Vera: Prediction Techniques for Reducing Harmful Misinformation in Consumer Health Search

Ronak Pradeep, Xueguang Ma, Rodrigo Nogueira, and Jimmy Lin

David R. Cheriton School of Computer Science
University of Waterloo, Ontario, Canada

ABSTRACT

The COVID-19 pandemic has brought about a proliferation of harmful news articles online, with sources lacking credibility and misrepresenting scientific facts. Misinformation has real consequences for consumer health search, i.e., users searching for health information. In the context of multi-stage ranking architectures, there has been little work exploring whether they prioritize correct and credible information over misinformation. We find that, indeed, training models on standard relevance ranking datasets like MS MARCO passage—which have been curated to contain mostly credible information—yields models that might also promote harmful misinformation. To rectify this, we propose a label prediction technique that can separate helpful from harmful content. Our design leverages pretrained sequence-to-sequence transformer models for both relevance ranking and label prediction. Evaluated at the TREC 2020 Health Misinformation Track, our techniques represent the top-ranked system: Our best submitted run was 19.2 points higher than the second-best run based on the primary metric, a 68% relative improvement. Additional post-hoc experiments show that we can boost effectiveness by another 3.5 points.

CCS CONCEPTS

• **Information systems → Users and interactive retrieval**.

KEYWORDS

Multi-Stage Ranking; Sequence-to-Sequence Models

ACM Reference Format:
Ronak Pradeep, Xueguang Ma, Rodrigo Nogueira, and Jimmy Lin. 2021. Vera: Prediction Techniques for Reducing Harmful Misinformation in Consumer Health Search. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '21), July 11–15, 2021, Virtual Event, Canada.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3404835.3463120

1 INTRODUCTION

The Internet has rapidly grown into an influential medium for producing and disseminating content to broad audiences. With uncontrolled growth come opportunists who use this advantage to distribute misinformation for personal gain. Search engines, which form the gateway to much of this information, can significantly influence user behavior. Thus, systems today face the monumental task of discerning authoritative and correct from dubious and incorrect information. In the current environment amidst the COVID-19 pandemic, there has been an increase in the general public's interest in consumer health issues. System responses to such user queries should attempt to promote *helpful* results while flagging (or even suppressing) *harmful* results to assist users in making informed decisions based on scientific consensus.

The related task of fact verification has been widely studied by the NLP community on corpora such as Wikipedia and discussion blogs [8, 19]. Fact verification systems must predict a claim's veracity and in some cases must provide relevant support from a corpus. More recently, researchers have built various fact verification datasets [9, 20] grounded on scientific corpora, such as the literature on coronaviruses and COVID-19 [22]. However, there are added complexities in consumer health search: Systems need to navigate a larger space of content which contains bad actors spreading misinformation and return documents that capture some collective notion of how helpful and credible the information is, while also suppressing harmful content.

The contribution of our work is a simple yet effective technique to reduce harmful misinformation in consumer health search. We begin with the "Mono-Duo T5" two-stage ranking architecture proposed by Pradeep et al. [16]. Initial first-stage retrieval using BM25 is followed by pointwise and then pairwise reranking, both of which use the sequence-to-sequence model T5 [17]. The main idea behind T5 (Text-to-Text Transfer Transformer) is to cast *every* natural language processing task—for example, classification, question answering, and summarization—as feeding a sequence-to-sequence model some input text and training it to generate some output text. Since this architecture focuses exclusively on relevance ranking, we propose an additional label prediction technique called Vera, which is inspired by the success of VerT5erini [15], a state-of-the-art fact verification system on the SciFACT task [20] that also uses T5. In our implementation, Vera takes advantage of effectiveness judgments from the TREC 2019 Decision Track [1] to promote helpful content and suppress harmful content. A linear combination of prediction scores from Vera and relevance scores from our two-stage reranking design produced the best system at the TREC 2020 Health Misinformation Track [4] in terms of the primary metric [5, 6].

2 TASK DEFINITION

The context of this work is the *ad hoc* retrieval task in the TREC 2020 Health Misinformation Track [4], where systems are provided with a corpus of new articles C and are tasked to return a ranked list of 1000 documents for a set of topics. Recognizing limitations in

SIGIR '21, July 11–15, 2021, Virtual Event, Canada

ACM ISBN 978-1-4503-8037-9/21/07…$15.00
https://doi.org/10.1145/3404835.3463120

Description	Score
Useful; correct; credible	4
Useful; correct; not credible or not judged	3
Useful; no answer or not judged; credible	2
Useful; no answer or not judged; not credible or not judged	1
Not useful; ignored; ignored;	0
Useful; incorrect; not credible or not judged	-1
Useful; incorrect; credible	-2

Table 1: Relevance grades for the TREC 2020 Health Misinformation Track.

the standard notion of topical relevance, the evaluation explicitly assesses whether correct and credible information are prioritized over incorrect information. The document collection comprises news articles from the CommonCrawl news crawl spanning the first four months of 2020 (January 1st, 2020 to April 30th, 2020), covering the onset of the COVID-19 pandemic. This corpus has over 65M articles and is about 1.7 TB in size.

Each of the 46 topics in the evaluation has a description field comprising a question of the form: "Can *A B* COVID-19?", where *A* is a treatment and *B* is one of five effect terms (cause, cure, help, prevent, and worsen). Each topic has an answer field, which is either a "yes" or "no" that corresponds to the medical consensus at the time of topic creation. NIST assessors first judged documents based on three aspects:

- **Usefulness**, which considers if the document includes content that a user might find useful in answering the topic.
- **Correctness**, which verifies if the document's answer aligns with the topic answer (i.e., medical consensus). Note that there is a difference between a document not answering the question and providing an incorrect answer.
- **Credibility**, which assesses the document's credibility.

These aspects are mapped into the graded relevance scale in Table 1. From this, the organizers created "helpful" and "harmful" qrels by taking only the documents with positive and negative grades, respectively. For evaluation, the harmful qrels took the absolute value of the relevance grade (since tools like `trec_eval` expect relevance grades to be positive integers). Note that, importantly, false information may be topically relevant, but harmful.

We focus on three official metrics: helpful compatibility measure, harmful compatibility measure, and their difference denoted by $\text{COMP}_{\text{HELP}}$, $\text{COMP}_{\text{HARM}}$, and COMP_{Δ} (= $\text{COMP}_{\text{HELP}}$ − $\text{COMP}_{\text{HARM}}$), respectively [5, 6]. We want systems that promote helpful content while suppressing harmful content, and hence the organizers selected COMP_{Δ} as the primary metric for the task.

3 SYSTEM ARCHITECTURE

We conceive of a multi-stage ranking architecture [3, 13, 21] as comprising a number of stages, denoted H_0 to H_N. Except for H_0, which retrieves k_0 candidates based on keyword search, each stage H_n receives a ranked list R_{n-1} comprising k_{n-1} candidates from the previous stage. Each stage, in turn, provides a ranked list R_n comprising k_n candidates to the subsequent stage. The ranked list generated by the last stage H_N in the pipeline is fed to Vera to explicitly separate helpful from harmful content.

The output of first-stage retrieval (H_0) is passed to a reranking pipeline comprised of a pointwise reranker, monoT5 (H_1), and then a pairwise reranker, duoT5 (H_2). This basic design was outlined in Pradeep et al. [16]. Note, critically, as demonstrated by our experiments, this design is *not* sufficient for our task, as it has a tendency to retrieve topically relevant but harmful information. In what follows, we describe not only the basic design but necessary modifications for our task.

3.1 H_0: Keyword Retrieval

The candidate generation stage H_0 (also called first-stage retrieval) receives as input the user query q and produces top k_0 candidates R_0. In our implementation, the query is treated as a bag of words for ranking documents from the corpus using a standard inverted index based on BM25 [18]. All our experiments used the Pyserini IR toolkit [2, 10], which provides a Python interface to Anserini [24, 25], itself built on the popular open-source Lucene search engine. At search time, we retrieve the top-1000 documents per query.

3.2 H_1: Pointwise Reranking with monoT5

In stage H_1, documents retrieved in H_0 are reranked by a pointwise reranker called monoT5. The model estimates a score s_i^{mono} quantifying how relevant a candidate $d_i \in R_{n-1}$ is to a query q, that is, $P(\text{Relevant} = 1 | d_i, q)$. Details of monoT5 are described in Nogueira et al. [14]; here, we only provide a short overview.

The monoT5 model uses T5-3B [17] and formulates the problem as a sequence-to-sequence task. Specifically, ranking is performed using the following input sequence template, as suggested by Nogueira et al. [14]:

$$\text{Query: } q \quad \text{Document: } d \quad \text{Relevant:} \tag{1}$$

where q and d are the query and document texts, respectively. The model is fine-tuned to produce the token "true" or "false" depending on whether the document is relevant or not to the query.

For the TREC 2020 Health Misinformation Track, the default question text is the topic description. We call this standard template the $\text{monoT5}_{\text{base}}$ variant. Alternatively, we also consider a variant, $\text{monoT5}_{\text{NL}}$, where we rephrase the question "Can *A B* COVID-19?" and the answer field in a natural language sentence form, i.e., as "*A* can *B* COVID-19" if the answer field is "yes" and as "*A* can not *B* COVID-19" if the answer field is "no". The goal of this template is to see if there are any improvements to be gained by aligning the query text with the answer field.

We train the monoT5 model by first fine-tuning on the MS MARCO passage dataset and then fine-tuning it again on Med-MARCO, which is a subset of the MS MARCO passage dataset where only queries containing medical terms are kept [12]. Zhang et al. [26] called this training strategy "pre–fine-tuning"; see Pradeep et al. [16] for additional details.

At inference time, to compute probabilities for each query–document pair, we apply a softmax only to the logits of the "true" and "false" tokens and rerank the top-1000 documents according to the probabilities assigned to the "true" token.

As discussed in Lin et al. [11], one reoccurring theme in the application of transformers to text ranking is the handling of texts that are longer than the input sequences that the models were

designed to handle (typically, 512 tokens). Following Pradeep et al. [16], we first segment each document into passages by applying a sliding window of six sentences with a stride of three. Each passage was then prepended with the title of the document. We obtain a probability of relevance for each passage by performing inference on it independently, and then select the highest probability among the passages as the relevance score of the document; this technique has been called MaxP [7, 26].

3.3 H_2: Pairwise Reranking with duoT5

The output R_1 from the previous stage serves as input to the pairwise reranker we call duoT5. In this pairwise approach, the reranker considers a pair of documents (d_i, d_j) and estimates the probability $p_{i,j}$ that candidate d_i is more relevant than d_j to query q, that is, $P(d_i \succ d_j | d_i, d_j, q)$, where $d_i \succ d_j$ denotes that d_i is *more relevant* than d_j (with respect to the query q).

Details of duoT5, including the default hyperparameters used in our work, are described in Pradeep et al. [16]; here, we only provide a short overview. As the name suggests, duoT5 is also based on T5-3B and takes as input the sequence:

$$\text{Query: } q \quad \text{Document0: } d_i \text{ Document1: } d_j \quad \text{Relevant:} \tag{2}$$

For the TREC 2020 Health Misinformation Track, we have the two variants, $\text{duoT5}_{\text{base}}$ and duoT5_{NL} taking the same query templates as the pointwise reranker.

At inference time, we aggregate the pairwise scores $p_{i,j}$ so that each document receives a single score s_i using the Sym-Sum method proposed by Pradeep et al. [16], where $J_i = \{0 \le j < k_1, j \ne i\}$:

$$\text{Sym-Sum} : s_i = \sum_{j \in J_i} \left(p_{i,j} + (1 - p_{j,i}) \right) \tag{3}$$

In previous work, the top-50 candidates in R_1 are reranked according to their scores s_i to obtain a ranked list of candidates R_2 designed for final consumption (thus requiring 50×49 individual inferences). However, in the TREC 2020 Health Misinformation Track, the *ad hoc* retrieval task required a ranked list of 1000 documents per topic. Since Vera performs a linear combination using the scores from multi-stage ranking and label prediction, to keep the combination meaningful we need top-1000 scores for all candidates. One solution would be to run duoT5 on all top-1000 candidates, but this would be very computationally expensive due to the quadratic nature of the pairwise approach. To address this issue, we devised an alternative solution: We still only rerank the top-50 R_1 candidates, but form an intermediate ranked list, denoted by R_2', the scores of which we post-process to combine with the scores from R_1 to obtain a final ranked list of 1000 candidates.

Let the score of a document d_i in R_1 be s_i^{mono}, and the highest and lowest monoT5 scores of candidates in R_2' be $s_{\text{max}}^{\text{mono}}$ and $s_{\text{min}}^{\text{mono}}$, respectively. Similarly, let the highest and lowest scores after aggregating over candidates in R_2' be s_{max} and s_{min}, respectively. Then we calculate the final duoT5 scores using one of two methods:

$$\begin{aligned} s_i^{\text{duo}_1} &= \begin{cases} s_{\text{min}}^{\text{mono}} + \frac{(s_i - s_{\text{min}})(s_{\text{max}}^{\text{mono}} - s_{\text{min}}^{\text{mono}})}{s_{\text{max}} - s_{\text{min}}}, d_i \in R_2' \\ s_i^{\text{mono}}, d_i \notin R_2' \end{cases} \\ s_i^{\text{duo}_2} &= \begin{cases} s_i^{\text{mono}} + s_i, d_i \in R_2' \\ s_i^{\text{mono}}, d_i \notin R_2' \end{cases} \end{aligned} \tag{4}$$

These scores determine the final top-1000 ranked list, R_2.

At inference time, we use the highest scoring monoT5 passage as the representative passage for each document. We feed the duoT5 model pairs of representative passages from the documents under consideration to compute the pairwise scores, which are then aggregated to yield the relevance score of each document. We increase the maximum input tokens from the default of 512 to 1024 to account for pairs of passages being twice as long.

3.4 Label Prediction

We cast the problem of separating helpful from harmful content as a label prediction task. Our Vera model, also based on T5-3B, was inspired by Pradeep et al. [15] and T5's pretraining on MNLI [23]. Given the topic q and the highest monoT5 scoring segment s_i from a document d_i, the model is tasked to predict a label $\hat{y}(q, s_i) \in \{\text{true}, \text{weak}, \text{false}\}$. Here, we use the following input sequence:

$$\text{Query: } q \quad \text{Document: } s_i \quad \text{Relevant:}$$

The query in this case is the topic description. We train the label prediction model using effectiveness judgments from the TREC 2019 Decision (Medical Misinformation) Track. We map effective and ineffective judgments to "true" and "false" respectively; documents judged as inconclusive, no info, or not relevant are all labeled "weak". In total, these judgments only constitute approximately 4K labelled examples, meaning that Vera operates in a low-data regime. We fine-tuned our Vera-3B model with a constant learning rate of 10^{-3} for 500 iterations with batches of size 128. We used a maximum of 512 inputs tokens and one output token. Training Vera-3B takes approximately 40 minutes on a single Google TPU v3-8.

At classification time, to compute probabilities for each query–document pair, we apply a softmax only to the logits of the "true", "weak", and "false" tokens. For a particular document d_i, suppose the probabilities assigned to the "true" token and "false" token are t_i and f_i, respectively. We adopt the following scoring scheme:

$$s_i^{\text{final}} = \lambda \cdot s_i^z + (1 - \lambda) \cdot \begin{cases} t_i - f_i, \text{answer field is "yes"} \\ f_i - t_i, \text{answer field is "no"} \end{cases} \tag{5}$$

which we denote as Vera (λ, z), where $z \in \{\text{mono}, \text{duo}_1, \text{duo}_2\}$ (referred to as the "relevance setting"). Then we rerank the candidates according to the scores s_i^{final} to obtain the final ranked list.

4 RESULTS

Table 2 reports results from the TREC 2020 Health Misinformation Track. For reference, row (a) provides the median score across 51 submissions from eight groups for the evaluation. Rows (b)–(d) present the three top-scoring submitted runs (per group); note that row (c) represents a manual submission. As we can see, the Vera technique described here was the top-scoring run submitted to the evaluation by a large margin. Rows (e)–(j) denote additional runs that were part of our official submission; rows (e)–(i) show the results of different configurations that used only relevance ranking (i.e., no label prediction). Rows (b) and (i)–(m) represent variants that combine both relevance and label prediction scores, as described in Section 3.4. Rows (k)–(m) show results of the highest scoring configuration on top of each of the three relevance ranking methods, discovered by an evenly spaced sweep of the linear

Model	$\text{COMP}_{\text{HELP}}$	$\text{COMP}_{\text{HARM}}$	COMP_{Δ}
(a) Median	0.334	0.075	0.259
(b) Vera ($\lambda = 0.5, z =$ mono) = h2oloo.m8	0.490^{ej}	0.016^{efghim}	0.474^{efgj}
(c) cn-kq-td (Webis)	0.334	0.052	0.282
(d) adhoc_run3 (KU)	0.401	0.121	0.280
(e) BM25 = h2oloo.m1	0.368	0.120	0.248
(f) + $\text{monoT5}_{\text{base}}$ = h2oloo.m2	0.440	0.113	0.327
(g) + $\text{duoT5}_{\text{base}}$ = h2oloo.m4	0.466^{e}	0.120	0.346^{e}
(h) + $\text{monoT5}_{\text{NL}}$ = h2oloo.m3	0.511^{ef}	0.075^{eg}	0.436^{efg}
(i) + duoT5_{NL} = h2oloo.m5	0.549^{efghj}	0.080^{eg}	0.469^{efg}
(j) Vera ($\lambda = 0.0, z =$ mono) = h2oloo.m7	0.449	0.015^{efghim}	0.434^{e}
(k) Vera ($\lambda = 0.95, z =$ mono)	0.507^{ef}	0.019^{efghim}	0.488^{efgj}
(l) Vera ($\lambda = 0.95, z = \text{duo}_1$)	0.520^{bef}	0.018^{efghim}	0.502^{efgj}
(m) Vera ($\lambda = 0.75, z = \text{duo}_2$)	$0.546^{befghjk}$	0.037^{efgi}	0.509^{efghj}

Table 2: Compatibility scores on the TREC 2020 Health Misinformation Track. Results of significance tests (t-tests, $p < 0.05$) are denoted by superscripts.

combination parameter λ. These configurations were not official submissions to the evaluation and come with the added benefit of hindsight. We applied t-tests ($p < 0.05$) to determine the statistical significance of metric differences, except for (a), (c), and (d); these results are denoted by the standard superscript notation.

Let us begin by focusing only on relevance ranking. We see that pointwise reranking helps on top of the BM25 baseline, row (e) vs. (f), and pairwise reranking helps on top of pointwise reranking, row (f) vs. (g), as expected. These three settings all have similar $\text{COMP}_{\text{HARM}}$ scores that are higher than the median, row (a). That is, our runs are surfacing not only more helpful content, but more harmful results as well. This comes as no surprise since our models are only trained on relevance ranking, and indeed, topically relevant information can be harmful. Note here also that these settings used only the topic description.

Both $\text{COMP}_{\text{HELP}}$ and $\text{COMP}_{\text{HARM}}$ scores improve when the query is rephrased to align with the topic answer, comparing the "base" and "NL" input template variants, row (h) vs. (f) and row (i) vs. (g). This suggests that these models are capturing notions of "answer correctness" despite being trained on relatively clean relevance ranking datasets like MS MARCO passage. Since we notice improved effectiveness across the board, all further experiments use this input template. Note that here, we still have not added label prediction, and our runs are already substantially better than other submissions to the evaluation, rows (c) and (d). However, these runs still score above the median in $\text{COMP}_{\text{HARM}}$, which is concerning.

The label prediction model in isolation (i.e., $\lambda = 0$) results in a large reduction in $\text{COMP}_{\text{HARM}}$ compared to the pointwise ranker, row (j) vs. (h). However, this comes with a drop in $\text{COMP}_{\text{HELP}}$ as well. This motivates our linear combination approach described in Section 3.4 that incorporates relevance and label prediction signals. Our top submission, Vera ($\lambda = 0.5, z =$ mono), row (b), greatly improves upon the label prediction model, row (j), in terms of $\text{COMP}_{\text{HELP}}$ with only a negligible increase in $\text{COMP}_{\text{HARM}}$.

In Figure 1, we plot $\text{COMP}_{\text{HELP}}$ and COMP_{Δ} scores as a function of λ for three different settings. First, we note that for both mono and duo_1, $\text{COMP}_{\text{HELP}}$ increases as we increase the weight on relevance ranking; COMP_{Δ} increases all the way until 0.95, after which the measure drops because we're reverting to pointwise and pairwise reranking, respectively. Second, we see that mono and duo_1 follow

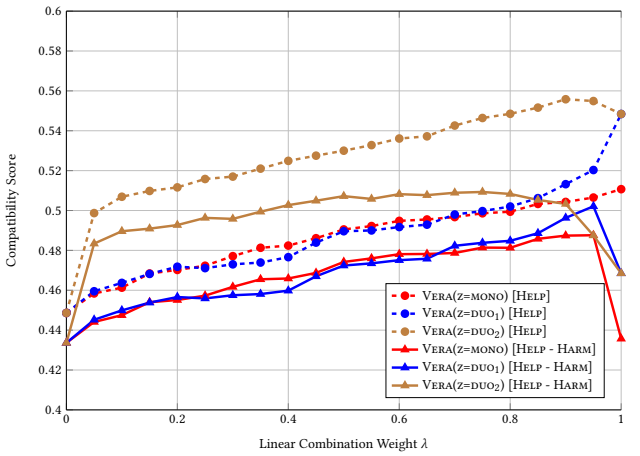

Figure 1: Compatibility scores of the system over three multi-stage ranking scoring schemes.

similar trajectories across both measures until $\lambda = 0.8$, after which duoT5's higher $\text{COMP}_{\text{HELP}}$, which we can see from row (i) vs. (h), "kicks in". We note an improvement of around 1.5 points in both $\text{COMP}_{\text{HELP}}$ and COMP_{Δ} at $\lambda = 0.95$, row (l) vs. (k).

The duo_2 relevance setting behaves differently from mono and duo_1: $\text{COMP}_{\text{HELP}}$ appears convex, with a maximum at $\lambda = 0.9$. As a result, COMP_{Δ} also follows a different trajectory in that the curve is flatter for intermediate λ values compared to other metrics, but performs consistently better than the other configurations. COMP_{Δ} is maximized at $\lambda = 0.75$, shown in row (m), and this represents our most effective system configuration. Note that this constitutes an 80% relative improvement compared the cn-kq-td manual run by the Webis team, shown in row (c). Overall, our experimental results show that adapting a multi-stage ranking pipeline to incorporate a harmful information classifier like Vera is an easy and effective solution to reduce misinformation in consumer health search.

5 CONCLUSIONS

In this paper, we analyze how multi-stage neural reranking designs perform at prioritizing correct and credible information over misinformation. We find that since these models focus on relevance ranking, they have a tendency to return both helpful information as well as topically relevant but harmful misinformation.

To combat this, we introduced Vera, a label prediction model that exploits a generation-based approach to rerank candidates from pure relevance ranking models to suppress harmful content. Experiments show that our system outperforms other systems submitted to the *ad hoc* retrieval task in the TREC 2020 Health Misinformation Track by a large margin. Our design can potentially improve consumer health search to combat misinformation, a challenge recently amplified by the COVID-19 pandemic.

ACKNOWLEDGEMENTS

This research was supported in part by the Canada First Research Excellence Fund, the Natural Sciences and Engineering Research Council (NSERC) of Canada, and the Waterloo–Huawei Joint Innovation Laboratory. Additionally, we would like to thank Google for computational resources in the form of Google Cloud credits.

REFERENCES

[1] Mustafa Abualsaud, Christina Lioma, Maria Maistro, Mark D. Smucker, and Guido Zuccon. 2019. Overview of the TREC 2019 Decision Track. In *Proceedings of the Twenty-Eigth Text REtrieval Conference (TREC 2019)*.

[2] Zeynep Akkalyoncu Yilmaz, Charles L. A. Clarke, and Jimmy Lin. 2020. A Lightweight Environment for Learning Experimental IR Research Practices. In *Proceedings of the 43rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2020)*. 2113–2116.

[3] Nima Asadi and Jimmy Lin. 2013. Effectiveness/Efficiency Tradeoffs for Candidate Generation in Multi-Stage Retrieval Architectures. In *Proceedings of the 36th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2013)*. Dublin, Ireland, 997–1000.

[4] Charles L.A. Clarke, Maria Maistro, and Mark D. Smucker. 2020. Overview of the TREC 2020 Health Misinformation Track (Notebook). In *Proceedings of the Twenty-Ninth Text REtrieval Conference (TREC 2020)*.

[5] Charles L.A. Clarke, Alexandra Vtyurina, and Mark D. Smucker. 2020. Offline Evaluation without Gain. In *Proceedings of the 2020 ACM SIGIR International Conference on Theory of Information Retrieval (ICTIR '20)*. 185–192.

[6] Charles L. A. Clarke, Mark D. Smucker, and Alexandra Vtyurina. 2020. Offline Evaluation by Maximum Similarity to an Ideal Ranking. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management (CIKM '20)*. 225–234.

[7] Zhuyun Dai and Jamie Callan. 2019. Deeper Text Understanding for IR with Contextual Neural Language Modeling. In *Proceedings of the 42nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2019)*. Paris, France, 985–988.

[8] Andreas Hanselowski, Christian Stab, Claudia Schulz, Zile Li, and Iryna Gurevych. 2019. A Richly Annotated Corpus for Different Tasks in Automated Fact-Checking. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*. Hong Kong, China, 493–503.

[9] Nayeon Lee, Yejin Bang, Andrea Madotto, and Pascale Fung. 2020. Misinformation has High Perplexity. *arXiv:2006.04666* (2020).

[10] Jimmy Lin, Xueguang Ma, Sheng-Chieh Lin, Jheng-Hong Yang, Ronak Pradeep, and Rodrigo Nogueira. 2021. Pyserini: A Python Toolkit for Reproducible Information Retrieval Research with Sparse and Dense Representations. In *Proceedings of the 44th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2021)*.

[11] Jimmy Lin, Rodrigo Nogueira, and Andrew Yates. 2020. Pretrained Transformers for Text Ranking: BERT and Beyond. *arXiv:2010.06467* (2020).

[12] Sean MacAvaney, Arman Cohan, and Nazli Goharian. 2020. SLEDGE: A Simple Yet Effective Zero-Shot Baseline for Coronavirus Scientific Knowledge Search. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 4171–4179.

[13] Irina Matveeva, Chris Burges, Timo Burkard, Andy Laucius, and Leon Wong. 2006. High Accuracy Retrieval with Multiple Nested Ranker. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2006)*. Seattle, Washington, 437–444.

[14] Rodrigo Nogueira, Zhiying Jiang, Ronak Pradeep, and Jimmy Lin. 2020. Document Ranking with a Pretrained Sequence-to-Sequence Model. In *Findings of EMNLP*.

[15] Ronak Pradeep, Xueguang Ma, Rodrigo Nogueira, and Jimmy Lin. 2021. Scientific Claim Verification with VerT5erini. In *Proceedings of the 12th International Workshop on Health Text Mining and Information Analysis*. 94–103.

[16] Ronak Pradeep, Rodrigo Nogueira, and Jimmy Lin. 2021. The Expando-Mono-Duo Design Pattern for Text Ranking with Pretrained Sequence-to-Sequence Models. *arXiv:2101.05667* (2021).

[17] 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. *Journal of Machine Learning Research* 21, 140 (2020), 1–67.

[18] Stephen E. Robertson, Steve Walker, Susan Jones, Micheline Hancock-Beaulieu, and Mike Gatford. 1994. Okapi at TREC-3. In *Proceedings of the 3rd Text REtrieval Conference (TREC-3)*. 109–126.

[19] James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: A Large-scale Dataset for Fact Extraction and VERification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. New Orleans, Louisiana, 809–819.

[20] David Wadden, Shanchuan Lin, Kyle Lo, Lucy Lu Wang, Madeleine van Zuylen, Arman Cohan, and Hannaneh Hajishirzi. 2020. Fact or Fiction: Verifying Scientific Claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 7534–7550.

[21] Lidan Wang, Jimmy Lin, and Donald Metzler. 2011. A Cascade Ranking Model for Efficient Ranked Retrieval. In *Proceedings of the 34th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2011)*. Beijing, China, 105–114.

[22] Lucy Lu Wang, Kyle Lo, Yoganand Chandrasekhar, Russell Reas, Jiangjiang Yang, Doug Burdick, Darrin Eide, Kathryn Funk, Yannis Katsis, Rodney Kinney, Yunyao Li, Ziyang Liu, William Merrill, Paul Mooney, Dewey Murdick, Devvret Rishi, Jerry Sheehan, Zhihong Shen, Brandon Stilson, Alex Wade, Kuansan Wang, Nancy Xin Ru Wang, Chris Wilhelm, Boya Xie, Douglas Raymond, Daniel S. Weld, Oren Etzioni, and Sebastian Kohlmeier. 2020. CORD-19: The COVID-19 Open Research Dataset. arXiv:2004.10706 [cs.DL]

[23] Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A Broad-Coverage Challenge Corpus for Sentence Understanding through Inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. New Orleans, Louisiana, 1112–1122.

[24] Peilin Yang, Hui Fang, and Jimmy Lin. 2017. Anserini: Enabling the Use of Lucene for Information Retrieval Research. In *Proceedings of the 40th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2017)*. Tokyo, Japan, 1253–1256.

[25] Peilin Yang, Hui Fang, and Jimmy Lin. 2018. Anserini: Reproducible Ranking Baselines Using Lucene. *Journal of Data and Information Quality* 10, 4 (2018), Article 16.

[26] Xinyu Zhang, Andrew Yates, and Jimmy Lin. 2021. Comparing Score Aggregation Approaches for Document Retrieval with Pretrained Transformers. In *Proceedings of the 43rd European Conference on Information Retrieval (ECIR 2021), Part II*. 150–163.