Explain like I am BM25: Interpreting a Dense Model's Ranked-List with a Sparse Approximation

Michael Llordes
micllrs@gmail.com
University of Glasgow
Glasgow, United Kingdom

Debasis Ganguly
debasis.ganguly@glasgow.ac.uk
University of Glasgow
Glasgow, United Kingdom

Sumit Bhatia
sumit.bhatia@adobe.com
Media and Data Science Research Lab, Adobe
Noida, India

Chirag Agarwal
chiragagarwall12@gmail.com
Media and Data Science Research Lab, Adobe
Noida, India

ABSTRACT

Neural retrieval models (NRMs) have been shown to outperform their statistical counterparts owing to their ability to capture semantic meaning via dense document representations. These models, however, suffer from poor interpretability as they do not rely on explicit term matching. As a form of local per-query explanations, we introduce the notion of *equivalent* queries that are generated by maximizing the similarity between the NRM's results and the result set of a sparse retrieval system with the equivalent query. We then compare this approach with existing methods such as RM3-based query expansion and contrast differences in retrieval effectiveness and in the terms generated by each approach.

CCS CONCEPTS

• **Information systems** → **Information retrieval**; **Content analysis and feature selection**; **Retrieval models and ranking**.

KEYWORDS

Interpretability, Explainability, Neural Ranking Models

ACM Reference Format:
Michael Llordes, Debasis Ganguly, Sumit Bhatia, and Chirag Agarwal. 2023. Explain like I am BM25: Interpreting a Dense Model's Ranked-List with a Sparse Approximation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '23), July 23–27, 2023, Taipei, Taiwan.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3539618.3591982

1 INTRODUCTION

Neural retrieval models (NRM) have gained prominence, achieving state-of-the-art results on various document and passage ranking tasks [7, 10, 16, 17, 31]. NRMs are capable of modeling the semantic similarity between the query and document representations for ranking, leading to their outperforming over traditional sparse retrieval methods (e.g., BM25 [21], LM [32] etc.), which rely explicitly on term matching. However, despite their success, NRMs suffer from poor *interpretability* of their results [1]. With the increasing deployment of more complex NRMs, it is essential to explain the retrieval decisions of a "black-box" complex model to its end-users, thereby increasing their trust in the model [18].

Given a query, while it is straightforward to understand why a document is retrieved by a sparse retrieval model as the relevance score depends on the explicit presence of the query (or expansion) terms, the results produced by NRMs are hard to interpret as these models rely on the closeness of query and document representations in the embedding space. This inherent opaqueness of NRMs makes them potentially non-trustworthy to end-users, especially in critical domains such as healthcare, finance, and law [1, 33]. Approaches to explaining NRMs include providing additional information such as informative snippets [4] and key noun phrases representing query aspects [19], visualizing regions in the embedding space that affect the model output [5] and personalized explanations based on user characteristics [26]. Furthermore, it has been found that explanations generated by popular interpretabilty models such as LIME [20] and SHAP [8, 13] vary widely raising concern about the robustness and utility of the explanations produced by directly applying these interpretability methods to NRMs [8].

Our Contributions: We propose an intuitive means of explaining the output of an NRM by introducing the notion of an *equivalent query*, which we define as follows. Given a query Q, the equivalent query Q^+ is the query which when executed on a sparse retrieval model ideally produces the same ranked list as produced by an NRM with Q, the original query. We posit that the equivalent query offers a minimalist and intuitive explanation of the "thought process" of an NRM. Since a sparse IR model relies on explicit term matching, a query that produces the same (or close enough) results to that of a complex NRM reveals the semantic concepts considered implicitly by the NRM, and thus potentially helps to interpret its behavior.

As an illustrative example, consider the query *what is the most popular food in switzerland* from the TREC-DL'19 topic set. The equivalent queries produced by our proposed method for MonoT5 [17] (an NRM) is '*dish food includ serv switzerland vacherin*' whereas for DCT [7] (another NRM) this equivalent query is '*appenzel food german meat neighbor popular*'. The interesting point

SIGIR '23, July 23–27, 2023, Taipei, Taiwan

ACM ISBN 978-1-4503-9408-6/23/07…$15.00
https://doi.org/10.1145/3539618.3591982

is that 'appenzeler' and 'vacherin' are two different varieties of Swiss cheese; the two equivalent queries produced by our method unravel the different concepts and terms used by these two models for retrieving the respective top documents.

While the notion of an 'equivalent query' may seem somewhat similar to adding terms to the original query via pseudo-relevance feedback (PRF), there are two major differences. First, expansion terms in PRF are simply the informative terms occurring in the top-documents, whereas the process of generating equivalent queries explicitly seeks to make the output of a sparse model locally similar to that of a target NRM. Second, unlike PRF expansion, terms from the original query may also be absent in the equivalent query.

Constructing the equivalent queries as explanations for a particular NRM is non-trivial as finding such an optimal query is an optimal subset-selection problem, which is NP-complete (Section 2). We adopt a discrete state-space exploration method to find solution states (expanded queries) which when executed on a sparse index maximises the overlap of the top-documents retrieved with the NRM for which explanations are sought. Our experiments on the MS-MARCO [15] reveal that the generated equivalent queries, on an average, achieve a fidelity score (RBO [29]) of up to 0.5194, and these equivalent queries when executed on BM25 lead up to obtaining 96% of the nDCG values of the target NRM.

2 PROPOSED METHODOLOGY

Problem Formulation. Let θ be a neural ranking model (NRM) that, given a query Q, retrieves an ordered set of documents $L_k(Q;\theta)$. Each document $D \in L_k(Q;\theta)$ is sorted in the decreasing order of the parameterized scores $\theta(Q,D)$ that measure the relevance of D to Q. The dense model θ can either re-rank an initial list produced by a sparse retriever (e.g. BM25) [10], or retrieve the list directly from the underlying corpus in an end-to-end manner [31]. Recall that our goal is to make use of a sparse retrieval model, which we denote as ϕ (e.g., BM25 [21] or LM [32]) to approximate the retrieval output of an NRM θ. Note that this idea is analogous to the existing work on developing local surrogate models to explain the behavior of complex black-box models [3, 13, 20].

The sparse IR model, when fed with the equivalent query to approximate the results of an NRM, offers a surrogate model for explaining the behavior of the target NRM. This is because the sparse model i) leverages discrete terms instead of embeddings, and ii) the scoring function $\phi(Q,D)$ is a closed form expression of several basic components, such as the term frequencies and IDFs of the matching terms, the length of D, etc. [24, 28].

There is one subtle difference of explaining document ranking from the per-instance based local explanation models for classifiers (e.g., LIME [20] or SHAP [13]) that explain the output by inducing weights over input features reflecting their relative influence on the predicted outcome. In the context of document ranking, similar ideas have been explored to measure the impact of individual terms on relative changes in the document scores [27].

In our setting, the key difference is that the *locality for approximation* to estimate term influence is not restricted to individual instances of query-document pairs. Instead, the approximator works at the level of a query and the top-k set of documents retrieved with a deep neural model. Due to this difference in the granularity of locality, the explanations generated do not correspond to the influence weights of words from individual documents, but rather they correspond to the words from the top-retrieved set. Specifically, we seek to find those terms which when added to the original query Q will effectively bridge the vocabulary gap of the sparse model ϕ and make its output similar to that of the NRM (θ) for Q. Formally speaking, the *equivalent query* Q^+ is the one that satisfies the following objective:

$$\underset{Q^+ \subset V(L_k(Q;\theta))}{\operatorname{argmax}} \quad \omega(L_k(Q^+;\phi), L_k(Q;\theta)), \tag{1}$$

where $V(L_k(Q;\theta))$ represents the vocabulary of the top-k documents retrieved with the query Q using the model θ, and ω is a similarity measure, e.g., the set-based Jaccard metric or the rank-based RBO metric [30] between the two ranked lists of documents (see also [25] which explored this idea of overlap between document lists retrieved with two different queries for measuring trustworthiness of NRMs). The output query, Q^+, thus obtained, can be interpreted as the set of terms, or *concepts*, that the black-box NRM takes into account in its computation of the top-k list. While in reality, θ works in the embedded (continuous) space, a discrete realisation of this concept set via this approximation helps gain insights into the behavior of θ, which would be useful to end-users and model practitioners.

Discrete State-Space Optimisation. Observe that Equation 1 is an optimal subset selection problem, which is NP-complete. A practical approach is to employ a standard discrete state-space exploration method, such as the Best First Search (BFS) exploration [23] that involves traversing a state-space transition tree by dynamically selecting a depth-first or breadth-first strategy via a heuristic. We now describe the state-space, the actions and the heuristics employed in the BFS exploration for our task.

States and Evaluation Function. In the context of our problem, a state refers to a query Q^+ which is executed by the sparse model ϕ to retrieve a top-k list. The goal state for this approximation problem is to find an optimal query Q^* which retrieves a top-k set identical to the black-box model θ. Thus, the evaluation function for a state, which measures how close a state is to the goal state, is the overlap measure shown in Equation 1.

Actions. We consider two types of actions - one that involves adding terms to an existing query of a state to generate a new query state, and the other that involves removing a term to generate a new state. An important decision to be made is to determine the set of candidate terms to be added to an existing state to create a new state. Although, in principle, one can consider the entire set of vocabulary terms, such an approach would lead to a substantially large branching factor for the tree-based exploration. A large volume of work on pseudo-relevance feedback (PRF) in IR has shown that most terms that are semantically related to the information need mostly occur within the top-k set of documents retrieved [2, 14, 22]. Thus, it is reasonable to restrict the set of candidate terms to the vocabulary of this set, denoted as $V(L_k(Q;\theta))$ in Equation 1.

To define the first type of transition, i.e., the one that involves adding a term t to transition from Q_i to $Q_{i+1} = Q_i \cup \{t\}$ (i being the depth of the BFS exploration tree), is given by the normalized probability of the RM3 [11] weights. In other words, the higher the

RM3 weight of a term $t \in V(L_k(Q_{i+1}; \theta))$, the higher is the likelihood of exploring along the branch Q_{i+1}. Similarly, for generating a new query (state) with one term removed from the current state, we set the probability of removing a term as *inversely proportional* to its tf-idf weight with the rationale of retaining the informative terms within a query.

Exploration Heuristic. For the BFS exploration, we start exploring from the root state - the empty query $\emptyset$. Exploring a state involves generating b child states by choosing one of the two actions randomly: add or remove (b denotes the maximum branching factor parameter). A child state is only added to the tree if it has not been generated before. throughout the tree exploration phase, we keep track of the child states, i.e., queries generated thus far.

Given a set of current unexplored nodes $S \in \mathcal{U}$, the next state considered for exploration (action of adding or removing terms to generate newer queries) is the state (say $S^* \in \mathcal{U}$) with the best value of the evaluation function, i.e., $\text{argmax}_{S \in \mathcal{U}}\, \omega(L_k(S; \phi), L_k(Q; \theta))$. An advantage of the best-first exploration is that it is able to continue exploring along a promising direction at greater depths, or is able to back-track to expand yet unexplored states at lower depths. During the execution of the algorithm, we keep track of the best state discovered and output it at the end of the exploration. The exploration itself is limited by the maximum number of depths, which we set to 10 in our experiments. The termination condition for the algorithm is given by this maximum number of depths, or when there is no state left to explore.

3 EXPERIMENT SETUP

Research Questions and Dataset. The objective of our experiments is to see how effectively can we approximate the top-retrieved documents of an NRM θ by a sparse model ϕ, i.e.,

- **RQ1**: How well our proposed method of BFS-based tree exploration approximates several black-box models (e.g., ColBERT [10], ANCE [31] etc.) with different modes of operation (sparse with reranking, or end-to-end dense with approximate search)?

Since the output of a model-aware local approximation is a set of additional terms, which are supposed to be those on which the target NRM puts emphasis, the next question to investigate is:

- **RQ2**: Can these additional terms, on top of help interpreting θ, can also help to improve the IR effectiveness of sparse models?

As the dataset for our experiments, we use the MS-MARCO passage ranking collection [15] and the TREC DL 2019 topic set [6].

Baselines. Since we propose to use BFS to solve the discrete state-space optimisation of optimal subset selection, we compare it with other computationally less intensive approaches, such as the greedy search. In the greedy exploration of the state-space, we generate b branches similar to the BFS method; however, we keep on exploring only along the best branch every time without saving the other branches for back-tracking purposes. Also, similar to the BFS method, in our greedy baseline we restrict the exploration to a maximum number of states and output the best state discovered during the exploration as the optimal solution. This baseline uses the same overlap-based state evaluation as used by BFS.

In addition to the *model-aware* greedy baseline, we also employ a *model-agnostic* baseline method, namely RM3-based query expansion [11], to find out how much of an overlap can a model-agnostic method such as RM3 on a sparse index achieve with the top-retrieved set obtained by an NRM. Note that the output of this method cannot be used as a model-specific explanation, and merely serves as a reference point for the overlap comparison.

Parameter Details. To solve the optimisation of Equation 1, for the greedy approach we set the maximum number of unique states visited to 1000, whereas for BFS, we set the tree depth to 10. The branching factor b of the BFS exploration was set to 30. We set the parameter k of Equation 1, which controls the locality of the explanations, to a value of 10 in all our experiments. A small value of $k = 10$ ensures that the objective is to approximate only the first search result page of a black-box model [9].

Evaluation Metrics. As a measure of how closely the sparse approximation fits a black-box model, i.e., as a fidelity measure, we report the RBO and Jaccard based overlaps between the top-10 documents retrieved with the original query by θ (the black-box) and those retrieved with the expanded query by BM25 (ϕ, the approximator). As IR evaluation measures, we report the MAP (relevance of at least 2 as per the TREC DL guidelines [6]) and the nDCG values for the top-10 results. A value of $k = 10$ was used to optimise the sparse approximation objective (see Equation 1).

Black-box and the Sparse Approximator Models. As the sparse model for approximation we employ BM25, i.e., $\phi = \{\text{BM25}\}$ with the standard settings of $k_{\text{BM25}} = 1.2$ and $b_{\text{BM25}} = 0.75$. As a concrete realisation of the state evaluation function ω of Equation 1, we use RBO, which is a rank-based overlap metric [30]. In our results, we report both Jaccard and RBO measures computed between the top-retrieved sets of θ and ϕ. As the black-box model θ, we employ a number of neural models operating in both the 'sparse + reranking' and the dense end-to-end modes [12]. Specifically, we used ANCE [31], ColBERT (CBERT) [10], and MonoT5 [17] as representative dense end-to-end models. Additionally, we employed two sparse-reranking based models, a DeepCT [7] augmented index with ColBERT (DCT), and ColBERT followed by a BERT-based query expansion (QE_{BERT}). Our implementation is available at https://github.com/micllordes/eliBM25.

4 RESULTS

Main observations. Tables 1 and 2 present the results of our experiments on approximating neural models with the sparse approximator - BM25; following are the interesting observations. *First*, in relation to RQ1, we observe that our proposed BFS-based optimisation consistently outperforms the baseline greedy approach by a large margin in terms of both the fidelity and the approximation, and also in terms of the quality of the search list retrieved with the model-aware expanded query Q^+ on BM25 (Table 2).

Second, in relation to RQ2, we observe that the MAP and the nDCG values obtained via approximation are close to the values obtained with the black-box models themselves, e.g., compare the MAP value of 0.2064 obtained with the sparse approximation to that of 0.2182 obtained with CBERT (Table 1). Another important observation in relation to RQ2 is that the BFS-based model-specific approximation yields queries that are qualitatively better than the ones generated by an unsupervised relevance feedback model, such

Table 1: Results of sparse approximation of NRMs in terms of IR effectiveness. The best results across each method (Greedy and BFS) are bold-faced and the ones across each column are under-lined. The gray cells corresponding to the sparse retrieval rows (BM25 and RM3) mean that these models, being sparse ones themselves, are not approximated by the baseline (greedy) or the proposed approach (BFS). All nDCG values obtained with the BFS approach show statistically significant differences compared to the baselines.

Model				Retrieval quality with BM25(Q^+)			
				Greedy (Baseline)		BFS (Ours)	
Type	IR Model	MAP	nDCG	MAP	nDCG	MAP	nDCG
Sparse	BM25	0.1067	0.4601				
	RM3	0.1411	0.4931				
Rerank	DCT	0.2192	0.7006	0.1544	0.5082	**0.2050**	**0.6540**
	QE_{BERT}	0.2199	0.7015	0.1414	0.4848	**0.2065**	**0.6760**
E2E	ANCE	0.1836	0.6537	0.1454	0.5511	**0.1723**	**0.6049**
	CBERT	0.2182	0.6934	0.1506	0.5057	**0.2064**	**0.6474**
	MonoT5	0.2184	0.7300	0.1470	0.5355	**0.1920**	**0.6623**

Table 2: Evaluating sparse approximation of dense black-box models in terms of fidelity (overlap). The naming and bold-face/underline conventions are the same as that of Table 1. All fidelity scores from our BFS approach show statistically significant differences from their corresponding BM25, RM3 and Greedy baselines.

Model		Fidelity (Overlap) Measures							
		BM25(Q)		RM3		Greedy (Baseline)		BFS (Ours)	
Type	IR Model	Jac	RBO	Jac	RBO	Jac	RBO	Jac	RBO
Rerank (CBERT)	DCT	0.1883	0.1173	0.1694	0.0956	0.3140	0.2207	**0.5194**	**0.4946**
	QE_{BERT}	0.1621	0.1144	0.1420	0.0916	0.2956	0.2213	**0.5111**	**0.5015**
E2E	ANCE	0.1501	0.0952	0.1256	0.0753	0.3106	0.2239	**0.4993**	**0.4969**
	CBERT	0.1641	0.1155	0.1442	0.0907	0.2974	0.2230	**0.5046**	**0.4888**
	MonoT5	0.1835	0.1120	0.1715	0.0970	0.3041	0.2224	**0.5327**	**0.5194**

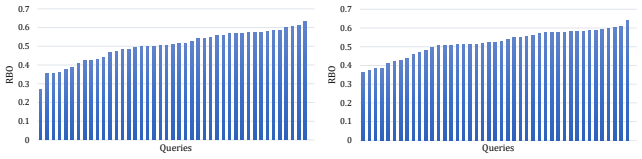

Figure 1: Fidelity (RBO overlap of top-10 documents) of the sparse approximation of the best performing models among the reranking ones - CBERT+BERTQE (*left*), and the best of the end-to-end ones - MonoT5 (*right*). The queries are sorted by the RBO values.

as RM3. This can be seen by comparing the MAP and the nDCG values of RM3 with the ones obtained by BFS approximation. *Third*, we see that high fidelity scores correlate well with the downstream task (retrieval) performance obtained via the approximation, which also indicates that the enriched query acts as a meaningful explanation of the model-specific influence of term weights.

Per-query statistics. Figure 1 shows that a majority of the queries exhibit relatively high RBO values indicating that most of the queries are good candidates for model-specific explanations via our proposed discrete state-space optimisation. We believe that queries with small fidelity scores are a result of the limitation of the size of the vocabulary $V(L_k(Q;\theta))$ as described in Section 2.

Table 3: Sample equivalent queries Q^+ generated by optimising Equation 1 (BFS-Gen). The BFS-Gen queries yield substantially better retrieval effectiveness than the original queries, or the RM3-expanded ones (RM3-Exp).

Query	Example Queries	Fidelity		
Type	(stemmed words)	Jac	RBO	nDCG
Original	Who formed the commonwealth of independent states	0.1764	0.3480	0.3695
RM3-Exp	british commonwealth countri form independ nation republ soviet state union	0.2500	0.3565	0.3034
BFS-Gen	commonwealth soviet union state form	0.5385	0.3934	0.6401
Original	what is durable medical equipment consist of	0.4286	0.1350	0.5739
RM3-Exp	benefit consist dme durabl equip ill item medic patient therapeut	0.6670	0.2039	0.8533
BFS-Gen	medic consist equip item patient	1.0000	0.2126	0.8807

Example queries generated. As an additional analysis, we now present in Table 3 some sample queries generated by the sparse approximation method and compare those with the RM3-based expanded ones. The first set of rows in Table 3 presents a situation, where the sparse equivalent query Q^+ is a subset of the RM3 terms. This particular example is notable as the BFS-generated equivalent query correctly extracts terms related to the true information need of the original query from the RLM expanded one, i.e., eliminating terms related to the Commonwealth of British origin, and retaining the ones related to the Commonwealth with that of Soviet origin.

The second group of rows in Table 3, shows an instance of BFS-generated query which achieved a Jaccard score of 1, i.e., BM25 when executed with the equivalent query Q^+ yields the identical top-10 documents as MonoT5 does with the original query. This shows that the equivalent query, in this case, represents an effective explanation of the term semantics with which MonoT5 operates for the original query.

5 CONCLUSIONS AND FUTURE DIRECTIONS

We considered the task of explaining the results of an NRM and introduced the notion of an equivalent query – one that when executed on a sparse retrieval model can lead to similar results as a complex NRM. We formulated the problem as an optimal subset selection problem (NP complete) and proposed a BFS-based state-space exploration method to approximate equivalent queries. Our empirical results on MS-MARCO benchmark show that the equivalent queries produced by our solution can approximate the top-k results of NRMs such as ColBERT and ANCE (Section 4), and the resulting queries unravel how the NRMs interpreted the user queries. Further, the equivalent queries when executed on a BM25 ranker achieved retrieval performance close to the complex NRMs.

Our proposed framework of equivalent queries offers a simple and intuitive interpretation of complex black-box retrieval models. One limitation of our solution is high latency due to the exploration of the state space (average latency of $\approx$ 6 seconds). Our future work will focus on exploring different subset selection methods to reduce the number of retrieval operations on the index needed by the search; exploring reinforcement learning for improving the latency at run-time by learning optimal state transitions. Further, another interesting direction is to use the equivalent queries for extracting better snippets from the retrieved documents.

REFERENCES

[1] Avishek Anand, Lijun Lyu, Maximilian Idahl, Yumeng Wang, Jonas Wallat, and Zijian Zhang. 2022. Explainable Information Retrieval: A Survey. https://doi.org/10.48550/ARXIV.2211.02405

[2] Anirban Chakraborty, Debasis Ganguly, and Owen Conlan. 2020. Retrievability based Document Selection for Relevance Feedback with Automatically Generated Query Variants. In *CIKM*. ACM, 125–134.

[3] Jianbo Chen, Le S., Martin W., and Michael Jordan. 2018. Learning to Explain: An Information-Theoretic Perspective on Model Interpretation. In *Proc. of ICML'18*, Vol. 80. 882–891.

[4] Ioannis Chios and Suzan Verberne. 2021. Helping results assessment by adding explainable elements to the deep relevance matching model. 3rd International Workshop on ExplainAble Recommendation and Search (EARS 2020). https://doi.org/10.48550/ARXIV.2106.05147

[5] Jaekeol Choi, Jungin Choi, and Wonjong Rhee. 2020. Interpreting Neural Ranking Models using Grad-CAM. https://doi.org/10.48550/ARXIV.2005.05768

[6] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, Ellen M Voorhees, and Ian Soboroff. 2021. TREC deep learning track: Reusable test collections in the large data regime. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*. 2369–2375.

[7] Zhuyun Dai and Jamie Callan. 2020. Context-Aware Term Weighting For First Stage Passage Retrieval. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval* (Virtual Event, China) *(SIGIR '20)*. Association for Computing Machinery, New York, NY, USA, 1533–1536. https://doi.org/10.1145/3397271.3401204

[8] Zeon Trevor Fernando, Jaspreet Singh, and Avishek Anand. 2019. A Study on the Interpretability of Neural Retrieval Models Using DeepSHAP *(SIGIR'19)*. Association for Computing Machinery, New York, NY, USA, 1005–1008. https://doi.org/10.1145/3331184.3331312

[9] Diane Kelly and Leif Azzopardi. 2015. How many results per page?: A Study of SERP Size, Search Behavior and User Experience. In *SIGIR*. ACM, 183–192.

[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*. 39–48.

[11] Victor Lavrenko and W. Bruce Croft. 2001. Relevance Based Language Models. In *Proc. of SIGIR '01*. 120–127.

[12] Hang Li, Shuai Wang, Shengyao Zhuang, Ahmed Mourad, Xueguang Ma, Jimmy Lin, and Guido Zuccon. 2022. To Interpolate or not to Interpolate: PRF, Dense and Sparse Retrievers. In *SIGIR*. ACM, 2495–2500.

[13] Scott M Lundberg and Su-In Lee. 2017. A Unified Approach to Interpreting Model Predictions. In *Advances in Neural Information Processing Systems*, I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Eds.), Vol. 30. Curran Associates, Inc. https://proceedings.neurips.cc/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf

[14] Ali Montazeralghaem, Hamed Zamani, and James Allan. 2020. A Reinforcement Learning Framework for Relevance Feedback. In *SIGIR*. ACM, 59–68.

[15] Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. MS MARCO: A human generated machine reading comprehension dataset. *choice* 2640 (2016), 660.

[16] Rodrigo Nogueira, Jimmy Lin, and AI Epistemic. 2019. From doc2query to docTTTTTquery. *Online preprint* 6 (2019).

[17] Ronak Pradeep, Rodrigo Nogueira, and Jimmy Lin. 2021. The expando-mono-duo design pattern for text ranking with pretrained sequence-to-sequence models. *arXiv preprint arXiv:2101.05667* (2021).

[18] Jiaming Qu, Jaime Arguello, and Yue Wang. 2021. A Deep Analysis of an Explainable Retrieval Model for Precision Medicine Literature Search. In *Advances in Information Retrieval: 43rd European Conference on IR Research, ECIR 2021*.

[19] Razieh Rahimi, Youngwoo Kim, Hamed Zamani, and James Allan. 2021. Explaining Documents' Relevance to Search Queries. https://doi.org/10.48550/ARXIV.2111.01314

[20] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (San Francisco, California, USA) *(KDD '16)*. Association for Computing Machinery, New York, NY, USA, 1135–1144.

[21] S.E. Robertson, S. Walker, M.M. Beaulieu, M. Gatford, and A. Payne. 1996. Okapi at TREC-4.

[22] Dwaipayan Roy, Debasis Ganguly, Mandar Mitra, and Gareth J. F. Jones. 2016. Word Vector Compositionality based Relevance Feedback using Kernel Density Estimation. In *CIKM*. ACM, 1281–1290.

[23] Stuart J. Russell and Peter Norvig. 2009. *Artificial Intelligence: a modern approach* (3 ed.). Pearson.

[24] Procheta Sen, Debasis Ganguly, Manisha Verma, and Gareth J. F. Jones. 2020. The Curious Case of IR Explainability: Explaining Document Scores within and across Ranking Models. In *SIGIR*. ACM, 2069–2072.

[25] Procheta Sen, Sourav Saha, Debasis Ganguly, Manisha Verma, and Dwaipayan Roy. 2022. Measuring and Comparing the Consistency of IR Models for Query Pairs with Similar and Different Information Needs. In *CIKM*. ACM, 4449–4453.

[26] Suzan Verberne. 2018. Explainable IR for Personalizing Professional Search. In *Joint Proceedings of the First International Workshop on Professional Search (ProfS2018); the Second Workshop on Knowledge Graphs and Semantics for Text Retrieval, Analysis, and Understanding (KG4IR); and the International Workshop on Data Search (DATA:SEARCH'18) Co-located with (ACM SIGIR 2018), Ann Arbor, Michigan, USA, July 12, 2018 (CEUR Workshop Proceedings, Vol. 2127)*, Laura Dietz, Laura Koesten, and Suzan Verberne (Eds.). CEUR-WS.org, 35–42. http://ceur-ws.org/Vol-2127/paper4-profs.pdf

[27] Manisha Verma and Debasis Ganguly. 2019. LIRME: Locally Interpretable Ranking Model Explanation. In *Proc. of SIGIR 2019*. 1281–1284.

[28] Michael Völske, Alexander Bondarenko, Maik Fröbe, Benno Stein, Jaspreet Singh, Matthias Hagen, and Avishek Anand. 2021. Towards Axiomatic Explanations for Neural Ranking Models. In *Proceedings of the 2021 ACM SIGIR International Conference on Theory of Information Retrieval* (Virtual Event, Canada) *(ICTIR '21)*. Association for Computing Machinery, New York, NY, USA, 13–22. https://doi.org/10.1145/3471158.3472256

[29] William Webber, Alistair Moffat, and Justin Zobel. 2010. A Similarity Measure for Indefinite Rankings. *ACM Trans. Inf. Syst.* 28, 4, Article 20 (2010).

[30] William Webber, Alistair Moffat, and Justin Zobel. 2010. A Similarity Measure for Indefinite Rankings. *ACM Trans. Inf. Syst.* 28, 4, Article 20 (nov 2010), 38 pages. https://doi.org/10.1145/1852102.1852106

[31] Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul Bennett, Junaid Ahmed, and Arnold Overwijk. 2020. Approximate nearest neighbor negative contrastive learning for dense text retrieval. *arXiv preprint arXiv:2007.00808* (2020).

[32] Chengxiang Zhai and John Lafferty. 2001. A Study of Smoothing Methods for Language Models Applied to Ad Hoc Information Retrieval. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '01)*. Association for Computing Machinery, New York, NY, USA, 334–342.

[33] Yongfeng Zhang, Yi Zhang, and Min Zhang. 2018. SIGIR 2018 Workshop on ExplainAble Recommendation and Search (EARS 2018). In *The 41st International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '18)*. ACM.