On Survivorship Bias in MS MARCO

Prashansa Gupta
University of Glasgow
Glasgow, United Kingdom
2539575G@student.gla.ac.uk

Sean MacAvaney
University of Glasgow
Glasgow, United Kingdom
sean.macavaney@glasgow.ac.uk

ABSTRACT

Survivorship bias is the tendency to concentrate on the positive outcomes of a selection process and overlook the results that generate negative outcomes. We observe that this bias could be present in the popular MS MARCO dataset, given that annotators could not find answers to 38–45% of the queries, leading to these queries being discarded in training and evaluation processes. Although we find that some discarded queries in MS MARCO are ill-defined or otherwise unanswerable, many are valid questions that *could* be answered had the collection been annotated more completely (around two thirds using modern ranking techniques). This survivability problem distorts the MS MARCO collection in several ways. We find that it affects the natural distribution of queries in terms of the type of information needed. When used for evaluation, we find that the bias likely yields a significant distortion of the absolute performance scores observed. Finally, given that MS MARCO is frequently used for model training, we train models based on subsets of MS MARCO that simulates more survivorship bias. We find that models trained in this setting are up to 9.9% worse when evaluated on versions of the dataset with more complete annotations, and up to 3.5% worse at zero-shot transfer. Our findings are complementary to other recent suggestions for further annotation of MS MARCO, but with a focus on *discarded* queries. Code and data for reproducing the results of this paper are available in an online appendix.[1]

CCS CONCEPTS

• **Information systems → Information retrieval**; **Evaluation of retrieval results**.

KEYWORDS

Survivorship Bias, MS MARCO, Evaluation, Neural IR

1 INTRODUCTION

Survivorship bias is a type of selection bias where one focuses more on outcomes that pass (survive) a selection process, rather than those that do not [9]. Focusing on only the survivors distorts one's view of reality, however. A well-known example of survivorship bias is that when aircraft return from combat and some are lost, the surviving aircraft should (perhaps unintuitively) be reinforced where damage is *not* apparent, since damage elsewhere is fatal [19].

Historically, most test collections in information retrieval (e.g., those from TREC) have used a deep pooling technique, where assessors judge many documents for each query (usually hundreds to thousands), pooled from a diverse set of systems [26]. This process yields datasets that are relatively complete and can sometimes even accurately measure the performance of ranking algorithms

developed decades later (e.g., as seen in [14, 27]). However, they are usually limited in the number of queries (usually tens to hundreds), which can reduce the statistical power of experiments that use them [2]. The prominent MS MARCO dataset diverges from this paradigm by assessing far more queries (hundreds of thousands), but with a shallow pooling technique over a single system [4]. We posit that this technique not only biases the "correct" answers for each query (a phenomenon studied by Arabzadeh et al. [1]), but also biases the queries included in the dataset itself; queries that couldn't be found in the top handful of results from a single system are discarded. This is a sizable number of queries, in fact: 38% and 45% of queries in the MS MARCO train and dev set, respectively, are discarded. We see this situation as ripe for survivorship bias.

To study the survivorship bias in MS MARCO, we (1) perform an initial qualitative and quantitative analysis of the queries discarded in the MS MARCO passage ranking dataset (Section 3); (2) measure the effect of survivorship bias on the capacity of MS MARCO to provide meaningful evaluations by simulating versions of the dataset that would result in fewer surviving queries (Section 4); and (3) use the simulation approach from (2) to test the effect survivorship has on models *trained* using MS MARCO (Section 5). We find that the majority of the discarded queries in MS MARCO are likely answerable in the available corpus. Further, through extrapolation from our simulation experiments on both training and evaluation, we find that conclusions may differ had MS MARCO contained queries that were discarded. This work motivates further annotation of the MS MARCO dataset with a focus on queries that were previously discarded (e.g., through the TREC Deep Learning Track).

2 BACKGROUND, DATA, AND SCOPE

For various parts of our analysis, it is important to understand the annotation process of MS MARCO. Assessors were shown a question-like query and the top 10 passages[2] retrieved by a single system. Based on those answers, the assessor was asked to write a natural-language answer to the question and indicate which passages contributed to it (usually just one passage). If no answer was present in the top results, the question is marked as having "No Answer Present". These annotations serve as the basis for a variety of tasks, the most popular of which being the passage and document ranking tasks. In passage and document ranking, the passages (or the documents from which they were extracted) that were marked by assessors are treated as the ground truth; systems are tasked with finding these passages from a corpus of possible answers (8.8M passages or 3.2M documents). Since queries marked as having "No Answer Present" have no associated relevant ground truth passages, these queries are discarded for the purposes of passage and document ranking. In other words, queries where the answer could not

---

[2] A small minority of queries have more than 10 associated passages, suggesting that assessors were given the option to expand the list if desired.

Table 1: Discarded queries by reason they were unanswered.

Split	no_ans	partial_ans	wrong_lab	ill_form	Total
Dev	70	15	14	1	100
Train	36	5	6	3	50

Table 2: Number of discarded queries that could be answered by other ranking models.

	BM25	ANCE	ColBERT	monoT5
No answer present	22 (44%)	22 (44%)	18 (36%)	16 (32%)
Answer in top 10	28 (56%)	28 (56%)	32 (64%)	34 (68%)
Answer in top 3	17 (34%)	17 (34%)	28 (56%)	30 (60%)

be found in the top 10 results of a single system (38% of queries in train and 45% in dev) are discarded and therefore not used for training or evaluation of ranking systems. Since the document task is simply a derivation of the passage task, we focus our study on the latter.

The passage ranking task data itself does not provide adequate information to thoroughly study this problem. However, we observe that another task derived from the same data – the MS MARCO QnA task – provides more complete information. Most importantly, it gives the ordered list of candidate passages shown to the assessors. As we show in Sections 4 and 5, this information allows us to simulate environments where a more stringent annotation process would be conducted (i.e., one where assessors were shown fewer passages, and therefore fewer queries survive). To further assist in our analysis, we also make use of the automatically-assigned question types, which are also only provided by the QnA task.

We note that there is also the opportunity for survivability bias in other datasets as well. A query (topic) development process is commonplace when constructing IR test collections, such as those from TREC. This process often aims to balance a variety of factors, including assessor interest, difficulty, and estimated number of relevant documents [24, 25]. This process is known to exhibit selection biases, but since the full space of queries considered in these settings are unknown (e.g., they be sourced from private query logs, social media calls, or other processes), the study of survivorship bias in these settings is challenging [20].

Further, a newer version of MS MARCO (named v2) was recently constructed that includes a larger passage corpus [6]. This dataset uses an even stricter subset of the MS MARCO passage labels, filtering out those where passages have insufficient textual overlap with any passage in the new corpus. This process resulted in 65% of queries being discarded (up from 38% in v1). The survival process here is clearer than the process for v1, since it relates exclusively to changes in online document content over time and changes to the passaging technique used to construct v2. Further, since MS MARCO v2 is not yet widely used for training or evaluation of systems, we focus our study on v1 of the dataset.

Relatedly, we note that a similar survivorship bias exists in the v1 corpus itself: only passages that were shown to assessors are included in the corpus. MS MARCO v2 fixes this problem by including more passages. Therefore, we limit our study to the survivorship bias in terms of queries.

We use ir-datasets [15] to access data for our analysis. This package provides access to both the MS MARCO passage ranking dataset[3] and the MS MARCO QnA[4]. Critically, it also provides a mapping between passages IDs the two datasets (which are not available by default). Matching is conducted by finding textual matches between the two datasets. Unique exact matches are found

for all passages once UTF encoding issues are corrected in the passage ranking dataset, so we are confident this process is accurate.

Throughout our analysis, we use four different scoring functions, representative of four different classes of models: sparse lexical (BM25 [22]), single-representation (ANCE [28]), multi-representation (ColBERT [10]), and cross-encoder (monoT5 [21]). For neural models (ANCE, ColBERT, monoT5) we use checkpoints released by the authors and model implementations provided by PyTerrier [16]. To isolate the effect of the model *scoring* function on performance, we use all as re-ranking models over the top 100 BM25 results (using default BM25 parameters).

3 ANALYSIS OF DISCARDED QUERIES

We start our investigation of survivorship bias in MS MARCO by performing a qualitative and quantitative assessment of the discarded queries. We select 100 unanswered queries (i.e., those without any relevance assessments) from the development set and 50 unanswered queries from the training set. We manually inspect these queries along with the top results shown to the assessors (usually 10). Through the inspection, we identify four main situations that occur: (1:no_ans) the assessor correctly identifies passages do not answer the query, (2:partial_ans) a passage partially (but not fully) answers the question, (3:wrong_lab) a passage was presented to the assessor that fully answers the question, and (4:ill_form) the question is underspecified or otherwise unanswerable.

A summary of the findings from our manual inspection are shown in Table 1. Importantly, we find that the vast majority of questions themselves are reasonable; only 4 of the 150 queries we inspected are ill-formed. This suggests that there is at least the *potential* for the remaining queries to be answered. We found a relatively low proportion of queries to have been incorrectly labeled as non-answerable (20 of 150, 13%), which we consider to be reasonable given the subjective nature of the task. The remaining 126 queries (84%) were correctly labeled as having no answer present (potentially due to incompleteness of the passages given). Overall, these results strongly suggest that the discarded queries from MS MARCO are a result of the shallow pooling technique; most questions are well-formed, the answer simply did not appear in passages shown to the assessor.

Although we now know that most discarded queries are well-formed, we do not yet know whether they could have actually been answered from the available passage corpus. For instance, the questions could be so specific that the answer simply isn't available online, and therefore could not have been part of the web corpus used to construct MS MARCO. To check whether questions reasonably *could* have been answered, we select 50 queries from the prior analysis that were well-formed and not mislabeled. For

---

[3] Dataset ID: msmarco-passage [4] Dataset ID: msmarco-qna

Table 3: Distribution of unanswered queries in MS MARCO dev by provided query type label from MS MARCO QnA.

Query Type	Total	Unanswered	(% of Total)
DESCRIPTION	54,616	25,519	(46.7%)
NUMERIC	25,547	12,048	(47.2%)
ENTITY	8,677	3,819	(44.0%)
PERSON	6,425	2,487	(38.7%)
LOCATION	5,828	1,642	(28.2%)

these queries, we retrieve the top 10 passages from MS MARCO passages using BM25, ANCE, ColBERT, and monoT5. We then label the passages that answer each of the questions from these pools.

Results for this analysis are shown in Table 2. We find that all four models are able to identify an answer to a majority of discarded queries within the top 10 results (56–68%). Note that this serves as a lower bound of the proportion of queries that are potentially answerable in the MS MARCO passage corpus; there could be answers available that were not retrieved by these models. These results suggest that not only are the majority of discarded queries in MS MARCO well-formed, but also answerable in the available corpus.[5] However, we did observe significant overlap within these pools; when considering all four systems, 15 of the queries remain unanswerable.

Next, we ask whether there are measurable differences between the types of queries that were discarded and those that survived. Table 3 breaks down the number of queries by the question type. (Recall from Section 2 that these question types are automatically assigned and provided by MS MARCO QnA dataset.) We find that DESCRIPTION and NUMERIC queries are far less likely to survive than LOCATION queries. This suggests that LOCATION queries are over-represented in MS MARCO, and that, in general, the queries that are discarded come from a different distribution than those that survive. This distribution shift is potentially concerning, given that others have found description queries to be among the more challenging and interesting of the query types [18].

4 SURVIVORSHIP BIAS IN EVALUATION

Given that the MS MARCO dev set is commonly used for the evaluation ranking models, we now examine whether the performance of systems may be affected due to survivorship bias. To this end, we conduct an experiment where we *simulate* stricter annotation settings based on the data available from the QnA task and extrapolate conclusions from these results. This setting is both low-cost and reproducible (it does not rely on subjective human assessment.)

4.1 Experimental Setup

Based on Section 3, we know that a major factor in query survivability is the annotation depth; had annotators inspected more documents, they likely would have been able to answer more of the queries. Collecting a suitable number of annotations for the 45,551

Table 4: Effect of simulated survivability levels on at different annotation depths in terms of RR@10. Significant differences between the scores of the full MS MARCO dev set (10+) are indicated with * (unpaired t-test, $p < 0.05$, Bonferroni corr.)

Depth	BM25	ANCE	ColBERT	monoT5	NumQ	(Δ NumQ)
1	*0.174	*0.240	*0.301	*0.332	8,192	
2	*0.176	*0.244	*0.305	*0.336	15,532	(+7,340)
3	*0.180	*0.249	*0.311	*0.342	22,041	(+6,509)
4	0.181	*0.251	*0.315	*0.347	27,837	(+5,796)
5	0.182	0.253	*0.318	*0.350	33,039	(+5,202)
6	0.183	0.255	*0.321	*0.354	37,998	(+4,959)
7	0.184	0.257	0.325	*0.357	42,640	(+4,642)
8	0.185	0.258	0.328	0.360	47,114	(+4,474)
9	**0.187**	0.259	0.330	0.363	51,309	(+4,195)
10+	**0.187**	**0.260**	**0.331**	**0.365**	55,542	(+4,233)

discarded queries in MS MARCO could be potentially very expensive, especially if a greater annotation depth is required for each query. Instead, we simulate settings where lower annotation depths were used and extrapolate from those results to higher depths. We accomplish this simply by filtering the full MS MARCO dev dataset's qrels to only those that occurred in the top k results shown to the annotator. We note that this setup makes the assumption that had a valid answer appeared at a higher rank, the annotator would have selected this answer instead (or selected both) – a reasonable assumption given well-known rank biases [8]. We report RR@10 results for BM25, ANCE, ColBERT, and monoT5 using the official MS MARCO evaluation script, and perform statistical testing, with the entire set of qrels as the baseline (representing a null hypothesis that the ranking depth has no effect on the performance of individual queries).

4.2 Results

Table 4 presents our experimental results for testing the effect of query survivability on evaluations that use the MS MARCO dev set. The results show that we can reject the null hypothesis that query performance is irrespective of the annotation depth, given that significant differences exist between model performance using only qrels found at lower annotation depths. We observe that this trend is consistent across four classes of models. Perhaps unexpectedly, we find that query performance actually *increases* as deeper judgments are included (e.g., monoT5 performance increases from 0.332 to 0.365, +9.0% relative improvement).[6] While this may be expected for models that employ neural networks (which could be overfit to particular annotation artefacts), we find that the trend also exists for the unsupervised BM25 model, though to a lesser extent (0.174 to 0.187, +7.5% relative improvement). Extrapolating these results to further depths we can conclude that the reported model performance is likely to differ had the collection been annotated to a further depth. This suggests MS MARCO may be a poor estimate of the *actual* performance of a ranker over a natural selection of queries. However, we note that all models we examine

---

[5] We performed the same analysis on the MS MARCO v2 passage corpus, which has over 15× as many passages. We found that even more of the discarded queries were answerable in that corpus (over 75%).

[6] One may suspect that this trend could alternatively be explained because more possible relevant documents are introduced to the pool at each step, increasing the chance of a hit. However, we find that this isn't the case, through an experiment where we kept only the first qrel per query and observed the same trend.

exhibit roughly the same behaviour and the *relative* ranking of systems remains the same. This mirrors findings that suggest similar behaviour is present with more qrels per query (both by means of the TREC DL datasets [4, 5] and by automatically conducting annotation using relevance feedback [17]). However, our study is limited by comparing only four systems that each have a relatively large absolute difference performance; an experiment that makes use of more systems (as was done in the aforementioned studies) is needed to provide conclusive evidence that the ranking of systems is stable.

5 SURVIVORSHIP BIAS IN TRAINING

Given our observation that survivability has a noticeable effect on the *evaluation* of systems that use MS MARCO, it is natural to ask whether systems *trained* on it are also affected by its survivorship bias. This is of particular concern given that it has become commonplace to perform either zero-shot transfer from MS MARCO to other datasets (e.g., [13, 21, 23]) or as an initial stage of training before fine-tuning on a target dataset [11]. In this section, we describe our experiments training a leading neural re-ranking model on a simulated set of training data that simulate a stricter annotation setup that would have resulted in more discarded queries.

5.1 Experimental Setup

Akin to our evaluation experiments, we simulate new versions of the MS MARCO dataset where annotators were given fewer documents upon which to find a possible answer. Given a minimum rank r, we filter down the official MS MARCO training triple sequence to only samples where the positive query-document pair appears within the top r documents presented to the annotator.[7] Given the high effectiveness and ease of training of the monoT5 model, we select it as a representative ranking model for this experiment. Using both $r = 1$ and $r = 5$, we train a monoT5 over 256k samples, a mini-batch size of 8, and a learning rate of 5×10^{-5}. As the baseline trained on all available data, we use the official model checkpoint released by the monoT5 authors.

We evaluate on four datasets. First, we use MS MARCO Dev (small), a commonly-used subset of 7980 queries from the full MS MARCO dev set. Since it is well-known that this dataset contains incomplete assessments [1], we also test on TREC DL 19 [5] and 20 [3], which contain 43 and 54 queries (respectively) from the hidden MS MARCO eval set with deeper relevance judgments (215 and 210 qrels per query on average, respectively, pooled from a variety of systems). Finally, to test the effect on model transferability, we test on TREC Robust04 [24], which contains 249 queries and 1251 qrels per query. We report the official evaluation measures for each dataset. For both MS MARCO-based datasets we use the passage corpus and the official initial ranking set. For Robust04, where there is no standard initial ranking set, we re-rank the top 100 BM25 results, performing MaxP aggregation over sliding passages of the documents.

---

[7] We acknowledge that in this setup, the true relevant answers that were observed above r are not considered as *negative* candidates either. Despite this, we prefer our experimental setup because it ensures as fair as possible a comparison between systems; they are trained on the same data in the same sequence, just potentially with some samples discarded. Further, since the negative documents were sampled from the top 1000 BM25 results for the official training sequence, the rate at which these false negatives would be encountered would be rather low.
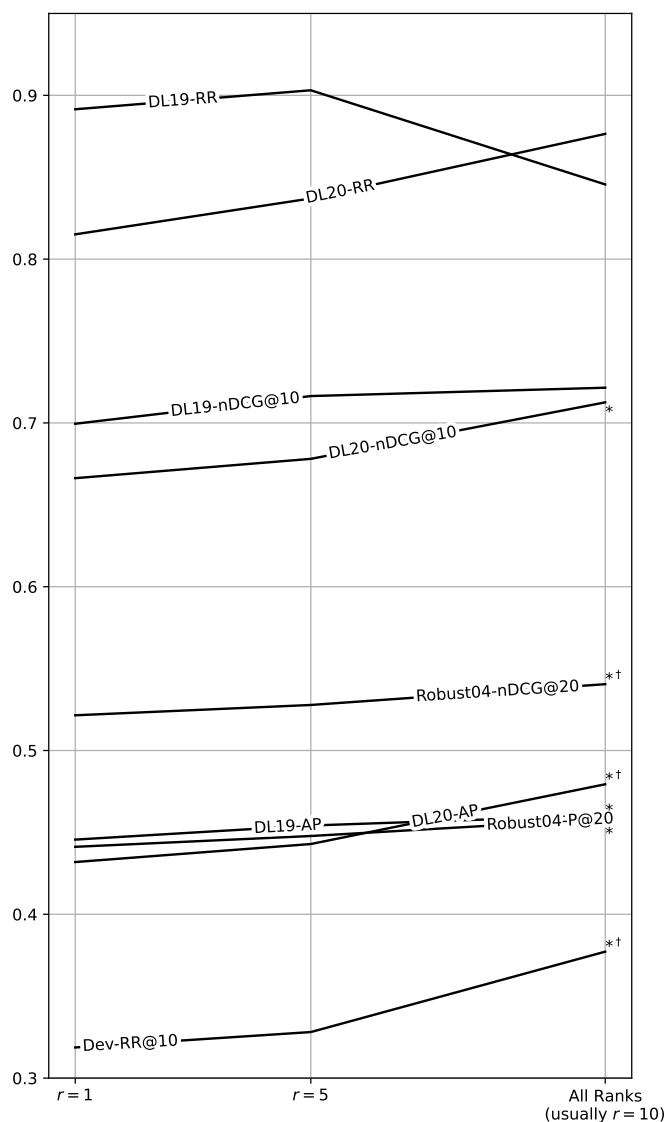

Figure 1: Effect of simulated survivorship filtering in monoT5 model training. Significant differences between the simulated settings are indicated with * ($r = 1$) and † ($r = 5$). Significance is determined using a paired t-test with $p < 0.05$ and a Bonferroni correction.

5.2 Results

Figure 1 presents the results of our training experiments. The figure shows an apparent upward trend in performance as more ranks are considered as candidates for training data. Specifically, we find that across all four datasets, there are significant differences in performance when training with $r = 1$ compared to the full dataset. Despite large absolute differences for RR on DL 19 and 20, the measure is statistically unstable [7] (especially with few queries available for TREC DL), and did not result in significance. In some cases, there is also statistically significant differences between $r = 5$ and the

full dataset (Dev RR@10, DL20 AP, and Robust04 nDCG@20). Extrapolating this trend suggests that if the MS MARCO assessments gone even deeper than the typical 10 documents (which would result in fewer discarded queries and increased survivability), performance on test collections like TREC DL and TREC Robust may be improved. This observation suggests that there may be value in annotating discarded queries in the MS MARCO *training* set, since it may improve model performance on both MS MARCO and datasets onto which models trained on MS MARCO are transferred.

6 DISCUSSION AND CONCLUSION

In this paper, we delved into the problem of survivorship bias in MS MARCO, a popular dataset used for training and evaluating IR systems. We observed that annotators could not identify answers for 38–45% of queries, which are subsequently discarded in the MS MARCO passage and document ranking tasks. Through an analysis of a sample of the discarded queries, we found that by far the most common reason that queries were discarded was simply that complete answers were presented to the annotators. We then confirmed that by pooling the results of multiple systems, answers to the majority of discarded queries could be identified. This observation clearly indicates that discarded queries are primarily the result of MS MARCO's shallow pooling annotation process. We further found that the discarded queries likely belong to a different distribution than the surviving queries, as evidenced by differences in the query type between the two sets.

Given that a major factor in query survivability is the annotation depth – had annotators inspected more documents, they likely would have been able to answer more of the queries – we simulated lower annotation depths to test the effect of the annotation depth on the training and evaluation of systems. We find that there is a marked difference in performance as the annotation depth is reduced. Importantly, we see this across a variety of ranking models, which suggests that the relative order of systems is likely preserved. Nevertheless, this makes the MS MARCO dev set a poor estimator of the *absolute* performance of systems. In terms of training, we found that training on a reduced-depth subset of MS MARCO training samples reduces the performance of a leading neural re-ranking approach – when testing in-domain, testing with more complete labels, and testing in a zero-shot setting.

We hope that this study serves as a cautionary tale of the potential for survivorship bias when using shallow pooling techniques. Even simply pooling a few candidate documents from a second system could have reduced the number of discarded queries in MS MARCO substantially. Though re-annotating MS MARCO is prohibitively expensive – especially for the train set – we recommend that discarded queries be considered in future annotation efforts, such as those at the Deep Learning Tracks or for preference assessment [1]. We further note the potential for discarded queries may be useful in pseudo-labeling settings [12], especially considering that we found existing ranking models can often effectively find answers to these queries.

ACKNOWLEDGMENTS

REFERENCES

[1] Negar Arabzadeh, Alexandra Vtyurina, Xinyi Yan, and Charles L. A. Clarke. 2021. Shallow pooling for sparse labels. arXiv:2109.00062 [cs.IR]

[2] Leonid Boytsov, Anna Belova, and Peter Westfall. 2013. Deciding on an adjustment for multiplicity in IR experiments. In *The 36th International ACM SIGIR conference on research and development in Information Retrieval, SIGIR '13, Dublin, Ireland - July 28 - August 01, 2013.* ACM, 403–412. https://doi.org/10.1145/2484028.2484034

[3] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, and Daniel Campos. 2020. Overview of the TREC 2020 Deep Learning Track. In *Proceedings of the Twenty-Ninth Text REtrieval Conference, TREC 2020, Virtual Event [Gaithersburg, Maryland, USA], November 16-20, 2020 (NIST Special Publication, Vol. 1266).* National Institute of Standards and Technology (NIST). https://trec.nist.gov/pubs/trec29/papers/OVERVIEW.DL.pdf

[4] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Jimmy Lin. 2021. MS MARCO: Benchmarking Ranking Models in the Large-Data Regime. In *SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021.* ACM, 1566–1576. https://doi.org/10.1145/3404835.3462804

[5] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Ellen M. Voorhees. 2019. Overview of the TREC 2019 Deep Learning Track. In *Proceedings of the Twenty-Eighth Text REtrieval Conference, TREC 2019 (NIST Special Publication).* National Institute of Standards and Technology (NIST).

[6] Nick Craswell, Emine Yilmaz, Daniel Campos, and Jimmy Lin. 2021. Overview of the TREC 2021 Deep Learning Track. In *Proceedings of the Thirtieth Text REtrieval Conference, TREC 2021 (NIST Special Publication).* National Institute of Standards and Technology (NIST).

[7] Marco Ferrante, Nicola Ferro, and Norbert Fuhr. 2021. Towards Meaningful Statements in IR Evaluation: Mapping Evaluation Measures to Interval Scales. *IEEE Access* 9 (2021), 136182–136216. https://doi.org/10.1109/ACCESS.2021.3116857

[8] Thorsten Joachims, Laura A. Granka, Bing Pan, Helene Hembrooke, and Geri Gay. 2005. Accurately interpreting clickthrough data as implicit feedback. In *SIGIR 2005: Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Salvador, Brazil, August 15-19, 2005.* ACM, 154–161. https://doi.org/10.1145/1076034.1076063

[9] Patricia Katopol. 2017. Maybe Best Practices Aren't: How Survivorship Bias Skews Information Gathering and Decision-Making. *Library Leadership & Management* 32, 1 (2017). https://doi.org/10.5860/llm.v32i1.7287

[10] 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 2020, Virtual Event, China, July 25-30, 2020.* ACM, 39–48. https://doi.org/10.1145/3397271.3401075

[11] Canjia Li, Andrew Yates, Sean MacAvaney, Ben He, and Yingfei Sun. 2020. PARADE: Passage Representation Aggregation for Document Reranking. arXiv:2008.09093 [cs.IR]

[12] Sheng-Chieh Lin, Jheng-Hong Yang, and Jimmy Lin. 2021. In-Batch Negatives for Knowledge Distillation with Tightly-Coupled Teachers for Dense Retrieval. In *Proceedings of the 6th Workshop on Representation Learning for NLP (RepL4NLP-2021).* Association for Computational Linguistics, Online, 163–173. https://doi.org/10.18653/v1/2021.repl4nlp-1.17

[13] Sean MacAvaney, Arman Cohan, and Nazli Goharian. 2020. SLEDGE-Z: A Zero-Shot Baseline for COVID-19 Literature Search. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP).* Association for Computational Linguistics, Online, 4171–4179. https://doi.org/10.18653/v1/2020.emnlp-main.341

[14] Sean MacAvaney, Luca Soldaini, and Nazli Goharian. 2020. Teaching a New Dog Old Tricks: Resurrecting Multilingual Retrieval Using Zero-Shot Learning. In *Advances in Information Retrieval - 42nd European Conference on IR Research, ECIR 2020, Lisbon, Portugal, April 14-17, 2020, Proceedings, Part II (Lecture Notes in Computer Science, Vol. 12036).* Springer, 246–254. https://doi.org/10.1007/978-3-030-45442-5_31

[15] Sean MacAvaney, Andrew Yates, Sergey Feldman, Doug Downey, Arman Cohan, and Nazli Goharian. 2021. Simplified Data Wrangling with ir_datasets. In *SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021.* ACM, 2429–2436. https://doi.org/10.1145/3404835.3463254

[16] Craig Macdonald, Nicola Tonellotto, Sean MacAvaney, and Iadh Ounis. 2021. PyTerrier: Declarative Experimentation in Python from BM25 to Dense Retrieval. In *CIKM '21: The 30th ACM International Conference on Information and Knowledge Management, Virtual Event, Queensland, Australia, November 1 - 5, 2021.* ACM,

4526–4533. https://doi.org/10.1145/3459637.3482013

[17] Joel Mackenzie, Matthias Petri, and Alistair Moffat. 2021. A Sensitivity Analysis of the MSMARCO Passage Collection. arXiv:2112.03396 [cs.IR]

[18] Iain Mackie, Jeffrey Dalton, and Andrew Yates. 2021. How Deep is your Learning: the DL-HARD Annotated Deep Learning Dataset. In *SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021.* ACM, 2335–2341. https://doi.org/10.1145/3404835.3463262

[19] Marc Mangel and Francisco J. Samaniego. 1984. Abraham Wald's Work on Aircraft Survivability. *J. Amer. Statist. Assoc.* 79, 386 (1984), 259–267. http://www.jstor.org/stable/2288257

[20] Massimo Melucci. 2016. Impact of Query Sample Selection Bias on Information Retrieval System Ranking. In *2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA).* 341–350. https://doi.org/10.1109/DSAA.2016.43

[21] 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 [cs.IR]

[22] Stephen E. Robertson and Hugo Zaragoza. 2009. The Probabilistic Relevance Framework: BM25 and Beyond. *Found. Trends Inf. Retr.* 3, 4 (2009), 333–389. https://doi.org/10.1561/1500000019

[23] Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. BEIR: A Heterogeneous Benchmark for Zero-shot Evaluation of Information Retrieval Models. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2).* https://openreview.net/forum?id=wCu6T5xFjeJ

[24] Ellen M. Voorhees. 2004. Overview of the TREC 2004 Robust Track. In *Proceedings of the Thirteenth Text REtrieval Conference, TREC 2004, Gaithersburg, Maryland, USA, November 16-19, 2004 (NIST Special Publication, Vol. 500-261).* National Institute of Standards and Technology (NIST). http://trec.nist.gov/pubs/trec13/papers/ROBUST.OVERVIEW.pdf

[25] Ellen M. Voorhees and Donna Harman. 2000. Overview of the Ninth Text REtrieval Conference (TREC-9). In *Proceedings of The Ninth Text REtrieval Conference, TREC 2000, Gaithersburg, Maryland, USA, November 13-16, 2000 (NIST Special Publication, Vol. 500-249).* National Institute of Standards and Technology (NIST). http://trec.nist.gov/pubs/trec9/papers/overview_9.pdf

[26] Ellen M Voorhees, Donna K Harman, et al. 2005. *TREC: Experiment and evaluation in information retrieval.* Vol. 63. MIT Press. https://mitpress.mit.edu/books/trec

[27] Ellen M. Voorhees, Ian Soboroff, and Jimmy Lin. 2022. Can Old TREC Collections Reliably Evaluate Modern Neural Retrieval Models? arXiv:2201.11086 [cs.IR]

[28] Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul N. Bennett, Junaid Ahmed, and Arnold Overwijk. 2021. Approximate Nearest Neighbor Negative Contrastive Learning for Dense Text Retrieval. In *International Conference on Learning Representations.* https://openreview.net/forum?id=zeFrfgyZln