Ensemble Distillation for BERT-Based Ranking Models

Honglei Zhuang
hlz@google.com
Google Research
USA

Zhen Qin
zhenqin@google.com
Google Research
USA

Shuguang Han*
hanshuguang@gmal.com
Alibaba
China

Xuanhui Wang
xuanhui@google.com
Google Research
USA

Michael Bendersky
bemike@google.com
Google Research
USA

Marc Najork
najork@google.com
Google Research
USA

ABSTRACT

Over the past two years, large pretrained language models such as BERT have been applied to text ranking problems and showed superior performance on multiple public benchmark data sets. Prior work demonstrated that an ensemble of multiple BERT-based ranking models can not only boost the performance, but also reduce the performance variance. However, an ensemble of models is more costly because it needs computing resource and/or inference time proportional to the number of models. In this paper, we study how to retain the performance of an ensemble of models at the inference cost of a single model by distilling the ensemble into a single BERT-based student ranking model. Specifically, we study different designs of teacher labels, various distillation strategies, as well as multiple distillation losses tailored for ranking problems. We conduct experiments on the MS MARCO passage ranking and the TREC-COVID data set. Our results show that even with these simple distillation techniques, the distilled model can effectively retain the performance gain of the ensemble of multiple models. More interestingly, the performances of distilled models are also more stable than models fine-tuned on original labeled data. The results reveal a promising direction to capitalize on the gains achieved by an ensemble of BERT-based ranking models.

CCS CONCEPTS

• **Information systems** → **Retrieval models and ranking**.

KEYWORDS

Ensemble distillation; ranker ensemble; BERT

ACM Reference Format:
Honglei Zhuang, Zhen Qin, Shuguang Han, Xuanhui Wang, Michael Bendersky, and Marc Najork. 2021. Ensemble Distillation for BERT-Based Ranking Models. In *Proceedings of the 2021 ACM SIGIR International Conference on the Theory of Information Retrieval (ICTIR '21), July 11, 2021, Virtual Event, Canada.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3471158.3472238

1 INTRODUCTION

With the availability of large-scale text-based data sets like MS MARCO [1] for ranking problems, much attention has been devoted towards developing effective ranking models that take raw query and document texts as inputs directly, compared to traditional learning to rank with only derived numerical inputs [20]. One of the most popular approaches is to use a large language model such as BERT [7] as encoder and then fine-tune it for ranking problems [12, 17]. The remarkable performance of BERT-based ranking models makes them the state-of-the-art text ranking models.

At the same time, the fine-tuned BERT models show high variances since the performances of different runs of the same BERT model could be statistically significantly different from each other [12]. A simple ensemble of these models not only achieves better results, but also reduces the performance variances. This has been confirmed by existing work on public text ranking benchmarks such as MS MARCO [12] and TREC-COVID [2].

However, serving an ensemble model is always costly as the computational cost required would be multiplied by the number of models. This issue is worsened for the ensemble of multiple BERT-based models due to their gigantic sizes. Serving even a single BERT-based model online would require non-trivial effort to address [15, 21]. Hence, it would be prohibitively expensive to directly serve a text ranking model ensemble containing multiple BERT-based ranking models for most online products.

In this paper, we study how to retain the advantages of an ensemble of models but reduce their cost by exploring how to distill them into a single BERT-based student ranking model. For neural networks, Knowledge Distillation (KD) [13] was proposed recently to distill a large and complex teacher model into a small student one. It has been widely studied for different variants of classification [11], but has not been well studied for ranking problems. Existing work on distilling ranking models mainly focuses on using pointwise regression losses on a single BERT model [10]. There is little work studying how to distill an ensemble of multiple *ranking* models. It is unclear what the best way is to jointly utilize the outputs of multiple teacher ranking models in the distillation process and whether a ranking-specific listwise loss is effective in this setting.

The contributions of this paper can be summarized as follows:

- We evaluate a few methods to derive teacher labels from a set of BERT-based ranking models, including using their ranking scores directly and their derived reciprocal ranks [6].
- We study KD with various strategies, including different ways to jointly leverage teacher labels from multiple teacher

[*] Work done while at Google Research.

ICTIR '21, July 11, 2021, Virtual Event, Canada.
© 2021 Copyright held by the owner/author(s).
ACM ISBN 978-1-4503-8611-1/21/07.
https://doi.org/10.1145/3471158.3472238

models and different distillation losses. We show that the listwise softmax cross entropy loss produces strong and robust performances with a number of distillation strategies.

- In contrast to the existing KD work, we find that the distilled BERT-based ranking models have little accuracy loss compared with the ensemble model. Meanwhile, the distilled model shows higher stability compared with a BERT ranking model trained directly with the original labels.

The rest of the paper is organized as follows. We review the related work in Section 2. The preliminaries are described in Section 3 and our methods are described in Section 4. We then present our experiments on the benchmark data sets in Section 5 and conclude the paper in Section 6.

2 RELATED WORK

Ensemble [8] is a widely-adopted machine learning technique. The technique takes a set of trained base models and aggregates their predictions as the ensemble predictions. The set of base models can contain multiple learning algorithms, different parameterizations of the same model, or even multiple instances of the same model trained with random parameter initializations. The ensemble model reduces the variances of individual models and often improves the overall performance. Ensemble is particularly helpful in aggregating BERT-based models as they tend to have higher variances potentially due to their gigantic size [2, 12].

Knowledge Distillation (KD), introduced in [13], is to transfer knowledge from a teacher model to a student model. Given a larger teacher model, a smaller student model can be trained using the teacher predictions as labels. The teacher model can be a model with the same architecture as the student model but larger capacity or an ensemble of multiple models with each having the same capacity as the student [24–26]. Recently, Born Again Networks (BAN) [9] used identical teacher and student models and found that the student model can outperform the teacher model. All these studies are on classification problems and the output of a teacher model is intuitively meaningful as the probability distribution over possible classes.

BERT [7] has been quite powerful for many different tasks. Recently, it has been introduced for ranking tasks [5, 12, 17]. Nogueira and Cho [17] solved the passage re-ranking task by fine-tuning BERT with a pointwise regression loss. Han et al. [12] used listwise ranking losses to fine-tune BERT and also showed that an ensemble is effective to improve the ranking accuracy, but they did not study ensemble distillation. Gao et al. [10] combined different objectives in the distillation losses for BERT rankers. However, the ranking objective is the same as [17] – a pointwise regression loss. Similarly, Chen et al. [5] studied KD on TinyBERT for document retrieval and Yang et al. [25] proposed an *m-o-m* method to distill m smaller student BERT models from m large teacher BERT models in the application of web question answering. Both also used pointwise regression losses. None of them studied more advanced ranking losses or the distillation of ensemble models. Our focus is on the distillation of a BERT ensemble with ranking losses.

Applying KD to ranking problems has not been extensively studied and only a couple of works are in the literature. For example, Ranking Distillation (RD) [23] was proposed for recommender systems. In RD, the teacher model is trained with a pairwise ranking loss, but the distillation loss is a pointwise regression loss by treating the top-k *unlabeled* documents ranked by the teacher model for training queries as *additional* positives. Zhang et al. [28] followed a similar setting but applied the teacher model to *holdout* queries and their documents as training data for distillation. Both of them need extra data and it is unclear how effective ranking distillation can be without it.

3 PRELIMINARIES

In this section, we briefly introduce the notations and the BERT-based ranking models used in our studies.

3.1 Text Ranking

We represent each query as $(q, D, \mathbf{y})$, where q represents the query text; $D = (d_1, \ldots, d_m)$ is a set of candidate text retrieved by the query text q; $\mathbf{y} = (y_1, \ldots, y_m)$ are real-valued labels where y_j indicates the relevance of d_j to the query q. Each candidate text d_j can be represented as a sequence of tokens $d_j = (w_{j1}, \ldots, w_{jL_j})$. Each query can also be regarded as a sequence of tokens similarly. Labels $\mathbf{y}$ are only given for training data.

In a supervised learning-to-rank setting, a training data set $\mathcal{D}_{\text{train}} = \{(q_i, D_i, \mathbf{y}_i)\}_1^n$ is given. The objective is to learn a scoring function f, such that for any test data set $\mathcal{D}_{\text{test}} = \{(q_i, D_i)\}_1^{n'}$, the scoring function can output ranking scores $\hat{y}_{ij} = f(q_i, d_{ij}) \in \mathbb{R}$ for each candidate text $d_{ij} \in D_i$ in each query q_i. Ideally, by sorting candidates in D_i for each q_i based on their predicted ranking scores $\hat{y}_{ij}$, we can obtain a ranking where more relevant candidates d_{ij} are ranked higher.

3.2 BERT Ranker

There have been many studies [14, 16, 18, 27] on leveraging the power of BERT [7] to conduct text ranking tasks. In this study, we adopt a methodology proposed in [12].

Specifically, for each query-document pair (q_i, d_{ij}), we concatenate the sequence of query text q_i and candidate text d_{ij}. We then feed the concatenated sequence into a BERT model and pass the output `[CLS]` vector through a single dense layer to obtain the ranking score $\hat{y}_{ij}$.

We initialize the parameters in the BERT model from a pretrained checkpoint. Then we fine-tune the entire model by a ranking loss which takes into account all the documents in candidate set D_i. For each query q_i, denote all the ground-truth labels as $\mathbf{y}_i$ and all the predicted scores as $\hat{\mathbf{y}}_i$, we can define a listwise softmax cross entropy loss [3], which is a simple version of ListNet [4], as:

$$\ell_{\text{softmax}}(\mathbf{y}_i, \hat{\mathbf{y}}_i) = \sum_{j=1}^{m} y_{ij} \log \left(\frac{\exp(\hat{y}_{ij})}{\sum_{j'=1}^{m} \exp(\hat{y}_{ij'})} \right) \qquad (1)$$

Notice that by training with the listwise softmax cross entropy loss, the predicted rankings score $\hat{y}_{ij}$ is merely a ranking score without any pre-determined semantics. It is not guaranteed that the ranking score would have roughly the same scale as the original ground-truth label. We choose this loss as it is shown more effective than the pointwise regression loss [12].

3.3 Ranker Ensemble

When there are multiple trained learning-to-rank models available, one can leverage the power of all the models by deriving an ensemble model. We recall two simple ensemble methods for ranking models.

Mean score. Suppose that there are K base rankers. For each query q_i and its candidate text set D_i, the ranking score of the j-th candidate text d_{ij} predicted by the k-th ranker can be represented as $\hat{y}_{ij}^{(k)}$. The most naïve method to obtain the ensemble ranking score s_{ij} is to simply take the mean of all the base rankers' scores:

$$s_{ij} = \frac{1}{K} \sum_{k=1}^{K} \hat{y}_{ij}^{(k)} \tag{2}$$

Reciprocal rank fusion (RRF). Another widely adopted ensemble method is reciprocal rank fusion (RRF) [6]. Instead of directly using the ranking scores from base rankers, we utilize the rank of each candidate based on the predicted ranking scores. For each query q_i, we denote the predicted *rank* of the j-th candidate text by the k-th ranker as $\hat{r}_{ij}^{(k)}$. The RRF-based ensemble ranking score is:

$$s_{ij} = \frac{1}{K} \sum_{k=1}^{K} \frac{1}{C + \hat{r}_{ij}^{(k)}} \tag{3}$$

where C is a user-defined constant. RRF is favored when the base rankers are trained with very different models as the ensemble derivation is agnostic to scale of the ranking scores.

4 ENSEMBLE DISTILLATION

An ensemble of multiple BERT rankers often provides further improvement over simply adopting a single BERT ranker, but serving such an ensemble can be prohibitive due to the size of BERT models. We thus propose our ensemble distillation in this section.

4.1 Teacher Labels

Recall that for query q_i and its j-th candidate text d_{ij}, the predicted ranking score by the k-th ranker is represented as $\hat{y}_{ij}^{(k)}$. We denote the correspondingly transformed teacher label as $s_{ij}^{(k)}$.

Ranking score. The simplest teacher label is to directly use the predicted ranking score, namely

$$s_{ij}^{(k)} = \hat{y}_{ij}^{(k)} \tag{4}$$

Reciprocal rank. Teacher models can have outputs with very different scales, which would make it difficult for the student models to fit. Similar to the RRF ensemble method, we use the reciprocal rank to be the teacher label:

$$s_{ij}^{(k)} = \frac{1}{C + \hat{r}_{ij}^{(k)}} \tag{5}$$

where C is the same constant as defined in Equation (3).

4.2 Distillation Strategies

We study two strategies to leverage outputs from multiple teacher models.

Aggregated teacher label distillation (AGG). A straightforward method is to first aggregate all the K teacher labels into an aggregated teacher label, then distill the student model from the aggregated labels. Specifically, for query q_i and candidate d_{ij}, we denote the aggregated teacher label as s_{ij} and use the mean of all the K teacher labels as the aggregated teacher label for simplicity:

$$s_{ij} = \frac{1}{K} \sum_{k=1}^{K} s_{ij}^{(k)} \tag{6}$$

Notice that this is equivalent to directly using the ensemble ranker scores as the aggregated teacher labels: if the teacher labels are simply ranking scores, the aggregated teacher labels equal to the ensemble scores in Equation (2); if the teacher labels are reciprocal ranks, the aggregated teacher labels equal to the ensemble scores in Equation (3).

For all the candidates D_i of query q_i, we denote all of their aggregated teacher scores as $\mathbf{s}_i$ and all of their distilled scores as $\hat{\mathbf{s}}_i$ (and $\hat{s}_{ij}$ for d_{ij}). We can fine-tune the student BERT model by minimizing the loss function:

$$\ell_{\text{distill-agg}}(\hat{\mathbf{s}}_i) = \ell(\mathbf{s}_i, \hat{\mathbf{s}}_i) \tag{7}$$

where ℓ is a ranking loss function defined based on a list of ground-truth labels and predicted scores. The concrete definition of ℓ can be found in Section 4.3.

Multi-objective distillation (MO). Another method is to treat the distillation as a multi-objective learning problem. Instead of aggregating the teacher labels of all the K base rankers, we can fine-tune the student BERT model by optimizing K loss functions simultaneously. We use $\mathbf{s}_i^{(k)}$ to represent all the teacher labels from the k-th base rankers for query q_i. The training loss used for multi-objective distillation is:

$$\ell_{\text{distill-mo}}(\hat{\mathbf{s}}_i) = \frac{1}{K} \sum_{k=1}^{K} \ell\left(\mathbf{s}_i^{(k)}, \hat{\mathbf{s}}_i\right) \tag{8}$$

Again, the instantiation of ℓ is described in Section 4.3.

4.3 Distillation Loss Function

The distillation loss function $\ell(\cdot, \cdot)$ can be instantiated by any ranking loss functions.

Softmax cross entropy. The first instantiation we explore is the softmax cross entropy loss function. We distill the teacher models by minimizing $\ell_{\text{softmax}}(\mathbf{s}_i, \hat{\mathbf{s}}_i)$ where $\ell_{\text{softmax}}(\cdot, \cdot)$ is defined in Equation (1). The listwise ranking loss emphasizes fitting the ranking obtained by the base rankers. The loss penalizes the model more when top-ranked items are incorrect.

Mean squared error. An alternative distillation loss function is a pointwise loss function which emphasizes fitting the predicted scores of each candidate text from the base rankers, regardless of their ranking. We explore using the mean squared error (MSE) loss to instantiate the distillation loss:

$$\ell_{\text{MSE}}(\mathbf{s}_i, \hat{\mathbf{s}}_i) = \sum_{j=1}^{m} (s_{ij} - \hat{s}_{ij})^2 \tag{9}$$

4.4 Leveraging Ground-Truth Labels

In addition, we can also leverage the ground-truth labels to further regularize the distillation process. We can fine-tune the student BERT model by optimizing a mixture loss function:

$$\ell_{\text{mixture}}(\hat{\mathbf{s}}_i) = \alpha \cdot \ell_{\text{distill}}(\hat{\mathbf{s}}_i) + (1-\alpha) \cdot \ell(\mathbf{y}_i, \hat{\mathbf{s}}_i) \tag{10}$$

where α is a user-defined constant between 0 and 1. The first term $\ell_{\text{distill}}(\cdot)$ can be instantiated by either $\ell_{\text{distill-agg}}(\cdot)$ or $\ell_{\text{distill-mo}}(\cdot)$. The second term $\ell(\cdot,\cdot)$ is the ranking loss between the predicted ranking scores and the ground-truth labels. In our experiments, we utilize the ranking loss in base ranking model training, which is a softmax cross entropy loss as described in Equation (1).

5 EXPERIMENTS

In this section, we describe the experiments we conduct to verify the effectiveness of distillation for an ensemble of BERT rankers. All our experiments are conducted based on the TF-Ranking library [19].

5.1 Data Sets

MS MARCO. We utilize MS MARCO passage ranking data set [1] in our experiments. The task is to rank passages based on their relevance to questions. There are more than 530,000 questions in the "train" data partition, and the evaluation is usually performed on around 6,800 questions in the "dev" and "eval" data partition. The ground-truth labels for the "eval" partition are not published. The original data set contains more than 8.8 million passages. For more efficient experiments, we use a BERT-based ranker similar to the one described in [12] to retrieve the top-50 documents for each question in the "train" and "dev" partition as candidate text sets. We train the base rankers and infer their ranking scores for these top-50 documents in the "train" partition and distill student models from them. Then we infer the ranking scores of distilled student models for the top-50 documents in the "dev" partition. The evaluation metrics are calculated on the "dev" partition.

TREC-COVID. We also conduct experiments on TREC-COVID [22] data set. The data set contains 50 topics (queries) and a corpus of around 190,000 abstracts. Topics from Round 1-4 and about 90% of their relevance judgments are used for training, and their remaining relevance judgments and all the relevance judgements for Round 5 topics are held out for evaluation. We use retrieval methods described in [2] to retrieve a subset of candidate abstracts for each topic. Further, we use 1 relevant and 5 random irrelevant abstracts to create ranking lists for training.

5.2 Parameter Configurations

For all the base rankers and the distillation student model, we utilize BERT-Large pretrained with whole-word masking. During the fine-tuning, we set the learning rate as 1e-5 and the batch size as 32. Notice that the batch size indicates the number of queries in the batch. We train 10 base rankers for ensemble and distillation. For RRF, we try different constant parameters and eventually set $C = 0$ for MS MARCO and $C = 50$ for TREC-COVID.

Although all the base rankers and the distillation student model are identical BERT-Large models, the student model is still much easier to serve than the ensemble of 10 base rankers, because serving the ensemble model online would require 10x computing resources or 10x time.

Table 1: Overall performance comparison. The best performances are bolded. Distillation results with * are statistically significantly ($p \leq 0.05$) better than average performance of base rankers.

Methods	MS MARCO MRR@10	TREC-COVID MAP
Avg of Base Rankers	0.3995	0.2020
Best of Base Rankers	0.4026	0.2094
Mean Score Ensemble	0.4045*	0.2108*
RRF Ensemble	0.4045*	0.2104*
Ensemble Distillation	**0.4053***	**0.2138***

Table 2: Comparing different variants of distillation methods. The best performances are bolded. Distillation results with * are statistically significantly ($p \leq 0.05$) better than average performance of base rankers, and those with $^{\downarrow}$ are significantly worse than the ensemble model.

Teacher label	Methods	MS MARCO MRR@10	TREC-COVID MAP
Ranking Scores	AGG Softmax	0.4036*	0.2095*
	AGG MSE	0.4039*	0.2105*
	MO Softmax	**0.4053***	**0.2138***
	MO MSE	0.4023*	0.2102*
Reciprocal Rank	AGG Softmax	0.4041*	0.2117*
	AGG MSE	0.3903$^{\downarrow}$	0.0906$^{\downarrow}$
	MO Softmax	0.4007	0.2082
	MO MSE	0.3915$^{\downarrow}$	0.0855$^{\downarrow}$

We evaluate our models by the official metrics of both data sets. We use mean reciprocal rank (MRR@10) for MS MARCO and mean average precision (MAP) for TREC-COVID.

5.3 Results

Effectiveness of distilled models. First we examine whether the distilled student model remains as effective as the ensemble model. We calculate the average and the best performance of base rankers. The average performance reflects the expected performance when users train a single base ranker without further selection, while the best performance corresponds to the scenario when users can choose from a set of trained models. We also evaluate the performance of both ensemble rankers before distillation as the "upper-bound" for distilled rankers. We try distillation methods with different teacher label derivation methods, different distillation strategies and different loss functions and report the results from the best configuration, which utilizes the multi-objective distillation strategy with Softmax loss. The results are shown in Table 1.

We can observe that before distillation, both the mean score ensemble and the RRF ensemble outperform any single base ranker. More importantly, the distilled models successfully retain the advantage of the ensemble models. The best distilled student models on both data set achieve similar performances as the ensemble models, and are still better than any single base ranker.

Distillation configuration comparison. We also analyze the performances of the distilled student models with different configurations on both data sets in Table 2.

First, we can observe that using either ranking scores or reciprocal ranks as teacher labels produces a distilled student model with equivalent performance to that of the ensemble model. Using the original ranking scores as teacher labels seems to perform well with different losses and strategies. However, it is worth noting that our experiments use the same ranking models trained with different initialization as base rankers, and hence they have very similar ranking score distributions. These results may not generalize to base models with different model structures.

In addition, using aggregated teacher labels (AGG) for distillation or the multi-objective distillation strategy (MO) does not seem to have much difference in our experiments. Both strategies perform well when appropriate teacher labels and distillation losses are selected.

We also compare the listwise distillation loss (Softmax) and the pointwise distillation loss (MSE) on both data sets. It can be seen that the student models distilled with the listwise Softmax loss almost always remain similarly effective to the ensemble model, while the pointwise MSE loss does not perform well when reciprocal ranks are used as the teacher labels. A possible explanation is that the absolute differences between reciprocal ranks are too small for MSE loss to distinguish, especially when C is large.

Stability of distilled models. We further study the performance stability of distilling from the ensemble model. We perform the distillation process with "MO Softmax" (multi-objective distillation strategy with softmax distillation loss) for 5 times, and measure the performance of the 5 runs to see how much they vary from each other. We also fine-tune the base ranker for 5 times as a comparison. We visualize their performances on both data sets in Figure 1.

It can be observed that the variance of base ranker performances is extremely large. On the MS MARCO data set the lower end of the 95% confidence interval reaches below 0.3990 in terms of MRR@10, and on the TREC-COVID data set the best and the worst performances of base rankers can be about 0.01 in terms of MAP. This suggests that, with a fairly high chance, simply fine-tuning a single BERT-based base ranker can yield a suboptimal model. In fact, we even observe that the performance between two identical base rankers fine-tuned in the same manner can be statistically significantly different ($p \leq 0.05$) from each other.

In contrast, when distilled from multiple base rankers, the model performance on both data sets is more stable and consistently higher than the best base ranker performance. Notice that our proposed ensemble methods and distillation strategies are simple and do not require any separate validation data sets. If there are sufficient labeled data to create a validation data set, one can use a more sophisticated method to obtain the ensemble model which can lead to a better distilled model.

Impact of ground-truth labels. We also evaluate the idea of leveraging ground-truth labels during distillation. We conduct experiments on MS MARCO to distill models based on aggregated teacher labels, while using softmax cross entropy loss to instantiate both the distillation loss and the training loss with ground-truth labels in Equation (10). We use both ranking scores and reciprocal rank as teacher labels. The results are shown in Table 3, where we tune the parameter $\alpha \in \{1.0, 0.5, 0.1\}$.

Table 3: Impact of ground-truth labels on MS MARCO based on softmax loss and aggregated teacher labels. The best performances measured by MRR@10 are bolded per column.

α	Ranking Score	Reciprocal Rank
1.0	0.4036	0.4041
0.5	**0.4053**	0.4042
0.1	0.4011	**0.4061**

For both settings, we can observe that the distilled model partially using ground-truth labels (with an appropriate $\alpha < 1.0$) can achieve better performance than simply distilling from ensemble of base rankers ($\alpha = 1.0$). The results indicate that ground-truth labels can still be helpful in the distillation process to provide extra signals that are not necessarily captured by the ensemble model.

6 CONCLUSION

We explored the effectiveness of distilling an ensemble of multiple BERT-based ranking models into a student ranking model instantiated by a single BERT. We studied the combinations of different distillation formulations, teacher labels, and distillation losses. Our experiments are conducted on the MS MARCO and TREC-COVID data sets. We showed that the distilled student model can be as effective as the ensemble model and a listwise ranking loss is more robust than a pointwise loss in a variety of settings. Moreover, when ground-truth labels are also leveraged during the distillation, the distilled model performance can be further improved.

There are a number of potential future directions. An interesting direction is how to develop more sophisticated teacher label derivation [29] for ranking model distillation. For BERT-based ranking models, we mainly show that the advantage of an ensemble can be retained when distilled into a student model with the same model size. Studying the effects of distilling into smaller student models such as BERT-small is an interesting follow-up work.

REFERENCES

[1] Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, et al. 2016. MS MARCO: A Human Generated MAchine Reading COmprehension Dataset. arXiv:1611.09268

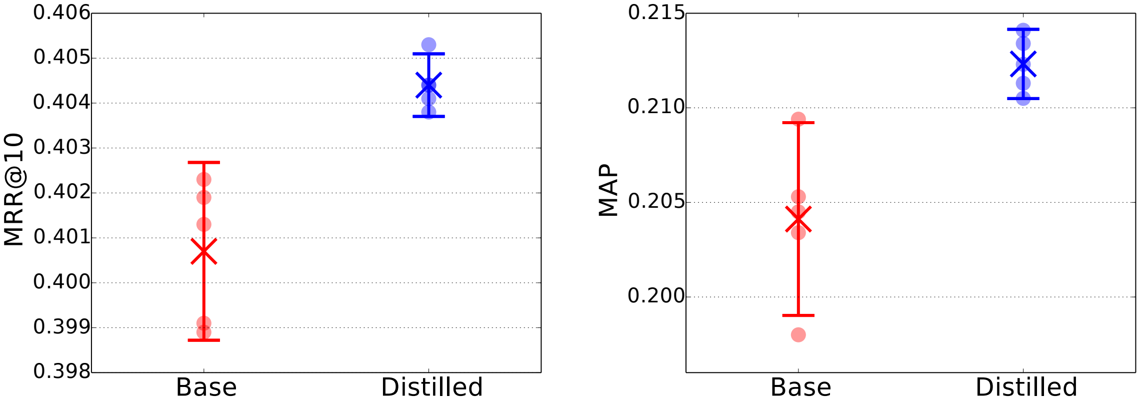
[2] Michael Bendersky, Honglei Zhuang, Ji Ma, Shuguang Han, Keith Hall, and Ryan McDonald. 2020. RRF102: Meeting the TREC-COVID challenge with a 100+ runs ensemble. arXiv:2010.00200

[3] Sebastian Bruch, Xuanhui Wang, Michael Bendersky, and Marc Najork. 2019. An Analysis of the Softmax Cross Entropy Loss for Learning-to-Rank with Binary Relevance. In *Proceedings of the 2019 ACM SIGIR International Conference on Theory of Information Retrieval (ICTIR '19)*. 75–78.

[4] Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. 2007. Learning to Rank: From Pairwise Approach to Listwise Approach. In *Proceedings of the 24th International Conference on Machine Learning (ICML '07)*. 129–136.

[5] Xuanang Chen, Ben He, Kai Hui, Le Sun, and Yingfei Sun. 2021. Simplified TinyBERT: Knowledge Distillation for Document Retrieval. In *Advances in Information Retrieval - Proceedings of the 43rd European Conference on IR Research, Part II (Lecture Notes in Computer Science)*, Vol. 12657. Springer, 241–248.

[6] Gordon V Cormack, Charles LA Clarke, and Stefan Buettcher. 2009. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In



(a) MS MARCO

(b) TREC-COVID

Figure 1: Performance stability of directly using a base ranker vs. using a distilled ensemble of base rankers. Each round dot indicates a separate fine-tuning/distilling process, with their mean performance and 95% confidence intervals also plotted.

Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval. 758–759.

[7] 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 and Short Papers)*. 4171–4186.

[8] Thomas G. Dietterich. 2000. Ensemble Methods in Machine Learning. In *International workshop on multiple classifier systems*. Springer, 1–15.

[9] Tommaso Furlanello, Zachary Lipton, Michael Tschannen, Laurent Itti, and Anima Anandkumar. 2018. Born Again Neural Networks. In *Proceedings of the 35th International Conference on Machine Learning*. 1607–1616.

[10] Luyu Gao, Zhuyun Dai, and Jamie Callan. 2020. Understanding BERT Rankers under Distillation. In *Proceedings of the 2020 ACM SIGIR on International Conference on Theory of Information Retrieval (ICTIR '20)*. 149–152.

[11] Jianping Gou, Baosheng Yu, Stephen John Maybank, and Dacheng Tao. 2020. Knowledge distillation: A survey. arXiv:2006.05525

[12] Shuguang Han, Xuanhui Wang, Mike Bendersky, and Marc Najork. 2020. Learning-to-Rank with BERT in TF-Ranking. arXiv:2004.08476

[13] Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean. 2015. Distilling the Knowledge in a Neural Network. In *NIPS Deep Learning and Representation Learning Workshop*.

[14] Omar Khattab and Matei Zaharia. 2020. ColBERT: Efficient and effective passage search via contextualized late interaction over BERT. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '20)*. 39–48.

[15] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. In *Proceedings of the 8th International Conference on Learning Representations (ICLR '19)*.

[16] Jimmy Lin, Rodrigo Nogueira, and Andrew Yates. 2020. Pretrained transformers for text ranking: BERT and beyond. arXiv:2010.06467

[17] Rodrigo Nogueira and Kyunghyun Cho. 2019. Passage Re-ranking with BERT. arXiv:1901.04085

[18] Rodrigo Nogueira, Wei Yang, Kyunghyun Cho, and Jimmy Lin. 2019. Multi-stage document ranking with BERT. arXiv:1910.14424

[19] Rama Kumar Pasumarthi, Sebastian Bruch, Xuanhui Wang, Cheng Li, Michael Bendersky, Marc Najork, Jan Pfeifer, Nadav Golbandi, Rohan Anil, and Stephan Wolf. 2019. TF-Ranking: Scalable tensorflow library for learning-to-rank. In *ACM SIGKDD International Conference on KnowledgeDiscovery and Data Mining (KDD '19)*. 2970–2978.

[20] Zhen Qin, Le Yan, Honglei Zhuang, Yi Tay, Rama Kumar Pasumarthi, Xuanhui Wang, Michael Bendersky, and Marc Najork. 2021. Are Neural Rankers still Outperformed by Gradient Boosted Decision Trees?. In *International Conference on Learning Representations (ICLR '21)*.

[21] Zhen Qin, Honglei Zhuang, Rolf Jagerman, Xinyu Qian, Po Hu, Chary Chen, Xuanhui Wang, Mike Bendersky, and Marc Najork. 2021. Bootstrapping Recommendations at Chrome Web Store. In *Proceedings of the 27th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '21)*.

[22] Kirk Roberts, Tasmeer Alam, Steven Bedrick, Dina Demner-Fushman, Kyle Lo, Ian Soboroff, Ellen Voorhees, Lucy Lu Wang, and William R Hersh. 2020. TREC-COVID: rationale and structure of an information retrieval shared task for COVID-19. *Journal of the American Medical Informatics Association* 27, 9 (2020), 1431–1436.

[23] Jiaxi Tang and Ke Wang. 2018. Ranking Distillation: Learning Compact Ranking Models With High Performance for Recommender System. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2289–2298.

[24] Meng-Chieh Wu, Ching-Te Chiu, and Kun-Hsuan Wu. 2019. Multi-teacher knowledge distillation for compressed video action recognition on deep neural networks. In *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '19)*. IEEE, 2202–2206.

[25] Ze Yang, Linjun Shou, Ming Gong, Wutao Lin, and Daxin Jiang. 2020. Model Compression with Two-Stage Multi-Teacher Knowledge Distillation for Web Question Answering System. In *Proceedings of the 13th International Conference on Web Search and Data Mining (WSDM '20)*. 690–698.

[26] Fei Yuan, Linjun Shou, Jian Pei, Wutao Lin, Ming Gong, Yan Fu, and Daxin Jiang. 2021. Reinforced Multi-Teacher Selection for Knowledge Distillation. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI '21)*. 14284–14291.

[27] Jingtao Zhan, Jiaxin Mao, Yiqun Liu, Min Zhang, and Shaoping Ma. 2020. An analysis of BERT in document ranking. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '20)*. 1941–1944.

[28] Wangshu Zhang, Junhong Liu, Zujie Wen, Yafang Wang, and Gerard de Melo. 2020. Query Distillation: BERT-based Distillation for Ensemble Ranking. In *Proceedings of the 28th International Conference on Computational Linguistics: Industry Track*. 33–43.

[29] Honglei Zhuang, Xuanhui Wang, Michael Bendersky, and Marc Najork. 2020. Feature transformation for neural ranking models. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1649–1652.