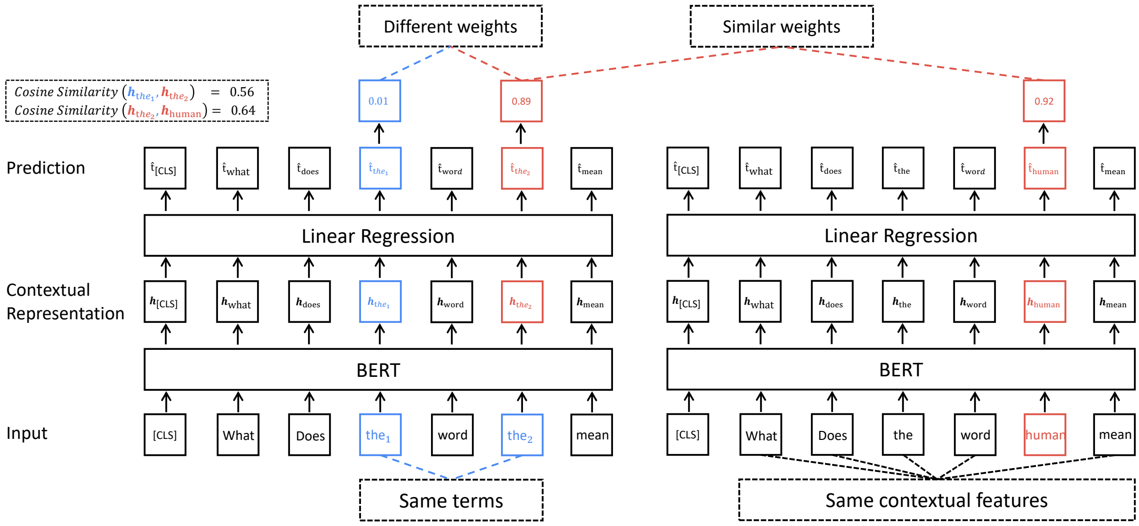


Figure 4: BERT is firstly adopted to generate contextualized representation. A linear regression layer is then used to estimate term weights for each token, with the corresponding TF-IDF scores as supervision signals. Two concrete queries are used as examples. Based on TF-IDF, the word "human" in q_2 can be easily identified as a critical term. Since the second "the$_2$" in q_1 has a similar context with "human", we can predict a more reasonable weight for "the$_2$" by incorporating TF-IDF into contextualized representations.

w_i is essential to the document or not. We propose to generate initial term weights by the TF-IDF method as follows:

$$t_{w_j}^{x_i,y_i} = \frac{\eta_{w_j,y_i}}{|y_i|} \log \frac{|Y|}{\eta_{w_j,Y}+1} \qquad (1)$$

where $t_{w_j}^{x_i,y_i}$ is the term weight of w_j in x_i corresponding to y_i, and η_{w_i,y_i} equals the number of times w_i appears in document y_i. Y is the set of all documents, and $\eta_{w_i,Y}$ equals the number of documents in which w_i appears in Y.

3.2 BERT-Scorer

The traditional term weight method such as TF-IDF is based on statistical features of documents. They produce static and context-free term weight and fail to capture the complex semantic features. To estimate the importance of a word in a specific text, the most critical problem is to generate features that characterize a word's relationships to the context. Recent contextualized neural language models like BERT have been shown to capture such properties through a deep neural network effectively (Dai and Callan, 2019).

As shown in Figure 4, for the example sentence q_1 "What does the word 'the' mean", the first "the$_1$" is a definite article and the second "the$_2$" is a noun. Another example sentence q_2 is "What does the word 'human' mean", which has the same context as the first sentence except for the keyword "human". Although the TF-IDF scores of "the$_1$" and "the$_2$" are equal, most words that have a similar context with "the$_2$" (e.g., the word "Human" in q_2) will be given reasonable TF-IDF scores. BERT can generate contextualized representations that characterize words' syntactic and semantic role in a given context. In this way, we can get relatively similar contextual embeddings for these words, hence predicting similar scores (e.g., 0.92 for "Human" and 0.89 for "the$_2$" according to actual BERT-Scorer predictions).

Based on BERT, we build a regression model named BERT-Scorer to generate dynamic context-aware term weights for queries and documents. Given the query x with n tokens $\{w_1, w_2, ..., w_n\}$, BERT is firstly adopted to encode each word sequence into a sequence of continuous representations as following:

$$\vec{H} = (\vec{h}_1, ..., \vec{h}_n) = \mathrm{BERT}(w_1, ..., w_n) \qquad (2)$$

A linear regression layer is then used to estimates the term weight for each word w_i as follows:

$$\hat{t}_{w_i}^{x} = \vec{W}\vec{h}_i + b \qquad (3)$$

where $\vec{W}$ and b are model parameters. Under such circumstance, our BERT-Scorer can effectively discriminate "the$_1$" and "the$_2$" according to the differences between h_{the_1} and h_{the_2}. The "human" and "the$_2$" have similar weights while the weight of "the$_1$" is much smaller.

During training, the initial term weights by TF-IDF are utilized as supervised signals. The optimization objective function is defined as the mean square error (MSE) between the predicted weights $\hat{t}$ and the target weights t as follows:

$$J(\theta) = \sum_{\langle x,y \rangle \in D} \sum_{w \in x} (t_w^{x,y} - \hat{t}_w^x)^2 \tag{4}$$

Note that tokens with negative term weight are recognized as insignificant thus discarded in the following.

3.2.1 Adaptation For Chinese

BERT-Scorer estimates weights for word-level terms while existing pre-trained BERT-style models for Chinese are character-level. To bridge the gap, we evenly distribute the weight of a word to each character in-between. Besides, we utilize the position information where character lies within the word by tagging each character via the widely-used BMES (Begin, Middle, End, and Single) schema and incorporating BMES embedding into BERT's input representation.

3.3 Label Distribution Generation

After BERT-Scorer generates term weights for query x_i and document y_i respectively, we calculate the label distribution based on their term weights of overlapped words as follows:

$$d_i = \tanh\left(\frac{1}{|\{x_i \cap y_i\}|} \sum_{w \in \{x_i \cap y_i\}} \hat{t}_w^{y_i} \hat{t}_w^{x_i}\right) \tag{5}$$

3.4 *Retriever* Models Utilizing Label Enhancement

We exploit a two-tower BERT-style *Retriever* model in this paper, as Figure 3 (b) shows. Each tower of our *Retriever* model exactly follows the architecture and hyper-parameters of the 12 layers BERT model[1], except the sequence length is set to be 64. An average-pooling operation is adopted on the output of BERT to produce the final representation for query and document ($\mathbf{u}$ and $\mathbf{v}$ respectively). Finally, the output score f is calculated by the cosine distance between $\mathbf{u}$ and $\mathbf{v}$ as follows:

$$f(x_i, y_i) = \frac{1}{2}\left(1 - \frac{\mathbf{u} \cdot \mathbf{v}}{||\mathbf{u}|| \times ||\mathbf{v}||}\right) \tag{6}$$

We incorporate the generated label distributions into the *Retriever* model as auxiliary supervision information. Given the training data with both hard labels and label distributions as follows:

$$X_i = \{(\langle x_i, y_i \rangle, d_i, l_i)\}_{i=1}^N \tag{7}$$

The model parameters are estimated by minimizing the following loss function:

$$\begin{aligned} L = \sum_{i=1}^N (\alpha(f(x_i, y_i) + d_i - 1)^2 \\ + (1-\alpha)(f(x_i, y_i) + l_i - 1)^2) \end{aligned} \tag{8}$$

where $\alpha \in [0, 1]$ denotes the loss weight of label distribution, which is used as a trade-off to get a suitable fitting target.

4 Experiment Settings

4.1 Datasets

Following Chang et al. (2020), we consider the Retrieval Question-Answering (ReQA) benchmark proposed by Ahmad et al. (2019). We use SQuAD (Rajpurkar et al., 2016) and Natural Questions (Kwiatkowski et al., 2019) for English, and CMRC 2018 (Cui et al., 2019) and DRCD (Shao et al., 2018) for Chinese. Note that Ahmad et al. (2019) is targetting at *Ranker*, while our goal is to improve the *Retriever*. Therefore our approaches are not directly comparable to the results presented in their paper.

Each entry of QA datasets is a tuple (q, a, p), where q is the question, a is the answer span, and p is the evidence passage containing a. Following Ahmad et al. (2019); Liu et al. (2021), we split a passage into sentences $p = s_1 s_2 ... s_n$. For a query q, we need to retrieve the correct sentence from a candidate set consisting of sentences of all passages. A query-sentence pair (q, s) is labeled as 1 if s is the sentence containing the corresponding answer span, and labeled as 0 otherwise. This problem is more challenging than retrieving the evidence passage only since the larger number of candidates to be retrieved.

For each dataset, the training/test split of the data is 60%/20%, and the 20% of the training set is held out as the validation set for hyper-parameter tuning[2]. We apply four-fold cross-validation to do significant tests.

[1] https://github.com/google-research/bert

[2] Note that all of our LE methods are only used in the training set, and we split the dataset according to questions so that there are no same questions in the training, validation and test set.

Dataset	Model	R@1	R@10	R@50	R@100
SQuAD	TF-IDF	40.24	62.01	71.09	75.81
	BM25	41.33	63.42	72.45	76.27
	F-EBR	21.78	45.43	66.30	71.72
	SBERT	35.27	48.48	68.21	78.85
	LE-BS	**48.94**	**64.62**	**80.21**	**86.03**
Natural Questions	TF-IDF	5.07	16.58	24.48	26.99
	BM25	5.02	16.33	24.11	26.74
	F-EBR	13.22	36.84	53.48	59.03
	SBERT	20.02	44.69	58.40	69.42
	LE-BS	**23.62**	**56.23**	**73.38**	**78.31**
CMRC	TF-IDF	50.88	71.23	77.29	81.58
	BM25	50.82	71.07	77.21	81.44
	F-EBR	52.60	72.33	79.03	84.14
	SBERT	52.89	73.34	81.90	85.72
	LE-BS	**63.60**	**82.71**	**90.09**	**94.48**
DRCD	TF-IDF	3.12	37.13	45.29	52.18
	BM25	3.48	37.81	46.13	53.38
	F-EBR	4.01	39.38	47.01	54.54
	SBERT	4.07	40.23	50.56	62.43
	LE-BS	**4.59**	**52.49**	**65.00**	**67.87**

Table 1: Experimental results of TF-IDF, BM25, F-EBR, SBERT, and LE-BS, where R@K represents Recall@K. Numbers are in percentage (%).

4.2 Baselines

We compare our method against the following six baselines. The first four are existing widely used large-scale *Retriever* models, and the latter two are models equipped with alternative label enhancement methods.

- **TF-IDF** and **BM25** are two widely used term weighting methods (Spärck Jones, 1972, 2004; Robertson and Zaragoza, 2009).

- **F-EBR** is the most widely used word-embedding-based multi-tower *Retriever* model proposed by Facebook Search (Huang and Sharma, 2020).

- **SBERT** is a competitive BERT-based multi-tower *Retriever* proposed by Reimers and Gurevych (2019).

- **LE-TFIDF** is a variant of our method in which the label distribution is generated based on static TF-IDF weights.

- **LE-Distill** is another variant in which the label distribution set as predicting scores of SBERT. This method is similar to self-distillation process in born-again networks (Furlanello et al., 2018).

For the convenience of comparison, we refer to our **L**abel **E**nhancement method based on **B**ERT **S**corer as **LE-BS**.

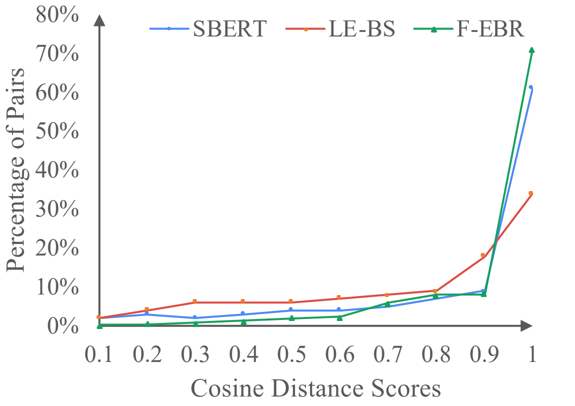


Figure 5: Comparison of the prediction score distribution.

4.3 Evaluation Metric

Since the goal of the retrieval phase is to capture the positives in the top-k results, we select **Recall@k** as the evaluation metric. Recall@k is computed by the following equation:

$$Recall@k = \frac{1}{|D|} \sum_{x_i \in D} \frac{\sum_{y_i \in R_k} l_{<x_i,y_i>}}{\sum_{y_i \in D} l_{<x_i,y_i>}} \qquad (9)$$

where R_k is the top k results recalled by our model. D is the dataset. x_i and y_i are the i-th query and i-th document separately.

5 Experiment Results

5.1 Comparison with *Retriever* Models

The experimental results[3] are shown in the Table 1, from which we have three observations:

1. Term weighting methods perform exceptionally well for the SQuAD benchmark, as the data collection process and human annotations of this dataset are biased towards question-answer pairs with overlapping tokens. They perform poorly in the Natural Questions dataset, where there are fewer overlapping tokens and the embedding-based model perform well. Our LE-BS combines the advantage of term weighting and embedding-based methods to perform well in all datasets.

2. It is as expected that LE-BS and SBERT outperform F-EBR by a large margin since pre-trained language models yield much more robust representation than word embeddings.

[3] The experiment results in this paper are statistically significant with $p < 0.05$.

Dataset	Model	R@1	R@10	R@50	R@100
Standard					
SQuAD	LE-None	35.27	48.48	68.21	78.85
	LE-Distill	37.62	55.89	72.54	80.50
	LE-TFIDF	41.03	61.29	78.45	83.89
	LE-BS	**48.94**	**64.62**	**80.21**	**86.03**
Natural Questions	LE-None	20.02	44.69	58.40	69.42
	LE-Distill	21.71	48.37	62.60	72.18
	LE-TFIDF	21.80	48.75	68.43	74.03
	LE-BS	**23.62**	**56.23**	**73.38**	**78.31**
Cold-start					
SQuAD	LE-None	6.20	9.85	16.69	21.35
	LE-Distill	6.35	9.97	17.24	21.66
	LE-TFIDF	7.11	10.90	17.93	22.51
	LE-BS	**11.80**	**14.42**	**19.77**	**24.85**
Natural Questions	LE-None	4.68	5.90	6.45	6.80
	LE-Distill	5.11	6.44	7.34	8.69
	LE-TFIDF	5.20	6.96	8.30	10.90
	LE-BS	**7.21**	**8.78**	**11.60**	**14.08**

Table 2: Experimental results of different LE method.

3. LE-BS further achieves significant improvement over SBERT. LE-BS can be viewed as an enhanced SBERT variant that incorporates label enhancement. We could observe the improvement of LE-BS over SBERT on both English and Chinese datasets, verifying that the label distributions generated by our BERT-Scorer provide helpful supervision signals for *Retriever* models in a language-independent manner.

5.2 Impact of Label Distribution

We further investigate why label distribution can bring recall improvement observed above. We take the SQuAD dataset as an example and get all predicting distance scores of testing pairs. We split the range of [0,1] into ten equal sub-ranges including (0, 0.1], (0.1, 0.2],..., and (0.9, 1], and count proportions of pairs whose scores are in each sub-range. The three multi-tower models' statistics are shown in Figure 5.

From Figure 5, we find the distance scores of most testing pairs are close to 1. It is a natural result since most testing pairs are labeled as irrelevant by hard labels. Compared with F-EBR and SBERT, the curve of LE-BS is much smoother, meaning more pairs have a smaller query-document distance. We attribute this to the supplementary training objective of fitting the label distribution in addition to the 0-1 hard label. The trend of LE-BS's curve partly expresses why LE-BS achieves much better recall scores. In other words, we can safely conclude that with label distribution LE-BS can identify more relevant candidates without introducing too many false positives. Note that better recall is a fundamental goal of *Retriever* because we want to feed *Ranker* with as many relevant candidates as possible.

Dataset	Model	R@1	R@10	R@50	R@100
Standard					
SQuAD	$\alpha = 0$	24.69	47.11	68.39	74.70
	$\alpha = 0.2$	28.35	59.89	72.77	80.99
	$\alpha = 0.5$	30.09	62.52	76.80	81.16
	$\alpha = 0.8$	31.47	63.84	78.83	83.26
	$\alpha = 1$	**48.94**	**64.62**	**80.21**	**86.03**
Natural Questions	$\alpha = 0$	21.13	44.97	53.29	68.78
	$\alpha = 0.2$	**23.62**	**56.23**	**73.38**	**78.31**
	$\alpha = 0.5$	22.86	51.97	72.47	77.60
	$\alpha = 0.8$	21.05	52.53	72.04	77.65
	$\alpha = 1$	22.11	53.55	71.84	76.80
Cold-start					
SQuAD	$\alpha = 0$	5.87	9.79	16.86	20.54
	$\alpha = 0.2$	7.58	8.84	16.05	20.51
	$\alpha = 0.5$	7.33	10.00	16.65	21.93
	$\alpha = 0.8$	8.86	11.70	16.76	23.06
	$\alpha = 1$	**11.80**	**14.42**	**19.77**	**24.85**
Natural Questions	$\alpha = 0$	3.79	5.35	6.51	7.45
	$\alpha = 0.2$	5.10	6.85	8.82	9.78
	$\alpha = 0.5$	5.00	6.06	9.55	9.72
	$\alpha = 0.8$	5.59	7.78	9.40	11.40
	$\alpha = 1$	**7.21**	**8.78**	**11.60**	**14.08**

Table 3: Effect of different weights of label distribution.

5.3 Analysis of Label Enhancement Method

The intuition of our label enhancement method in retrieval scenarios is to incorporate prior knowledge from static term weighting methods into dynamic contextual embeddings. To verify the superiority of our label enhancement method, we compare two alternative label enhancement techniques. The empirical results are demonstrated in Table 2. For the convenience and clarity of comparison, here we also put the performance of SBERT. Its experimental results are demonstrated as **LE-None** to indicate that no LE method is employed.

To further analyze the effectiveness of label enhancement, we consider two different settings for each dataset. The first one is the **standard-setting**, where the training/test split of the data is 60%/20%, and the 20% of the training set is held out as the validation set. The second one is the **cold-start setting** that assumes there are not enough training data to use. The only difference from the standard-setting is that the training/test split of the data is 20%/60%. We have the following five observations:

1. All LE-based models outperform the LE-None model, which clearly verifies the effectiveness of label distribution for the retrieval task.

2. The improvement of LE-TFIDF over LE-None shows that static TF-IDF weights serve as beneficial prior knowledge to characterize label distribution.

3. LE-Distill also achieves notable enhancements. This observation is consistent with other knowledge distillation works (Hinton et al., 2015; Furlanello et al., 2018). The self-distillation process brings valuable dark knowledge via the generated soft predicting scores even without utilizing TF-IDF information.

4. Relative performance improvement brought by LE under the cold-start setting is more evident than the standard-setting. The possible reason is that relevance degree information could play a more important role when there are not enough training data. This observation is also consistent with other data-lacking scenarios of using label distribution (e.g., knowledge distillation (Hinton et al., 2015)).

5. Our LE-BS has clear superiority over LE-Distill and LE-TFIDF among all datasets under both the standard and cold-start settings. Rather than predicting relevance score directly as LE-Distill, LE-BS predicts dynamic term weights by BERT-Scorer in a way incorporating useful TF-IDF information into contextual BERT representation. Therefore, the final generated label distribution integrates explicit prior TF-IDF knowledge, and some helpful "dark" knowledge (Furlanello et al., 2018) is produced during the training step. We believe that is the main reason behind this superiority of our method.

5.4 Collaboration between Label Distribution and Hard Label

As a critical hyper-parameter of our LE-BS method, α denotes how to weight the optimization objectives of hard labels and label distributions. This section investigates the collaboration between hard labels and label distributions with different α settings. This analysis could provide more systematic guidance on how to incorporate label distribution.

We train our LE-BS with α is set to 0, 0.2, 0.5, 0.8, and 1, respectively, and report the empirical results of the SQuAD and Natural Questions datasets. Note that setting α as 0 means using only hard labels, and setting α as 1 means using only label distributions. The experimental results are shown in Table 3, from which we find tuning α is essential – different α can result in recall variation of $5\% - 10\%$.

For the standard-setting, we find that when α is set to be larger, our LE-BS performs exceptionally well for the SQuAD benchmark. Note that the data collection process and human annotations of SQuAD are biased towards question-answer pairs with overlapping tokens (Rajpurkar et al., 2016). We can naturally expect that the generated label distribution could better characterize query-document relevance degree in the SQuAD dataset due to the capability of BERT-Scorer to identify overlapped highly-representative tokens. Regarding the Natural Question dataset, LE-BS is best performed when the α is set as 0.2. This dataset is built based on Google search logs, so the connection between queries and document are more challenging to capture. In this scenario, if we rely too much on the supervision signal from the generated label distributions, unreasonable noisy information can be brought in and thereby hinders model performance.

For the cold-start setting, models with a larger α consistently achieve better performance. In such data-lacking scenarios, models cannot get sufficient supervision information from training sets' hard labels. When α becomes larger, more auxiliary supervision information from the label distribution could be utilized. Though this is a rather rough explanation for this observation, it can serve as trustworthy guidance in practice for information retrieval researchers and engineers.

6 Conclusion

This paper first introduced label distribution to characterize the relevance degree between queries and documents in large-scale retrieval problems. Then we designed a novel and effective label enhancement method that generates label distributions via fusing context-free TF-IDF information and contextual BERT representation. An improved *Retriever* model was achieved easily by incorporating the generated label distributions as auxiliary supervision information. Our method's superiority can be observed on four datasets of English and Chinese.

Acknowledgments

We thank anonymous reviewers for valuable comments. This research was supported by the National Key Research And Development Program of China (No.2019YFB1405802) and the Central Government Guiding Local Science And Technology Development Foundation Projects (Science and Technology Innovation Base Projects) No. 206Z0302G.

References

Amin Ahmad, Noah Constant, Yinfei Yang, and Daniel Cer. 2019. Reqa: An evaluation for end-to-end answer retrieval models. In *MRQA@EMNLP*, pages 137–146.

Wei-Cheng Chang, Felix X. Yu, Yin-Wen Chang, Yiming Yang, and Sanjiv Kumar. 2020. Pre-training tasks for embedding-based large-scale retrieval. In *ICLR*. OpenReview.net.

Yiming Cui, Ting Liu, Wanxiang Che, Li Xiao, Zhipeng Chen, Wentao Ma, Shijin Wang, and Guoping Hu. 2019. A span-extraction dataset for Chinese machine reading comprehension. In *EMNLP-IJCNLP*, pages 5886–5891, Hong Kong, China.

Zhuyun Dai and Jamie Callan. 2019. Context-aware sentence/passage term importance estimation for first stage retrieval. *CoRR*, abs/1910.10687.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Tommaso Furlanello, Zachary Chase Lipton, Michael Tschannen, Laurent Itti, and Anima Anandkumar. 2018. Born-again neural networks. In *ICML*, volume 80 of *Proceedings of Machine Learning Research*, pages 1602–1611. PMLR.

Yongbiao Gao, Yu Zhang, and Xin Geng. 2020. Label enhancement for label distribution learning via prior knowledge. In *IJCAI*, pages 3223–3229. ijcai.org.

Xin Geng. 2016. Label distribution learning. *IEEE Transactions on Knowledge and Data Engineering*, 28(7):1734–1748.

Ruiqi Guo, Sanjiv Kumar, Krzysztof Choromanski, and David Simcha. 2016. Quantization based fast inner product search. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics, AISTATS 2016, Cadiz, Spain, May 9-11, 2016*, volume 51 of *JMLR Workshop and Conference Proceedings*, pages 482–490. JMLR.org.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *stat*, 1050:9.

Jui-Ting Huang and Ashish Sharma. 2020. Embedding-based retrieval in facebook search. In *SIGKDD*, pages 2553–2561. ACM.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2017. Billion-scale similarity search with gpus. *CoRR*, abs/1702.08734.

Tom Kwiatkowski, Jennimaria Palomaki, and Olivia Redfield. 2019. Natural questions: a benchmark for question answering research. *Trans. Assoc. Comput. Linguistics*, 7:452–466.

Peiyang Liu, Sen Wang, Xi Wang, Wei Ye, and Shikun Zhang. 2021. QuadrupletBERT: An efficient model for embedding-based large-scale retrieval. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3734–3739, Online. Association for Computational Linguistics.

Peiyang Liu, Wei Ye, Xiangyu Xi, Tong Wang, Jinglei Zhang, and Shikun Zhang. 2020. Not all synonyms are created equal: Incorporating similarity of synonyms to enhance word embeddings. In *2020 International Joint Conference on Neural Networks, IJCNN 2020, Glasgow, United Kingdom, July 19-24, 2020*, pages 1–8. IEEE.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Wenhao Lu, Jian Jiao, and Ruofei Zhang. 2020. Twinbert: Distilling knowledge to twin-structured compressed BERT models for large-scale retrieval. In *CIKM*, pages 2645–2652. ACM.

Jiaqi Lv, Ning Xu, RenYi Zheng, and Xin Geng. 2019. Weakly supervised multi-label learning via label enhancement. In *IJCAI*, pages 3101–3107. ijcai.org.

Yosi Mass and Haggai Roitman. 2020. Ad-hoc document retrieval using weak-supervision with BERT and GPT2. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4191–4197, Online. Association for Computational Linguistics.

Tomás Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. In *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information*

Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States, pages 3111–3119.

Ramith Padaki, Zhuyun Dai, and Jamie Callan. 2020. Rethinking query expansion for BERT reranking. In *ECIR*, volume 12036 of *Lecture Notes in Computer Science*, pages 297–304. Springer.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1532–1543. ACL.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100, 000+ questions for machine comprehension of text. In *EMNLP*, pages 2383–2392.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 3980–3990. Association for Computational Linguistics.

Stephen E. Robertson and Hugo Zaragoza. 2009. The probabilistic relevance framework: BM25 and beyond. *Found. Trends Inf. Retr.*, 3(4):333–389.

Chih-Chieh Shao, Trois Liu, Yuting Lai, Yiying Tseng, and Sam Tsai. 2018. DRCD: a chinese machine reading comprehension dataset. *CoRR*, abs/1806.00920.

Anshumali Shrivastava and Ping Li. 2014. Asymmetric LSH (ALSH) for sublinear time maximum inner product search (MIPS). In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 2321–2329.

Karen Spärck Jones. 1972. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 28(1):11–21.

Karen Spärck Jones. 2004. Idf term weighting and ir research lessons. *Journal of documentation*, 60(5):521–523.

Ashish Vaswani and Noam Shazeer. 2017. Attention is all you need. In *NeurIPS*, pages 5998–6008.

Ning Xu, Yun-Peng Liu, and Xin Geng. 2019. Label enhancement for label distribution learning. *IEEE Transactions on Knowledge and Data Engineering*.

Ning Xu, Jun Shu, Yun-Peng Liu, and Xin Geng. 2020. Variational label enhancement. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 10597–10606. PMLR.

Linfeng Zhang, Jiebo Song, Anni Gao, Jingwei Chen, Chenglong Bao, and Kaisheng Ma. 2019. Be your own teacher: Improve the performance of convolutional neural networks via self distillation. In *ICCV*, pages 3712–3721. IEEE.