Model Agnostic Interpretability of Rankers via Intent Modelling

Jaspreet Singh
L3S Research Center
Hannover, Germany
singh@l3s.de

Avishek Anand
L3S Research Center
Hannover, Germany
anand@l3s.de

ABSTRACT

A key problem in information retrieval is understanding the latent intention of a user's under-specified query. Retrieval models that are able to correctly uncover the query intent often perform well on the document ranking task. In this paper we study the problem of interpretability for text based ranking models by trying to unearth the query intent *as understood by complex retrieval models*.

We propose a model-agnostic approach that attempts to locally approximate a complex ranker by using a simple ranking model in the term space. Given a query and a blackbox ranking model, we propose an approach that systematically exploits preference pairs extracted from the target ranking and document perturbations to identify a set of intent terms and a simple term based ranker that can faithfully and accurately mimic the complex blackbox ranker in that locality. Our results indicate that we can indeed interpret more complex models with high fidelity. We also present a case study on how our approach can be used to interpret recently proposed neural rankers.

ACM Reference Format:
Jaspreet Singh and Avishek Anand. 2020. Model Agnostic Interpretability of Rankers via Intent Modelling. In *Conference on Fairness, Accountability, and Transparency (FAT* '20), January 27–30, 2020, Barcelona, Spain.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3351095.3375234

1 INTRODUCTION

In the context of data-driven models, interpretability can be defined as "*the ability to explain or to present in understandable terms to a human*" [14]. Recently, in the machine learning (ML) and NLP communities there has been growing interest in the interpretability of ML models [5, 24, 27] but there has been limited work on interpreting retrieval/ranking models considered central to information retrieval (IR).

Ranking models are used in a variety of domains. The most traditional use is in a variety of search engines for differing or mixed modalities: images, video, Web (news, web pages, social media posts, patents, scientific papers) and data. In these scenarios, the ranking model is responsible for ranking search results given a query. Ranking models are now pervasive in recommender systems as well where no user generated query is apparent. Instead they are used for ranking similar items in application areas like shopping, movies, music, news and social media or prioritizing information for a day like in a news feed. Machine learned ranking models are also used in systems where a ranking is not the final output. For example, ranking models are used to order candidate answers in question answering systems which are then consumed by other models to select or generate answers.

When using machine learning models to rank results, the training data (clicks, human annotations) informs how signals/features should be combined. Various models have been employed, ranging from linear regression and decision trees to deep neural networks [19, 26, 30, 31] more recently. Document relevance ranking, also known as adhoc retrieval or search[55], is the task of ranking documents from a large collection using the query and the text of each document only. Ranking based on textual features is particularly important in cold-start learning scenarios when there is no existence of click-logs and features like link-structure are non-informative. Examples include various domains in digital libraries, e.g., patents [3] or scientific literature [52]; enterprise search [18]; and personal search [8].

In this paper, we tackle the problem of post-hoc interpretability of adhoc rankers. Post-hoc here means that we try to explain a trained ranking model. There are 2 major challenges when explaining ranking models in particular that we seek to address in this work:

(1) Ranking models ideally output a single ranked list of documents for a given query. In practice however, rankings are an *aggregation of decisions* – (i) individual document relevance scores are first assigned and then the documents are sorted to get a list or (ii) pairwise document preferences are combined using approaches such as [1].
(2) This in turn makes evaluating the explanation of a ranking model more difficult. What kind of explanation is best suited to a ranked list of documents given a query? How do we measure if this explanation is accurate?

Interpreting intents in rankings. We first ponder on what interpreting a ranking really means. Are we only interested in explaining why a document was relevant or ranked above another ? Although that is certainly interesting, what we are really interested in is if a model is performing in accordance with the *information need* of the user who issues the (usually under-specified) query – a key concept in IR. In other words, what is the actual *query intent* as understood by the trained ranking model.

There are two key benefits to uncovering the learned intent. First, users can immediately identify biases induced by the training data or learning procedure. For example, consider the query *how to find the mean* and two different rankers:

Ranker 1 Top Doc. `Find what they mean in the following songs`

FAT '20, January 27–30, 2020, Barcelona, Spain*

ACM ISBN 978-1-4503-6936-7/20/02…$15.00
https://doi.org/10.1145/3351095.3375234

Ranker 2 Top Doc. `Tutorial on using SPSS to find mean and other stats`

Both rankers seemingly prefer very different documents even though they both contain the query terms. However why they choose different top documents is unclear. If we present the user with terms that encode the intent then this becomes easier to discern. The intent of **Ranker 1** is {`actually, say, want, meant`} and that of **Ranker 2** {`x, statistics, plus, know`}. Ranker 1 prefers documents which are semantically similar to a different sense of the word *mean* whereas Ranker 2 accurately captures the query intent.

The second benefit is being able to help identify over-fitting or under-performance. If spurious patterns like copyright messages are identified as part of the intent, developers can quickly rectify the data and improve the model. While terms are easy to understand as explanations, it may not be enough to accurately answer two important questions: Why is A relevant? Why is A ranked above B? Hence we additionally need an explanation model that can easily show how a document is relevant to the query using the intent terms.

Our Contribution. We take a *local* view to the problem of post-hoc interpretation of a black-box ranking model, i.e. interpreting the output for a single query at a time. Our approach is to learn a simple ranking model, with an expanded query, that approximates the original ranking. In doing so, we argue that the expanded query forms the intent or explanation of the query *as perceived by the black-box ranker*. In coming up with the explanation or the intent we hypothesize that an expanded query along with a simpler and interpretable model, is an accurate interpretation of the black-box model if it produces a similar ranking to the ranking of the black-box model. Towards this we exploit the pairwise preferences between documents in the original ranking to extract relevant terms to be considered for expansion by postulating a combinatorial optimization problem called the Maximum Preference Coverage.

Experimental Evaluation. To quantitatively evaluate the goodness of our explanations, we need to collect ground truth explanations given a model. Note that this is not always possible for ranking models that do not have a query understanding procedure that describes the inferred intent. In fact most learning to rank and neural models do not have an explicit query understanding component. In order to overcome these hurdles, we devised blackbox ranking models that explicitly perform query understanding using term expansions. The expansion sets then serve as ground truth explanations. We conducted extensive experiments using queries and documents from the TREC test collections. Our results indicate that we can interpret well understood explicit query understanding based retrieval models with good accuracy and high fidelity (a measure of how closely we can reproduce the behavior of a model). In a case study we also show how our framework can be employed to explain the results of neural ranking models with high fidelity in various scenarios.

2 PRELIMINARIES

Intents: A query in IR is an encoding of a user's intent. A subtle point is that the relevance of a document to a query is driven by the intent and not its encoding. In traditional relevance evaluation (like TREC [1]), trained judges are first presented with a topic description that describes the user's information need clearly rather than just the keyword query submitted to the search engine.

For example, in the TREC web track dataset from 2014, query id 283 is *hayrides in pa*. The intent/topic description is *Where can I go on a hay ride in Pennsylvania?* which is the information assessors use to judge document relevance. Note here that the description details the true intention – pa stands for Pennsylvania and the user is more interested in where to go for a hay ride rather than a history of hayrides for instance.

Keyword queries tend to be short and often ambiguous and hence identifying the intent behind them is key to determining relevance and in turn a good ranking.

Query Expansions: Early on IR researchers used query expansion techniques such as [9, 45] to explicitly model the intent of the query through terms. The under-specificity is treated by adding more terms to the query unbeknownst to the user. For the aforementioned example, terms such as *location distance philadelphia pennsylvania hayrack wagon* can provide crucial context to aid the retrieval model.

These terms can be mined from an external source [40, 57] or from the pseudo relevant documents, i.e. driven by the collection [32, 44, 54]. More recently neural approaches to query modelling have been proposed – either by using word embeddings to identify terms for expansions [13, 31] or converting the discrete query to a vector representation in a continuous space [37, 47]. The objective behind all of this work is to be able to get a better specification of the query so that (i) more context is available to estimate document relevance and (ii) simpler term based retrieval models can be used for ranking.

ML ranking models: Machine learned models rank documents based on a representation of queries and documents that can be either hand crafted (as is the case for traditional learning to rank [28]) or learned (as in the case of neural ranking models [11, 17, 34, 38]). These models learn query-document interaction patterns from the training data in order to estimate the relevance or ranked order of documents given a query. Here the intent behind a query is derived/learned from the training data, i.e. the documents marked relevant or irrelevant for keyword queries. The use of neural ranking models in adhoc search has shown significant improvements but we lack the ability to explain individual decisions made by them. These models in particular learn a new representation space for queries and documents which is difficult to understand. This however is not an issue with the more explicit query modeling approaches where a query is expanded to better specify the intent.

In this work, we detail how to locally explain complex ranking models by estimating a simpler query expansion based model where the intent behind the output ranking is made explicit.

3 RELATED WORK ON INTERPRETABILITY

Interpretability in machine learning has been studied extensively in classical machine learning. However, the success of neural networks (NN) and other expressive yet complex ML models have only intensified the discussion.

[1] https://trec.nist.gov/

On one hand they have largely improved performance, but on the other they tend to be opaque and less interpretable. Consequently, interpretability of these complex models has been studied in various other domains to better understand decisions made by the network – image classification and captioning [10, 48, 56], sequence to sequence modeling [2, 27], recommender systems [7] etc.

Post-hoc methods for interpretability can be categorized into two broad classes: *model introspective* and *model agnostic*. Model introspection refers to either interpretability by design or access to all the model parameters. In interpretability by design, more inherently "interpretable" models such as decision trees, rules [25], additive models [6] and attention-based networks [56] are utilized. Instead of supporting models that are functionally black-boxes, such as an arbitrary neural network or random forests with thousands of trees, these approaches use models in which there is the possibility of meaningfully inspecting model components directly, e.g. a path in a decision tree, a single rule, or the weight of a specific feature in a linear model.

When using deep neural networks or other complex models where we have access to the model parameters but direct introspection is not possible, approaches such as [22, 46] (explanations through influential training examples) and [29] (explanations through input attributions) are more useful. We however operate in the model agnostic regime where we do not assume any access to the ranking model's parameters. Recent work in this space has tended to focus on classification and regression models [42, 43] but not on ranking models. For other notions of interpretability and a more comprehensive description of the approaches we point the readers to [16].

In information retrieval there has been limited work on interpreting rankings. In [51], the authors estimate the feature importance of a document in a ranked list given a learned ranker. This becomes unwieldy if the number of features is large or un-interpretable itself. Singh and Anand [49] tried to approximate an already trained learning to rank model by a subset of (the original features) interpretable features using secondary training data from the output of the original model. Recently, [15] shows how a model introspective method meant for computer vision can be applied to interpreting the relevance score of a single query document pair. Firstly, these works do not focus on interpreting intents and secondly they are not model agnostic limiting their usability.

Closest to our work is [50, 53] which propose using LIME [42] directly for the task of explaining the relevance between a single query document pair. While the results of [50] are anecdotal, [53] evaluate how close complex ranking models are to the topic description as provided by TREC. Note that their objective is to evaluate how close the explanation terms are to the terms in the human created intent description for a query. This is different from our objective which is to accurately determine the intent terms that drive the ranking output of the model – not just a single document and not how close it is to the intended intent for a given query.

In terms of approach we utilize similar perturbation techniques as the aforementioned works but only to identify a set of candidates. Instead of training a local classification or regression model, we optimize for ranking preferences directly for better intent determination.

Tangential but related to our line of work is the explainability of recommender systems. We point readers to [58] for an overview of the field. A key difference to ranking models is that explanations here focus on explaining why a product was recommended to a user who has a rich interaction history. Recent works such as [4] detail how a transparent and scrutable model can be built directly instead of explaining it posthoc.

4 EXPLAINING RANKINGS

When explaining rankings, we wish to understand why for a query Q does a given ranker $\mathcal{BB}$ output the list $\mathcal{BB}(Q)$ as the top k documents from the set of documents retrieved from the index. The explanation is a simple human understandable representation that will help justify the behavior of $\mathcal{BB}$ for the given query Q to the user.

We consider a post-hoc model agnostic setting to understand ranking decisions, i.e. we do not assume access to the learning algorithm or model parameters (such as coefficients encoding feature weights or neural network parameters). Doing so allows us to be independent of the underlying ranking model. Our assumption is that we can approximate a complex ranker $\mathcal{BB}$ operating on an under-specified query Q with a simple ranker $\mathcal{E}$ with an over-specified query, i.e., with additional expansion terms $\mathbb{I}$. We consider the expansion terms as the intent representation as understood by the $\mathcal{E}$ and hence intent terms forms our explanation for the intent of $\mathcal{BB}$.

Terms (words or phrases) in adhoc retrieval are not only central to devising retrieval models but are also intuitive and understandable for humans.

Retrieval models or rankers that are explicit functions of well-understood IR features like term-level statistics, document lengths and proximity (like BM25, statistical language models [23, 39] are both interpretable and have been shown to be essential indicators of textual relevance. We chose [23] as our explanation model both because its easy to understand and because it naturally supports query expansions.

The PDR framework [33] outlines 3 main desiderata for evaluating interpretabilitily methods. The first is the predictive accuracy or *fidelity* of the explanation model. For an explanation to be convincing it should *faithfully* mimic the underlying model. Given a ranking $\mathcal{BB}(Q)$ to interpret or explain, the explanation model should produce a prediction (ranking in this case) that approximates $\mathcal{BB}(Q)$. We measure the fidelity of the mimiced ranking by the rank correlation measure Kendall's τ.

The second aspect to evaluating an explanation model is its *descriptive accuracy*. Here we are interested in measuring how accurately the explanation model has learned the same data relationships as the blackbox model. In essence, we evaluate if the explanation model is right for the right reasons. Note that descriptive accuracy can only be measured in cases where the data relationships learned by the blackbox model can easily be determined. In Section 6 we describe how we constructed blackbox models that explicitly expand the query under-the-hood. This allows us to directly quantify descritptive accuracy. We denote descriptive accuracy as *accuracy* henceforth for the sake of simplicity.

The final aspect when evaluating explanations is the relevance of the explanation to an audience for a given task. In Section 8 we briefly show use cases where our explanations can help legal and non-technical users quickly identify biases in the learned intents. Section 7 is dedicated to measuring the fidelity and accuracy of our approach on a variety of blackbox rankers in uncovering the intent $\mathbb{I}$.

4.1 Problem Statement

Formally, for a given keyword query Q and ranking $\mathcal{BB}(Q)$, produced by a complex black box ranker, $\mathcal{BB}$, we wish to identify a *high fidelity local explanation* $\mathbb{I}$. The fidelity of $\mathbb{I}$ is measured by the rank correlation metric τ between the rankings $\mathcal{BB}(Q)$ and $\mathcal{E}(Q \cup \mathbb{I})$.

5 APPROACH

To explain the top-k documents we should be in principle be able to explain all preference pairs $\boldsymbol{d}_{\pi(i)} \succ \boldsymbol{d}_{\pi(j)}$ where $i < j \leq k$. We denote a document at position i in a ranking π as $\boldsymbol{d}_{\pi(i)}$. In what follows we explain how to identify terms $\boldsymbol{w} \in \mathbb{I}$ and an easy-to-understand $\mathcal{E}$ such that an expanded query $Q \cup \mathbb{I}$ using $\mathcal{E}$ preserves most of the preference pairs in π.

Approach Overview. Our approach to explaining rankings produced by arbitrary blackbox ranking models consists of 3 major steps: (i) First we identify candidate terms from the documents retrieved (Section 5.1) (ii) Next we construct a *preference matrix* that is a data structure we use to compute the influence of each candidate term in preserving pairwise preferences according to a given $\mathcal{E}$ (Section 5.2). (iii) Finally for a given $\mathcal{E}$ we chose an $\mathbb{I}$ that maximizes fidelity (modelled as Maximum Preference Coverage) with the original ranking π using a greedy procedure described later in this section(described in Section 5.3). Figure 1 provides a visual overview of our approach.

5.1 Candidate Term Selection

The initial input to our approach is the query and resulting ranked list of documents. In this section we propose approaches to meaningfully chose a set of terms that should be considered as candidates for $\mathbb{I}$.

Candidate pre-selection. We pre-select an initial set of terms from the top-1000 documents of $\mathcal{BB}(Q)$ after removing common and generic terms using tf-idf filtering. In our experiments we chose 1000 terms. Not all chosen terms will contribute to the explanation and many of them would be false positives.

Document Perturbation. We assume that for a term to be an intent term its presence or absence in a document influence the relevance of the document. Exploiting this hypothesis, we systematically perturb documents by adding and removing terms from the pre-selected terms. Similar notions of perturbation have been used for text classification, question answering and machine translation [2, 20, 42]. Adhoc retrieval models are often sensitive to document lengths. Consequently, in our document perturbation procedure that we ensure that document lengths are maintained.

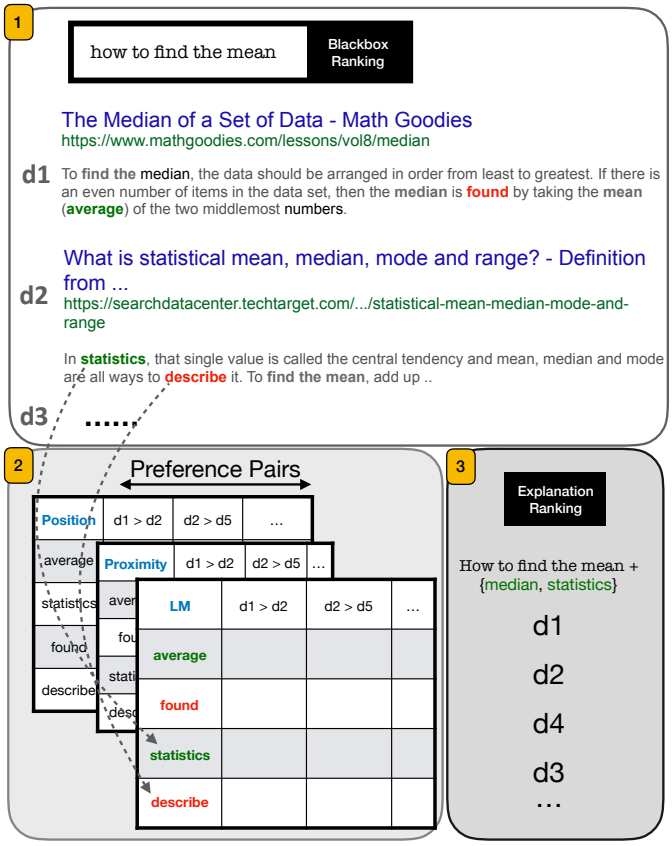

Figure 1: Approach Overview: (1) Extract candidate terms and preference pairs from the ranking produced by the blackbox (2) Construct preference matrix using candidate terms and pre-selected interpretable rankers $\mathcal{E}$ (3) Choose the set of terms $\mathbb{I}$ and $\mathcal{E}$ that maximizes coverage of preference pairs in the target ranking. The explanation model E is a combination of $\mathbb{I}$ and $\mathcal{E}$ that tries to mimic the rank order produced by the blackbox $\mathcal{BB}$ Q .

We estimate the importance of a pre-selected term by substituting all instances of the term in a result document by an out-of-vocabulary(OOV) term. We call this *perturbation by removal*. Similarly, in *perturbation by term addition*, we add terms (absent from the document) from the vocabulary or pre-selected terms. Additive perturbation allows us to control not only the frequency of the added candidate term but also the position and order. The relevance contribution of a term on the document is estimated as the score difference (computed using the original ranking model) with and without the perturbation. We retain the top terms (250 in our experiments that ensure a substantial recall) from the pre-selected set that have the maximum relevance contribution over the top-k in the result set.

In our experiments we use a combination of the two to efficiently select a set of candidates with minimal potential false positives. We call this refined subset of terms from the pre-selected terms as the *candidate set.*

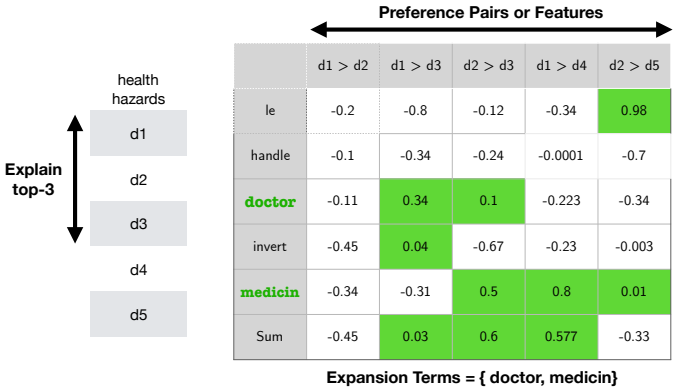

Figure 2: preference Matrix for the ranking for the query "health hazards"

.

5.2 Modeling Preference Pairs

Until now we were concerned with the terms that are relevant to the query. However, not all terms are responsible for preference pairs induced by the original ranking. In what follows, we describe a procedure on how to select terms from the candidate set that preserve the preference pair ordering in $\mathcal{BB}$ (Q).

In principle, for a ranking of size k we have $k.(k-1)/2$ concordant pairs. However, only considering the preference pairs from the top-k documents could result in higher likelihood of false positive terms being included in the selection set. *False positive terms* tend to be general terms that co-occur with relevant terms resulting in an increased local fidelity (local here refers to the top ranked documents). We counteract the effect of false positives by sampling additional preference pairs from outside the top-k and requiring that the algorithm covers a larger number of preference pairs. Carefully sampling pairs can help focus on alternate yet important query aspects and prevent overfitting. Conversely, improper sampling or sampling too many pairs from the tail could lead to good global rank fidelity but poor local rank fidelity.

Sampling Preference Pairs/Features. We use multiple strategies to sample preference pairs.

(1) RANDOM : randomly select preference pairs from the target ranking π.
(2) RANK BIASED : sample preference pairs from π that are weighted by rank. Each pair $(\boldsymbol{d}_{\pi(i)} \succ \boldsymbol{d}_{\pi(j)})$ is weighted by $1/rank(d_i)+1/rank(d_j)$.
(3) TOP-K + RANK RANDOM : construct preference pairs based on a combination of rank and random sampling. In this method, for a preference pair $(\boldsymbol{d}_{\pi(i)} \succ \boldsymbol{d}_{\pi(j)})$, d_i is rank bias sampled but d_j is randomly sampled.
(4) TOP-K + RANDOM : consider all pairs from the top-k results to explain and a fixed number of randomly sampled pairs.

We contrast these against TOP-K that are all preference pairs in top-k results in the experiments.

Constructing the Preference Matrix The next step in our approach is to construct an $n \times m$ preference matrix for a given candidate $\mathcal{E}$, where n is the number of candidate terms and m is the number of preference pairs. For each preference pair / feature, we compute a score that encodes the degree of concordance the candidate term maintains for the pair. We employ $\mathcal{E}$ to first estimate the importance of a term for each document. The score for the term $\boldsymbol{w}$ given a pair $\boldsymbol{d}_{\pi(i)} \succ \boldsymbol{d}_{\pi(j)}$ is computed as $\mathcal{S}_{\mathcal{E}}(\boldsymbol{w}, \boldsymbol{d}_{\pi(i)})$ - $\mathcal{S}_{\mathcal{E}}(\boldsymbol{w}, \boldsymbol{d}_{\pi(j)})$.

$\mathcal{S}_{\mathcal{E}}(,)$ is the relevance score estimated by $\mathcal{E}$ as $\mathcal{S}_{BB}(,)$ is to $\mathcal{BB}$. The score for each cell is computed by then multiplying it with a weight corresponding to each preference pair. We make the assumption that more distant concordant pairs (rank 1 vs rank 10) as opposed to close ranked document pairs (rank 11 vs rank 12) contains stronger evidence of relevant intent terms. We weight a given preference pair $\boldsymbol{d}_{\pi(i)} \succ \boldsymbol{d}_{\pi(j)}$ by $w_{ij} = 1 + ln(j - i)$.

In the next section we show how this choice allows us to directly optimize for fidelity if we select an $\mathcal{E}$ akin to a language model where $q_i \in q$ are terms in a query.

$$\mathcal{S}_{\mathcal{E}}(q, d) = P(d|q) = \prod_{q_i \in q} P(q_i|d) = \sum_{q_i \in q} logP(q_i|d) \quad (1)$$

5.3 Optimizing Preference Pair Coverage

Once we construct a good set of candidates we then build the preference matrix corresponding to a set of pre-selected $\mathcal{E}$. In this section we describe how to find the set of terms $\mathbb{I}$ given a single preference matrix.

We start with a set of candidate expansion terms $\mathcal{X}$ ($|\mathcal{X}| = n$), where each expansion term $\boldsymbol{t} \in \mathcal{X}$ is described by a feature vector; thus, $\boldsymbol{t}$ has a vector $(p_1, \ldots, p_d) \in \mathbb{R}^d$, and feature vectors in $\mathcal{X} \subseteq \mathbb{R}^d$. Each feature corresponds to a preference pair $\boldsymbol{d}_{\pi(i)} \succ \boldsymbol{d}_{\pi(j)}$ and its value determines to what degree is the preference pair satisfied if $\boldsymbol{t}$ is chosen (described earlier as $\mathcal{S}_{\mathcal{E}}(\boldsymbol{w}, \boldsymbol{d}_{\pi(i)})$ - $\mathcal{S}_{\mathcal{E}}(\boldsymbol{w}, \boldsymbol{d}_{\pi(j)})$). We build the preference matrix P from the term vectors and intend to find a minimal set of terms $\mathbb{I} \subseteq \mathcal{X}$ as expansion terms.

Preference Coverage. Given a selection set represented as a Boolean vector $\boldsymbol{s}$, the *preference coverage* PCov over the aggregate vector $\boldsymbol{y} = \boldsymbol{s}^T\mathcal{X}$ is given by $\text{PCov}(\boldsymbol{s}) = \sum_i [\![y_i > 0]\!]$.

The best selection of expansion terms naturally is the set that maximizes the preference coverage or explanation fidelity. We pose the MAXIMUM PREFERENCE COVERAGE problem as choice of a set of terms where maximum preferences are covered. Writing it as an Integer Linear Program we have:

$$\max \sum_{0 \le j < m} [\![y_j > 0]\!] \quad (2)$$

$$s.t. \quad (3)$$

$$s_i \in \{0, 1\}, 0 \le i < n \quad (4)$$

$$y_j = \sum_{0 \le i \le n} s_i.P_{i,j}.w_{i,j} \quad (5)$$

Note that $P_{i,j} = \mathcal{S}_{\mathcal{E}}(\boldsymbol{w}, \boldsymbol{d}_{\pi(i)})$ - $\mathcal{S}_{\mathcal{E}}(\boldsymbol{w}, \boldsymbol{d}_{\pi(j)})$

The MAXIMUM PREFERENCE COVERAGE problem is NP hard. We do not include the proof in the paper for space reasons but we note that the proof sketch follows from the fact that the MAXIMUM PREFERENCE COVERAGE is a generalization of the well known SET COVER problem. Not only is the MAXIMUM PREFERENCE COVERAGE problem NP hard, it is also easy to see that it is not sub-modular.

The Maximum Preference Coverage with concordant document pairs as features naturally tries to maximize fidelity as defined by Kendall's τ. Now selecting terms that maximize coverage of concordant pairs is equivalent to selecting terms that when used as expansions closely reproduces $\mathcal{BB}$ (q). A positive score for a feature indicates that a term is more relevant for $\boldsymbol{d}_{\pi(i)}$ than $\boldsymbol{d}_{\pi(j)}$ according to $\mathcal{E}$ where $\boldsymbol{d}_{\pi(i)} \succ \boldsymbol{d}_{\pi(j)}$. For a set of terms, simply adding the scores for that feature will indicate if concordance is maintained and in turn maximize fidelity.

Greedy Algorithm. Although Maximum Preference Coverage does not induce submodularity, we propose a heuristic greedy algorithm that intends to maximize the preference coverage of the input.

At each iteration the algorithm greedily chooses the term into the selection set that provides the maximum utility. The utility of a term $\boldsymbol{t}$ at any stage of the algorithm $U(\boldsymbol{t}, T)$ is the increase in the preference coverage when $\boldsymbol{t}$ is added to the selection set T or $U(\boldsymbol{t}, T) = \text{PCov}(T \cup \boldsymbol{t}) - \text{PCov}(T)$.

This is clearer from the example in Figure 2. Say the term `medicine` is chosen into the selection set with a PCov({`medicine`}) = 3. Now choosing the term `handle` in fact reduces PCov({ `medicine`, `handle` }) = 2. In case of ties, the $\boldsymbol{t}$ that has the highest column sum (denoted by *Psum*) of the covered features are considered ,i.e., $Psum(\boldsymbol{t}) = \sum_{t_i>0 \& t_i \in \boldsymbol{t}} t_i$.

6 EXPERIMENTAL SETUP

The major challenge in evaluation is measuring accuracy (descriptive accuracy according to PDR [33]). In order to do so, we must obtain a ground truth of *perceived intents of black-box rankers* (note that this is different from the actual user or query intent) which is difficult for complex neural models. In order to quantify the quality of our explanations we resort to black-boxes whose intents are fully understood. This in fact is common practice in evaluation of local posthoc-interpretable approaches [21, 41] with the underlying assumption that if an explanation model can correctly locally interpret a simple well understood model then it can faithfully (locally) interpret other complex models. For the scenarios where the rankers intents are unknown we resort to measuring the explanation performance using fidelity.

Dataset and Queries. For all experiments we use the web track Clueweb09 TREC test collection (category B) for adhoc retrieval. We use the first 200 queries only. Note that in our experiments we are more interested in showing that our approach can be applied to a variety of retrieval models (trained on a large training set) rather than the same retrieval model across multiple test collections. For that reason we consider only one Web scale collection with a large set of queries rather than more datasets.

Metrics. We use Kendall's τ to measure **fidelity** between the explanation ranking and the original ranking (i.e. the predictive accuracy) in order to establish our approach's ability to extract intents $\mathbb{I}$ that produce a similar output as $\mathcal{BB}$ (q). The (descriptive) **accuracy** of the explanation is computed as the fraction of intent terms that overlap with the set of ground truth intents (whenever available). We select the top-10 documents to explain for a given q and $\mathcal{BB}$ and set max $|\mathbb{I}|$ to 10 in all our experiments.

Blackbox Ranking Models. We consider three black box rankers RM3 [23], EMB [13] and DESM [34] where we know the actual intent terms due to the explicit modelling of relevance computation for the expansion terms. These are the ground truth terms used in computing the metrics. We set the number of ground truth terms to 10.

- **RM3.** Using the RM3 [23] algorithm, we determined a set of relevant expansion terms from the top-k documents for a given query. We then re-ranked the results using the aforementioned language model. We use RM3-k where $k = \{10, 20\}$.
- **EMB.** Instead of using pseudo-relevant documents to find expansion terms, an external collection is used for the expansions. We use glove embeddings (300 dimensions) trained on English Wikipedia dump(2016) to find semantically related terms. We first average the embeddings of the words in the query to create a query vector. Next we search the embedding space for the 10 nearest terms that also occur in the top 10 documents.
- **DESM.** [34] models the relevance score of d given q as a parameterized sum between the syntactic relevance and semantic similarity, P_{sem}, between a learned query vector representation and the document vector representation. We select terms closest to the query vector that are also present in the top 50 documents of the initial result list as expansions. They are used to compute the syntactic relevance whereas all terms are used for the semantic similarity. The expansion terms are considered as ground truth terms but this set is not exact. The two relevance components are combined linearly with a parameter α set to 0.7. Here the objective is to uncover the term expansions in-spite of the noise. To compute the vectors we employ the same glove embeddings from EMB.

In Section 8, we consider three neural retrieval models that have been shown to have improvements over BM25: DRMM [17], P-DRMM [30] and M-Pyramid [38]. DRMM operates on query-document term similarity histograms whereas M-Pyramid uses a query-document term similarity matrix as input and then learns hierarchical representations. P-DRMM is based on the PACRR model that operates on query-document interaction matrices along with considering the position of terms in a document. All models were trained on an adhoc search test collection (Robust04). Here we focus on measuring fidelity and use cases where our intent based explanations are useful.

Explanation Model Settings: While accuracy is measured against a ground truth set of terms, fidelity however is sensitive to the choice of $\mathcal{E}$ used to estimate $\mathcal{BB}$. For the explanation model we fix $\mathcal{E}$ as a language model estimated using MLE with additive smoothing, i.e., $P(q_i|d) = \frac{count(q_i,d)}{|d|}$. We had to be careful in choosing a language model that is sufficiently different from the ranking function in $\mathcal{R}$ since we are only concerned with bag of words based models.

7 RESULTS

In order to establish the effectiveness of our approach, we (i) evaluate the quality of our explanation approach in terms of fidelity

and accuracy, (ii) performance under different sampling and feature generation strategies, and finally (iii) present scenarios where explanations can be used to debug rankers.

7.1 Effectiveness of Explanations

Table 1 reports the accuracy and fidelity values of all the models under consideration for the best individual setting of our explanation method. Local fidelity measures the rank correlation between our explanation ranking and original ranking for only the top-10 results while global considers all retrieved documents, i.e., 1000 in our experiments.

Accuracy vs Fidelity. We find that for these simpler models, we're able to achieve an accuracy greater than 85%. In figure 3, we see the correlation between accuracy and fidelity in a scatter plot. We uniformly sampled 600 query explanations (in total) from our devised models to create this plot. We find that global fidelity is strongly correlated with accuracy whereas local fidelity shows relatively weak correlation. This result is useful in estimating how well we do on neural models that do not have a ground truth. Note that a $\tau > 0.5$ indicates that we can faithfully reproduce over 75% of concordant pairs with our simple interpretable model (ignoring score ties).

How many pairs to sample ? Creating preference matrices with many pairs can be computationally expensive however we observe diminishing returns after sample 500 pairs. For all our presented results we set the number of pairs to 500. For the neural models, we found that increasing the number of pairs improves fidelity. We set the number of pairs to 2500 for the neural models.

Which sampling strategy works best ? In Table 2, we see that sampling pairs at random is the worst option. TOP-K + RANDOM and TOP-K + RANK RANDOM substantially improve accuracy (and in turn global fidelity). Both of these techniques allow for documents towards the bottom of the ranking to be selected. The benefit of doing so is observed when comparing against RANK BIASED and TOP-K . Sampling pairs only from the top-k leads to highest overall local fidelity as expected but poor accuracy. This indicates that sampling pairs from the top and bottom of the ranked list gives us the best "regularization" effect.

7.2 Effect of Perturbation

False positive terms have a large impact on the quality of our approach that we try to address by exploiting the optimizing preference pair coverage. For the non-DESM rankers chosen for our experiments, we can remove all false positive terms this way since score changes will only occur due to the terms in the ground truth (Table 1). However for DESM, false positives are harder to determine due to soft matching between query and document terms. This according to us is also hard to evaluate because of lack of access to all terms that are indicators of relevance. Hence the accuracy score reported for DESM is a lower bound on our actual performance. Perturbation helps reduce the number of false positives with no effect on recall in DESM. Nonetheless, our approach can still locally explain DESM with an accuracy of 0.55 when using TOP-K + RANK RANDOM sampling which is similar to RM3-10 (0.57).

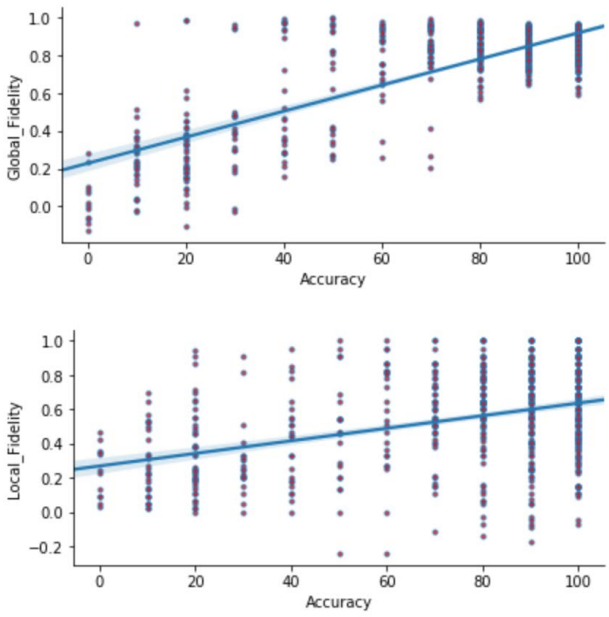

Figure 3: Scatter plot of Accuracy (%) vs Fidelity (Kendall's τ). Global Fidelity is highly correlated with Accuracy. Local Fidelity is less correlated. Global Fidelity is a better indicator of accuracy. Data points represent queries sampled from all models where ground truth is available.

Model	Accuracy	local fidelity	global fidelity
RM3-10	0.87	0.577	0.793
RM3-20	0.86	0.568	0.866
EMB	0.86	0.630	0.926
DESM	0.55	0.189	0.368
DRMM	-	0.449	0.491
P-DRMM	-	0.537	0.398
M-PYRAMID	-	0.401	0.445

Table 1: RM3-10, RM3-20, EMB and DESM (500 features & TOP-K + RANK RANDOM). DRMM, P-DRMM and M-PYRAMID (2500 features & TOP-K + RANK RANDOM). Note that $\tau \in [-1, +1]$. A negative τ implies there are more discordant pairs than concordant. Usually $\tau > 0.5$ is considered to be strong correlation.

8 EXPLAINING NEURAL MODELS

The goal of this case study is to demonstrate the utility of our explanation framework when interpreting complex models like DRMM, Match Pyramid and DESM (described in Section 6) where ground truth is absent in a posthoc model agnostic setting. Here we use the full fledged approach with multiple candidate $\mathcal{E}$. We chose 3 simple rankers based on well studied features that are known to improve adhoc text retrieval: (i) term frequency and document length (language model), (ii) position and (iii) proximity of terms.

Sampling	Accuracy	Local fidelity	Global fidelity
RANDOM	0.327	0.294	0.255
TOP-K	0.406	**0.407**	0.273
+ Random	0.535	0.210	0.363
+ Rank Random	**0.553**	0.189	**0.368**
RANK BIASED	0.456	0.216	0.329

Table 2: Interpreting DESM (500 features).

8.1 Comparing against TREC intents

In the previous sections we evaluated the ability of our approach in uncovering the blackbox model's intent which is the true test of explanation accuracy. Now we turn towards evaluating which neural model's inferred intent is most human like by comparing against hand crafted query intents.

As described in Section 2, TREC assessors estimate relevance given a clearly described intent. Table 3 shows a selection of queries and their corresponding TREC descriptions. We see that queries tend to be under-specified but with the aid of the description, the context becomes clearer. We generated explanations for DRMM and Match Pyramid and compared them against the TREC descriptions to identify if the rankers do indeed capture the human defined intent provided by TREC for the given query. We found that both rankers tend to capture the intended intent accurately, albeit with slightly differing terms.

For the query *churchill downs*, DRMM is able to better capture the schedule part of the intent as can be seen by the terms `year`,`date` and `hour`. Similarly, DRMM is also able to capture more of the intent for the query *bph treatment*. Note that we are able explain over 75% of the concordant pairs in both cases (global fidelity of approximately 0.5 and 0.7 for both models respectively) indicating that the explanations are more likely to be accurate.

8.2 Detecting Bias through Intent Terms

Training Data Bias. Table 4 highlights queries that illustrate the power of our explanations in this scenario. DESM uses wikipedia embeddings which is reflected in the more generic intent explanation terms (nurses as opposed to war for *alexian brothers hospital*). Since DRMM was trained on Robust04, which is a news collection from 2004, we find more terms related to news-worthy events. This is particularly evident for *afghan flag*. USA was involved in many conflicts in Afghanistan in the early 2000s and is promptly the reason why documents related to the USA get ranked higher for DRMM. DESM on the other hand favors documents more directly related to the concept of a flag.

We also find evidence for temporal bias in the queries *fidel castro* and *electoral college 2008 results*. DRMM ranks documents related to events in 2004 higher. Vice President and brother of Fidel, Raul Castro was handed control in 2006 (evidenced in DESM) due to Fidel Castro's illness which was a more prominent topic in 2004. Similarly DRMM considers documents related to Al Gore more relevant as compared to DESM for *electoral college 2008 results*.

Model Bias. The explanation also gives us insight into the nature of the ranker. For the query *how to find mean*, even though the semantics of the query terms is resilient to temporal shifts, DRMM's gating mechanism helps capture the right semantics of the query. DESM on the other hand computes semantic similarity in a more simplistic manner, relying heavily on the pertained word embeddings to capture the right semantics.

8.3 Explaining Rankings for DRMM

Can we mimic DRMM's ranking ? We generated the explanations for 50 randomly selected queries from the dataset we used in the quantitative experiments for DRMM. We found that our approach could effectively explain nearly half of all preference pairs and could also explain nearly the same fraction in the top 10 (global fidelity = 0.48 and local fidelity = 0.47). Based on the quantitative results where we showed that global fidelity is correlated with accuracy, we can be confident that the explanations produced are accurate.

Why is d relevant to q ? Once we have identified the intent terms, we can reason why a given document is relevant to the query. The simplest way to visualize this explanation is to highlight terms in the document. For the query *fidel castro* and document $clueweb09-en0005-70-11327$, according to our explanation for DRMM the terms *cuba* and *intestine* are indicators of relevance whereas the same document for DESM is also ranked in a similar position but for different reasons. Here `raul` and `communist` are the reasons why $clueweb09-en0005-70-11327$ is considered relevant. This difference can be visualized using snippets as shown in Figure 4 to help users easily discern why a document is relevant to a query.

Why is d_i more relevant than d_j ? The intent explanation terms when used with $\mathcal{E}$ can further help us understand why a document was considered more relevant than another. $\mathbb{I}$ can effectively explain document pairs that are concordant in both target and explanation ranking. Figure 5 shows an anecdotal document pair explanation for the query *fidel castro* and DRMM. Due to our choice of an $\mathcal{E}$ that scores terms independently we can construct an easy-to-understand visual explanation that is a composition of term scores. In our experiments we found that our approach chose 10 terms along with the language model (TF and document length based) $\mathcal{E}$ to best explain the ranking produced by DRMM (local fidelity = 0.51 and global fidelity = 0.57). In this ranking d_2 and d_5 seem to be similar when just looking at the explanation terms – {`havana, domestic, cuba, invest, intestine, medical, real`} (both are related to medical issues). However on closer inspection, using $\mathcal{E}$, it becomes clear that `intestine` is a key term that is more prominent in d_2 than in d_5. Similarly, d_{10} is ranked considerably lower since it only matches a few intent terms.

9 APPLICATION TO OTHER DOMAINS

In our experiments, we demonstrated the applicability of our approach to models trained on a document ranking task where queries tend to be under-specified and documents tend to be at least a few paragraphs.

Query	TREC Topic Description	DRMM Intent Explanation	M-Pyrmaid Intent Explanation
eggs shelf life	*What is the shelf life of a chicken egg —that is, how old can it be and still be safe to eat?*	store chicken month year actual eat fat fish tend	product develop air pick chemical old food remove
bph treatment	*What are the treatment options for BPH (benign prostatic hyperplasia)? Which medications are used for BPH? What are the symptoms of BPH? What are the side effects of BPH treatments?*	prostatic symptom grow inflammation bladder red medical urologist	prostatic hyperplasia urine bladder medicine drug hormone psa
churchill downs	*Find information on the (horse) racing schedule at Churchill Downs.*	horse winner track begin hour year date patron registration radio run	horse oak park win cup fair rider anderson jones reed
bart sf	*Find information about the BART (Bay Area Rapid Transit) system in San Francisco.*	transit service metro rider ac oakland san	transit station train downtown union castro san

Table 3: Comparing intent descriptions from TREC with explanations produced by our approach for DRMM and Match Pyramid.

Query	DRMM Intent Explanation	DESM Intent Explanation
how to find the mean ?	x statistics plus know	actually say want meant
alexian brothers hospital	war person patient course	nurses father physical doctors
afghanistan flag	US official inscription time	symbol nation flagpole hoist
fidel castro	havana domestic cuba invest	cuba havana dictator communist
electoral college 2008 results	president popular statistic senate nominee gore	2009 2004 following election outcome expected

Table 4: Anecdotal explanations. The Intent Explanation terms ($\mathbb{I}$) are terms that optimize for Maximum Preference Coverage according to our approach.

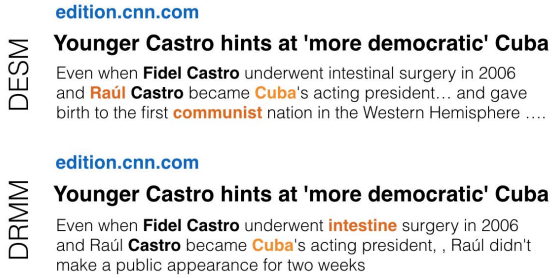

Figure 4: Explanation for why *clueweb09−en0005−70−11327* is relevant for *fidel castro* according to DRMM and DESM. Snippets containing the most important terms that were found as intent terms using our approach are highlighted.

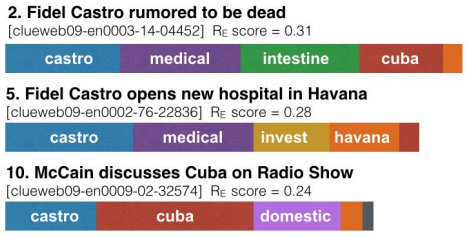

Figure 5: Explanation for $d_2 > d_5 > d_{10}$ for DRMM for the query *fidel castro*. The length of a cell in the bar indicates term importance to d as estimated by our explanation model. Since $\mathcal{E}$ is known, we can also explain how term importance is computed.

QA Models Rankers are also used in question answering (QA) tasks where the query is a question and the answers can be an arbitrary piece of text. A typical QA system first retrieves a set of candidate answers and then ranks them before displaying/generating the final answer. We successfully applied our approach to a QA model similar to [36] (a BERT [12] based method) to explain anecdotal questions from the MSMarco dataset [35]. Table 5 shows explanations for 5 randomly selected questions from the dev set. Here we can see how the BERT based model perceives the question when ranking candidate answers (passages in this dataset) which can further shed light on answers delivered to the end user.

Recommender Systems A subtle point to note in our work is that the approach itself does not require a query as input. Given

Question	BERT Based Model Intent Explanation
Ludacris Net Worth	`million actor rapper atlanta`
Explain what a bone scan is and what it is used for	`body abnormalities joint radionuclide`
Does Suddenlink Carry ESPN3	`espn2 computer service games`
Androgen receptor define	`estrogen assay positive therapy`
3 levels of government in canada and their responsibilities	`commons queen laws federal`

Table 5: Anecdotal explanations for a BERT based QA model.

any ranking produced by a black box model, we can select a set of terms (from the mined candidates) and a simpler ranking model that serve as the explanation. This implies that our approach can also be applied to ranking models used in recommender systems where the query is not a set of terms input by the user but the user itself, the date or an item being viewed. The key is the ability to mine candidates via document perturbation. For instance, consider the ranking of similar products in a shopping application. Here the query can be considered to be the product currently being viewed. The ranked list of similar items can be explained with our approach by: (i) mining candidate terms from the product descriptions and titles (ii) constructing the preference matrix based on concordant product pairs (iii) selecting terms using the greedy approach suggested earlier. Now the intent based explanation can inform the user as to why certain products are recommended first for the product she is currently viewing.

10 CONCLUSION

In this paper we detailed our framework for post-hoc explanations of adhoc black box rankers. Our setting enables us to tackle a multitude of text based retrieval models (predominantly used in news search, medical search, patent retrieval, product search, etc.) irrespective of the underlying learning algorithm or training data. Note that web search tends to rely heavily on network and behavioral signals in most cases but textual relevance is crucial for tail queries (rarely seen or unseen queries). From the quantitative results we gathered that using preference pairs only from the top-k results leads to high local fidelity but low accuracy. Sampling additional pairs from lower in the ranked list on the other hand can substantially increase accuracy since it indirectly optimizes for global fidelity at the cost of local fidelity. We find that TOP-K + RANK RANDOM is usually the best sampling method which shows that taking pairs of documents that are mostly from the top with with finer differences (RANK BIASED) or just randomly selected (RANDOM) is not the best strategy. Qualitatively we have seen how our approach can be used to identify temporal and model biases. We also how we could use intent terms to explain pairwise rank differences. In the future we would try to incorporate evidence from model-introspective approaches to improve our explanation fidelity.

REFERENCES

[1] Nir Ailon, Moses Charikar, and Alantha Newman. 2008. Aggregating inconsistent information: ranking and clustering. *Journal of the ACM (JACM)* 55, 5 (2008), 23.

[2] David Alvarez-Melis and Tommi S Jaakkola. 2017. A causal framework for explaining the predictions of black-box sequence-to-sequence models. *arXiv preprint arXiv:1707.01943* (2017).

[3] Leif Azzopardi, Wim Vanderbauwhede, and Hideo Joho. 2010. Search system requirements of patent analysts. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*. ACM, 775–776.

[4] Krisztian Balog, Filip Radlinski, and Shushan Arakelyan. 2019. Transparent, Scrutable and Explainable User Models for Personalized Recommendation. In *Proceedings of the 42Nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'19)*. ACM, New York, NY, USA, 265–274. https://doi.org/10.1145/3331184.3331211

[5] Alexander Binder, Grégoire Montavon, Sebastian Lapuschkin, Klaus-Robert Müller, and Wojciech Samek. 2016. Layer-wise relevance propagation for neural networks with local renormalization layers. In *International Conference on Artificial Neural Networks*. Springer, 63–71.

[6] Rich Caruana, Yin Lou, Johannes Gehrke, Paul Koch, Marc Sturm, and Noemie Elhadad. 2015. Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 1721–1730.

[7] Shuo Chang, F. Maxwell Harper, and Loren Gilbert Terveen. 2016. Crowd-Based Personalized Natural Language Explanations for Recommendations. In *Proceedings of the 10th ACM Conference on Recommender Systems (RecSys '16)*. ACM, New York, NY, USA, 175–182. https://doi.org/10.1145/2959100.2959153

[8] Paul Alexandru Chirita, Rita Gavriloaie, Stefania Ghita, Wolfgang Nejdl, and Raluca Paiu. 2005. Activity based metadata for semantic desktop search. In *European Semantic Web Conference*. Springer, 439–454.

[9] W Bruce Croft and John Lafferty. 2013. *Language modeling for information retrieval*. Vol. 13. Springer Science & Business Media.

[10] Piotr Dabkowski and Yarin Gal. 2017. Real Time Image Saliency for Black Box Classifiers. *arXiv preprint arXiv:1705.07857* (2017).

[11] Mostafa Dehghani, Hamed Zamani, Aliaksei Severyn, Jaap Kamps, and W Bruce Croft. 2017. Neural Ranking Models with Weak Supervision. *arXiv preprint arXiv:1704.08803* (2017).

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[13] Fernando Diaz, Bhaskar Mitra, and Nick Craswell. 2016. Query Expansion with Locally-Trained Word Embeddings. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 367–377.

[14] Finale Doshi-Velez and Been Kim. 2017. Towards a rigorous science of interpretable machine learning. (2017).

[15] Zeon Trevor Fernando, Jaspreet Singh, and Avishek Anand. 2019. A Study on the Interpretability of Neural Retrieval Models Using DeepSHAP. In *Proceedings of the 42Nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'19)*. ACM, New York, NY, USA, 1005–1008. https://doi.org/10.1145/3331184.3331312

[16] Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. 2018. A survey of methods for explaining black box models. *ACM computing surveys (CSUR)* 51, 5 (2018), 93.

[17] Jiafeng Guo, Yixing Fan, Qingyao Ai, and W Bruce Croft. 2016. A deep relevance matching model for ad-hoc retrieval. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*. ACM, 55–64.

[18] David Hawking. 2004. Challenges in enterprise search. In *Proceedings of the 15th Australasian database conference-Volume 27*. Australian Computer Society, Inc., 15–24.

[19] Kai Hui, Andrew Yates, Klaus Berberich, and Gerard de Melo. 2017. A Position-Aware Deep Model for Relevance Matching in Information Retrieval. *arXiv preprint arXiv:1704.03940* (2017).

[20] Daniel Khashabi, Tushar Khot, Ashish Sabharwal, and Dan Roth. 2017. Learning what is essential in questions. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*. 80–89.

[21] Been Kim, Rajiv Khanna, and Oluwasanmi O Koyejo. 2016. Examples are not enough, learn to criticize! criticism for interpretability. In *Advances in Neural Information Processing Systems*. 2280–2288.

[22] Pang Wei Koh and Percy Liang. 2017. Understanding black-box predictions via influence functions. *arXiv preprint arXiv:1703.04730* (2017).

[23] Victor Lavrenko and W. Bruce Croft. 2001. Relevance Based Language Models. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '01)*. ACM, New York, NY, USA, 120–127. https://doi.org/10.1145/383952.383972

[24] Tao Lei, Regina Barzilay, and Tommi Jaakkola. 2016. Rationalizing neural predictions. *arXiv preprint arXiv:1606.04155* (2016).

[25] Benjamin Letham, Cynthia Rudin, Tyler H McCormick, David Madigan, et al. 2015. Interpretable classifiers using rules and Bayesian analysis: Building a better stroke prediction model. *The Annals of Applied Statistics* 9, 3 (2015), 1350–1371.

[26] Canjia Li, Yingfei Sun, Ben He, Le Wang, Kai Hui, Andrew Yates, Le Sun, and Jungang Xu. 2018. NPRF: A Neural Pseudo Relevance Feedback Framework for Ad-hoc Information Retrieval. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*.

[27] Jiwei Li, Xinlei Chen, Eduard Hovy, and Dan Jurafsky. 2015. Visualizing and understanding neural models in nlp. *arXiv preprint arXiv:1506.01066* (2015).

[28] Tie-Yan Liu et al. 2009. Learning to rank for information retrieval. *Foundations and Trends® in Information Retrieval* 3, 3 (2009), 225–331.

[29] Scott M Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems*. 4765–4774.

[30] Ryan McDonald, George Brokos, and Ion Androutsopoulos. 2018. Deep Relevance Ranking Using Enhanced Document-Query Interactions. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. ACL, 1849–1860. http://aclweb.org/anthology/D18-1211

[31] Bhaskar Mitra and Nick Craswell. 2017. Neural Models for Information Retrieval. *arXiv preprint arXiv:1705.01509* (2017).

[32] Mandar Mitra, Amit Singhal, and Chris Buckley. 1998. Improving automatic query expansion. In *SIGIR*, Vol. 98. 206–214.

[33] W James Murdoch, Chandan Singh, Karl Kumbier, Reza Abbasi-Asl, and Bin Yu. 2019. Interpretable machine learning: definitions, methods, and applications. *arXiv preprint arXiv:1901.04592* (2019).

[34] Eric Nalisnick, Bhaskar Mitra, Nick Craswell, and Rich Caruana. 2016. Improving Document Ranking with Dual Word Embeddings. In *Proceedings of the 25th International Conference Companion on World Wide Web (WWW '16 Companion)*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland, 83–84. https://doi.org/10.1145/2872518.2889361

[35] 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. (2016).

[36] Rodrigo Nogueira and Kyunghyun Cho. 2017. Task-oriented query reformulation with reinforcement learning. *arXiv preprint arXiv:1704.04572* (2017).

[37] Hamid Palangi, Li Deng, Yelong Shen, Jianfeng Gao, Xiaodong He, Jianshu Chen, Xinying Song, and R Ward. 2014. Semantic modelling with long-short-term memory for information retrieval. *arXiv preprint arXiv:1412.6629* (2014).

[38] Liang Pang, Yanyan Lan, Jiafeng Guo, Jun Xu, and Xueqi Cheng. 2016. A Study of MatchPyramid Models on Ad-hoc Retrieval. *SIGIR workshop on Neural Information Retrieval (NeuIR-16)* arXiv:1606.04648 (2016). arXiv:1606.04648 http://arxiv.org/abs/1606.04648

[39] Jay M Ponte and W Bruce Croft. 1998. A language modeling approach to information retrieval. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 275–281.

[40] Yonggang Qiu and Hans-Peter Frei. 1993. Concept based query expansion. In *Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 160–169.

[41] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. Model-agnostic interpretability of machine learning. *arXiv preprint arXiv:1606.05386* (2016).

[42] 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*. ACM, 1135–1144.

[43] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2018. Anchors: High-precision model-agnostic explanations. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

[44] Stephen E Robertson. 1990. On term selection for query expansion. *Journal of documentation* 46, 4 (1990), 359–364.

[45] Joseph Rocchio. 1971. Relevance feedback in information retrieval. *The Smart retrieval system-experiments in automatic document processing* (1971), 313–323.

[46] Boris Sharchilev, Yury Ustinovskiy, Pavel Serdyukov, and Maarten Rijke. 2018. Finding Influential Training Samples for Gradient Boosted Decision Trees. In *International Conference on Machine Learning*. 4584–4592.

[47] Yelong Shen, Xiaodong He, Jianfeng Gao, Li Deng, and Grégoire Mesnil. 2014. Learning Semantic Representations Using Convolutional Neural Networks for Web Search. In *Proceedings of the 23rd International Conference on World Wide Web (WWW '14 Companion)*. ACM, New York, NY, USA, 373–374. https://doi.org/10.1145/2567948.2577348

[48] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. 2013. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034* (2013).

[49] Jaspreet Singh and Avishek Anand. 2018. Posthoc Interpretability of Learning to Rank Models using Secondary Training Data. *arXiv preprint arXiv:1806.11330* (2018).

[50] Jaspreet Singh and Avishek Anand. 2019. EXS: Explainable Search Using Local Model Agnostic Interpretability. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining (WSDM '19)*. ACM, New York, NY, USA, 770–773. https://doi.org/10.1145/3289600.3290620

[51] Maartje ter Hoeve, Anne Schuth, Daan Odijk, and Maarten de Rijke. 2018. Faithfully Explaining Rankings in a News Recommender System. *arXiv preprint arXiv:1805.05447* (2018).

[52] George Tsatsaronis, Georgios Balikas, Prodromos Malakasiotis, Ioannis Partalas, Matthias Zschunke, Michael R Alvers, Dirk Weissenborn, Anastasia Krithara, Sergios Petridis, Dimitris Polychronopoulos, et al. 2015. An overview of the BIOASQ large-scale biomedical semantic indexing and question answering competition. *BMC bioinformatics* 16, 1 (2015), 138.

[53] Manisha Verma and Debasis Ganguly. 2019. LIRME: Locally Interpretable Ranking Model Explanation. In *Proceedings of the 42Nd International ACM SIGIR*.

[54] Ellen M Voorhees. 1994. Query expansion using lexical-semantic relations. In *SIGIR'94*. Springer, 61–69.

[55] Ellen M Voorhees, Donna K Harman, et al. 2005. *TREC: Experiment and evaluation in information retrieval*. Vol. 63. MIT press Cambridge.

[56] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *International Conference on Machine Learning*. 2048–2057.

[57] Yang Xu, Gareth JF Jones, and Bin Wang. 2009. Query dependent pseudo-relevance feedback based on wikipedia. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*. ACM, 59–66.

[58] Yongfeng Zhang, Yi Zhang, Min Zhang, and Chirag Shah. 2019. EARS 2019: The 2Nd International Workshop on ExplainAble Recommendation and Search. In *Proceedings of the 42Nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'19)*. ACM, New York, NY, USA, 1438–1440. https://doi.org/10.1145/3331184.3331649