LIRME: Locally Interpretable Ranking Model Explanation

Manisha Verma
Verizon Media, New York, USA
manishav@verizonmedia.com

Debasis Ganguly
IBM Research, Dublin, Ireland
debasis.ganguly1@ie.ibm.com

ABSTRACT

Information retrieval (IR) models often employ complex variations in term weights to compute an aggregated similarity score of a query-document pair. Treating IR models as black-boxes makes it difficult to understand or explain why certain documents are retrieved at top-ranks for a given query. Local explanation models have emerged as a popular means to understand individual predictions of classification models. However, there is no systematic investigation that learns to interpret IR models, which is in fact the core contribution of our work in this paper. We explore three sampling methods to train an explanation model and propose two metrics to evaluate explanations generated for an IR model. Our experiments reveal some interesting observations, namely that a) diversity in samples is important for training local explanation models, and b) the stability of a model is inversely proportional to the number of parameters used to explain the model.

CCS CONCEPTS

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

KEYWORDS

Interpretability, Ranking, Point-wise explanations

ACM Reference format:
Manisha Verma and Debasis Ganguly. 2019. LIRME: Locally Interpretable Ranking Model Explanation. In *Proceedings of Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, Paris, France, July 21–25, 2019 (SIGIR '19),* 4 pages.
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

It has been shown that complex machine learning (ML) models can encode different biases [4] and may present myopic results [2] to users, leading researchers to investigate ways of 'explaining' model outcomes. Providing explanations can also be now required by law, mandated by regulations such as GDPR [1], wherein commercial ranking services may now be required to 'explain' its consumers why a document is retrieved for a query. A wide range of model-agnostic local explanation approaches have been proposed for ML tasks [5, 8], most of them focusing on providing an instance-wise explanation of the output of the model as either a subset of input features [3, 10], or a weighted distribution of feature importance [6, 9]. While the explanation space itself and methods to generate explanations are widely known in practise for classification tasks, their utility is largely unexplored for ranking tasks. There is little existing work in the IR community to systematically investigate ways of generating explanations for an IR model. Given that IR models involve complex variations in term weighting functions for scoring query-document pairs, some models may not be easy to 'explain' to a search engine user, who may have questions such as 'Why does a search engine retrieve document D at rank k?'.

In this work, with the motivation of 'explanations' in IR, we explore ways of *generating* and *evaluating* explanations. We focus on model-agnostic point-wise explanations, i.e. estimating 'explanation vectors' with respect to a retrieval model without any knowledge about its internals. The weight of each token in the term importance vector indicates its contribution to a ranking model's output for a given query-document pair. This vector can then be analyzed to see the relative importance of terms contributing positively (e.g. frequent presence of informative terms) or negatively (otherwise) to the score of a document.

To estimate the explanation vector for a query-document pair (Q, D), it is useful to study how a model behaves on variations of D. Since we aim to generate model-agnostic explanations, we study a model's behavior by examining the retrieval scores of different sub-samples drawn from D. Each IR model, due to its different ways of addressing the document lengths, term weights or collection statistics, may behave differently on these sub-samples. Local explanation models will be as effective as the words sampled for producing explanations. Explanation models (such as [9]) trained on poor samples will generate noisy and illegible explanations for documents. Therefore, we investigate three ways of sampling terms for model training and evaluate these sampling strategies for different explanation lengths. Our proposed explanation model then seeks to capture local effects on the sub-samples and predict a distribution of term importance potentially capturing the IR model's inherent term weighting characteristics.

We propose two metrics for evaluating the stability and correctness of the generated explanations. The first metric evaluates the sensitivity of the explanation model, i.e. the scale of change in explanations with change in model parameters. The second evaluation metric is more specific to IR, in which we evaluate the effectiveness of the explanations in terms of document relevance. Our experiments on the TREC ad hoc dataset indicate that sampling methods that are biased with tf-idf or positional information produce weaker explanations than those generated by uniformly sampling words from documents. We also found that explanation stability decreases with an increase in the number of explanation words; and that this effect is more pronounced for non-relevant documents.

SIGIR '19, July 21–25, 2019, Paris, France

ACM ISBN 978-1-4503-6172-9/19/07…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

2 RELATED WORK

Singh et. al. [12] investigate methods to train an explanation model for a given base ranker. The document rankings generated from the explanation model are then compared to those generated by the base model. Contrastingly, in our work, the focus is to evaluate different sampling strategies and automatically evaluate the consistency and effectiveness of explanation models. Specifically, our work explores methods to *generate* data for explanation model training and *evaluate* its effectiveness across queries. The metrics and sampling methods explored in this work can be adapted easily to train and evaluate new explanation models such as those proposed in [11, 12]. We do not investigate ways of transforming document scores into class probabilities [13], as explanation models can be trained directly on 'scores' assigned by any ranker (even those trained using deep learning). Our sampling methods and evaluation metrics are ranker agnostic and can be used to evaluate explanations of different lengths across queries.

3 LOCAL EXPLANATION OF IR MODELS

In classification tasks, model predictions can be *understood* by analyzing the predictions of simpler (human interpretable) variations of the data instances [9]. These simpler and human interpretable explanations take different forms for different tasks, e.g., the authors of [9] argue that one can explain the labels of images predicted by a classifier with a number of focused regions extracted from an input image. Similarly, for text classification, one can examine the changes in predictions of the classifier when different subset of terms are sampled from an input document.

Since the working principle of a ranking task is different from that of classification, in this work we investigate different ways of generating model-agnostic interpretable explanations for ranked lists. Formally, given a set of documents $\mathcal{D}$ and a query Q, the ranking function $S(D, Q)$ induces a total order on the set $\mathcal{D}$. For traditional IR models, such as BM25 or language models (LM), the similarity function is computed by aggregating the term weight contributions from matching terms. A ranking function can be represented as $S(D, Q) = \sum_{t \in D \cap Q} w(t, D)$, where $w(t, D)$ represents the term weight of term t in a document D [7].

To generate explanations, it is required to select set of simple instances or sub-instances, where each sub-instance is comprised of partial information extracted from a particular document. We employ a weighted squared loss to predict the score of the entire document D with respect to the input query. We call this method - locally interpretable ranking model explanation (**LIRME**).

$$\begin{aligned}\mathcal{L}(D, Q, \sigma; \boldsymbol{\Theta}) &= \sum_{i=1}^{M} \rho(D, D'_i) \left(S(D, Q) - S_{\boldsymbol{\Theta}}(D'_i, Q)\right)^2 + \alpha|\boldsymbol{\Theta}| \\ &= \sum_{i=1}^{M} \rho(D, D'_i) \left(S(D, Q) - \sum_{j=1}^{p} \theta_j w(t_j, D'_i)\right)^2 + \alpha|\boldsymbol{\Theta}|. \end{aligned} \tag{1}$$

In Equation 1, $D'_i = \sigma_i(D)$ denotes the i^{th} sample extracted from a document D comprised of p unique terms; α is an L1 regularization term; and $\boldsymbol{\Theta} \in \mathbb{R}^p$ denotes a vector of p real-valued parameters used to approximate the score of the sub-sample D'_i with respect to the query Q. Additionally, the weight of the loss $\rho(D, D'_i)$, is a similarity between the document D and its sub-sample D'_i. A standard way to define ρ in Equation 1 is with a kernel function of the form

$$\rho(D, D') = \exp(-\frac{x^2}{h}), \;\; x = \arccos(D, D') \tag{2}$$

where $\arccos(D, D')$ denotes the cosine-distance (angle) between a document D and a sub-document sampled from it, and h denotes the width of a Gaussian kernel.

The weighted loss function of Equation 1 predicts $S(D, Q)$ using the given samples. Since a retrieval model computes the score of an entire document and also the scores of its sub-samples, the predicted vector $\hat{\boldsymbol{\Theta}} \in \mathbb{R}^p$ estimates the importance of each term, e.g. the j^{th} component of $\hat{\boldsymbol{\Theta}}$ denotes the likelihood of term t_j in contributing positively to the overall score $S(D, Q)$.

It is expected that weights in $\hat{\boldsymbol{\Theta}}$ that correspond to a query term will have larger weights (denoting higher importance). Non-query terms with high weights in $\hat{\boldsymbol{\Theta}}$ are potentially the ones that are semantically related to the query and hence are likely to be relevant to its underlying information need. A visualization of these terms may then provide the desired explanation of an observed score of a document D with respect to Q (high or low).

3.1 Sampling of Explanation Instances

We now describe three different ways to define the sampling function $\sigma(D)$ that can be used to construct a set of samples around the neighbourhood of D for the purpose of predicting the parameter vector, $\hat{\boldsymbol{\Theta}}$, to explain a retrieval model.

Uniform Sampling:. A simple way to sample from the neighbourhood of an given document D is to sample terms with a uniform likelihood (with replacement). This ensures that there is no bias towards term selection leading to likely generation of a diverse set of samples for a document.

Biased Sampling:. Another way to sample terms is to set the sampling probability of a term proportional to its tf-idf weight seeking to generate sub-samples with informative terms.

Masked Sampling:. In contrast to a bag-of-words based sampling approach, an alternative way is to extract segments of text from a document, somewhat analogous to selecting regions from an image [9]. More specifically, in this sampling method we first specify a segment size, say k, and then segment a document D (comprised of $|D|$ tokens) into $\frac{|D|}{k}$ number of chunks. A chunk is then made visible in the sub-sample with probability v (a parameter).

4 EXPLANATION EVALUATION METRIC

We now consider ways of automatically evaluating the quality of an explanations generated using different sampling methods. Since it is costly and laborious to manually label the quality of explanations for each query-document pair, we propose two metrics that exploit *relevance judgments* to measure explanation quality at scale. We focus on 2C's – *consistency* and *correctness* for evaluating explanations described in following sections.

4.1 Explanation Consistency

An explanation vector $\hat{\boldsymbol{\Theta}}_{Q,D}$ can be used to determine which terms are important for explaining the score of a document D with respect to a query Q, i.e. $S(D, Q)$. The first desirable quality of an explanation method is that the relative ranking of important terms should

not change significantly with variations in the parameters of the model, or in other words, a particular choice of samples around the pivot document, D, should not result in considerable differences in the predicted explanation vector.

Variances in term rankings (explanation terms sorted in decreasing order by their weights) can be measured with weighted inter-rank correlations of a particular query-document pair averaged over a number of samples. More specifically, these correlations are computed between the ordered lists of the explanation vector and the ground-truth terms, $R(Q)$ derived from documents judged relevant for the query Q. In particular, the term sampling probabilities from this ground-truth set, $R(Q)$, are used to derive the reference ordered list for computing rank correlations. As sampling probabilities, we used LM-JM, i.e., language modeling with collection smoothing (Jelinek-Mercer) weights to induce a reference order on $R(Q)$, with an objective to sample frequently occurring informative terms from $R(Q)$. Formally speaking,

$$\Theta_{R(Q)}(w) = \lambda \frac{f(w, R(Q))}{\sum_{v \in R(Q)} f(v, R(Q))} + (1-\lambda)\frac{\texttt{cf}(w)}{\texttt{cs}}, \tag{3}$$

where $\Theta_{R(Q)}$ denotes the set of relevant terms extracted from $R(Q)$, f and $\texttt{cf}$ respectively denote term and collection frequencies, and $\texttt{cs}$ denotes collection size. We then assume that an ideal explanation system should seek to predict the same ranking of terms as induced by the decreasing order of term weights. Formally speaking, if $\psi(\hat{\Theta}_{Q,D})$ denotes a *sequence* of terms sorted by the decreasing values of the components in $\hat{\Theta}_{Q,D}$ (predicted from Equation 1), then we measure the average rank correlation coefficient with respect to the ground-truth ranking of terms as

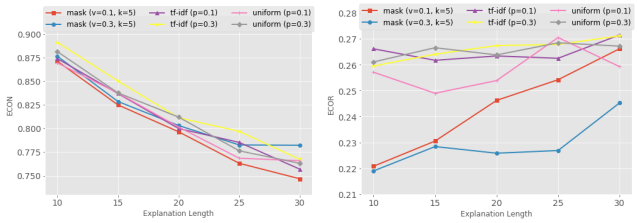
$$\begin{aligned} \gamma(Q, D) &= \frac{1}{|\Omega|} \sum_{\hat{\Theta}_{Q,D} \in \Omega} \tau(\psi(\hat{\Theta}_{Q,D}), \Theta_{R(Q)}) \\ \texttt{ECON} &= \frac{1}{|\boldsymbol{Q}|} \sum_{Q \in \boldsymbol{Q}} \sum_{D \in \texttt{TOP}(Q)} \gamma(Q, D), \end{aligned} \tag{4}$$

where Ω represents the set of different explanation vectors obtained with different samples, e.g. variations in the L1-regularization and kernel widths of LIRME (h in Equation 2), $\texttt{TOP}(Q)$ denotes the set of top-retrieved documents for a query Q and τ denotes the Kendall's rank correlation coefficient between the predicted and the ground-truth ordering of relevant terms. Note that the individual correlation scores $\gamma(Q, D)$ for a query-document pair are averaged over a set of benchmark queries $\boldsymbol{Q}$ present in collection judgments such as TREC datasets.

4.2 Explanation Correctness

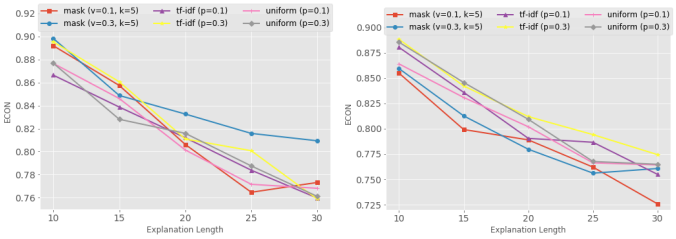
Intuitively, an explanation may be considered to be effective if it attributes higher weights to the components of $\hat{\Theta}_{Q,D}$ that correspond to relevant terms, i.e. the terms occurring in documents that are judged relevant by assessors. We measure explanation correctness by computing similarity between explanation vector terms $\hat{\Theta}_{Q,D}$ and relevant terms $R(Q)$. In particular, for a query-document pair (Q, D) we compute correctness as

$$\texttt{ECOR} = \frac{1}{|\boldsymbol{Q}|} \sum_{Q \in \boldsymbol{Q}} \sum_{D \in \texttt{TOP}(Q)} \frac{\hat{\Theta}_{Q,D} \cdot R(Q)}{|\hat{\Theta}_{Q,D}||R(Q)|} \tag{5}$$



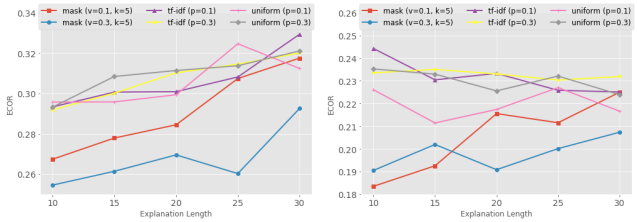
(a) Consistency (ECON) **(b) Correctness (ECOR)**

Figure 1: ECON and ECOR for top-5 retrieved documents.



(a) Relevant documents **(b) Non-relevant documents**

Figure 2: ECON for relevant and non-relevant documents.



(a) Relevant documents **(b) Non-relevant documents**

Figure 3: ECOR for relevant and non-relevant documents.

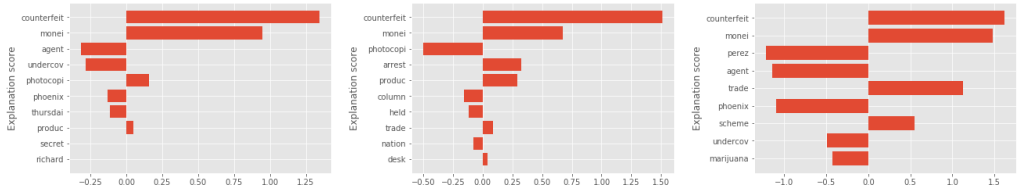
where $R(Q)$ represents the distribution of terms in the judged relevant documents. Similar to consistency ECON. we aggregate the relevance similarity values over a set of queries and number of top documents retrieved for each query.

5 EXPERIMENTS

The objectives of our experiments are to investigate - a) what term sampling approaches are effective in terms of the metrics consistency and ECOR (Section 4) to explain the scores assigned to query-document pairs by a retrieval model; and b) what is the optimal size of the explanation vector, i.e., the number of explanation terms (p) for yielding consistent and relevant explanations.

For our experiments, we use a standard benchmark dataset, namely the TREC-8, comprising 50 topics. To generate different sub-samples for explanation, we also employ uniform kernel in addition to Gaussian kernel, i.e. apply $\rho(D, \sigma(D)) = 1$. We generate explanations for top 5 documents for each TREC-8 query Q, i.e. $\texttt{TOP}(Q) = 5$ that are retrieved with LM-JM. The LM-JM score values constitute the $S(D, Q)$ values in Equation 1. For all our experiments, we set M (i.e. the number of document sub-samples) to 200.

In Figure 1a, we report expected consistency (ECON) values with a number of different LIRME settings. An interesting observation is that bag-of-words sampling (both uniform and tf-idf biased) yield



(a) Uniform sampling **(b) Tf-idf sampling** **(c) Masked samples ($v = 0.1$, $k = 5$)**

Figure 4: Visualization of explanation vectors $\hat{\Theta}(Q, D)$ estimated for a sample (relevant) document 'LA071389-0111' (D) and query (Q) 'counterfeiting money' (TREC-8 id 425). The Y-axis shows explanation terms, while the X-axis plots their weights.

higher consistency than the masking-based sampling. This indicates that on an average the relative ranks of the term weights in the explanation vector $\hat{\Theta}$ are more stable for these two sampling methods in comparison to the masking based sampler. Our experiments indicate that samples generated from a masking-based sampler exhibit less diversity because of a smaller degree of freedom in choosing individual terms independent of its context. Moreover, in the case of bag-of-words, uniform sampling yields higher consistency (ECON values). This is because the set of samples, D'_i tend to become similar to each other if the sampling strategy is biased towards informative terms. This again leads to a smaller variances in the scores of the sub-samples, which prevents the explanation loss function to effectively learn the term weights contributing to large increments or decrements in the similarity scoring function, $S(D, Q)$. Similar observations can be made if the averaging is split across the two partitions of relevant and non-relevant documents for each query in the dataset.

Biased sampling with tf-idf weights results in higher ECOR (as seen from Figure 1b). This shows that with tf-idf biased sampling, the explanation vector is better able to predict the terms frequently occurring in the relevant documents. The masking sampler produces worse results in terms of ECOR in comparison to the bag-of-words sampling approaches. While uniform sampling results in higher consistency, tf-idf biased sampling shows a higher recall with respect to the set of relevant terms. ECOR values tend to increase with the size of the explanation vector (p), which indicates that explanations are more effective with a higher number of parameters. However, consistency values ECON tend to decrease with increase in the model parameters because of the increase in likelihood of a change in the relative ordering of the terms by predicted weight values. When split across the set of relevant and non-relevant documents separately, it can be seen from Figure 3a that for relevant documents, ECOR values increase with more explanation terms relevant, whereas for the non-relevant ones these values tend to decrease. This behaviour shows that LIRME is able to predict high weights for true relevant terms.

For each sampling strategy investigated, Figure 4 plots the terms with their associated weights from explanation vectors, $\hat{\Theta}$, as histograms for a document judged relevant for the query 'counterfeiting money' (TREC-8 query id 425). All explanation vectors, independent of the sampling strategy, contribute positive weights to the terms constituting the query (e.g. see the weights of the word 'counterfeit'). However, for non-query terms, the explanation weights vary across sampling methods which indicates that the choice of sampling method can considerably impact the quality of the local explanations generated by any LIRME. Another observation is that sampling approaches were mostly able to find terms (output as negative weights) that are seemingly not relevant to the information need of the example query, such as the terms 'phoenix', 'agent' etc. From a user's point-of-view, the positive weights of a set of terms in a document are likely to help him discover the associated relevant sub-topics within it, whereas the negative weights on the other hand could indicate potential non-relevant aspects.

6 CONCLUSION

While research in explaining outputs of classification models exists, there is little work on explaining results of a ranking model. In this work, we addressed the research question: Why does an IR model assign a certain score (affecting its rank) to a document D for a query Q? We investigated the effectiveness of different sampling methods in generating local explanations for a document scored with respect to a query by an IR model. We also proposed two evaluation metrics to measure the *consistency* and *correctness* of explanations generated using different sampling methods. Our experiments indicate that sampling methods that use term position information produce weaker explanations than those generated by uniform or tf-idf based sampling of words from documents.

REFERENCES

[1] E. Alepis, E. Politou, and C. Patsakis. Forgetting personal data and revoking consent under the GDPR: Challenges and proposed solutions. *Journal of Cybersecurity*, 4(1), 03 2018.
[2] T. Bolukbasi, K. Chang, J. Y. Zou, V. Saligrama, and A. Kalai. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. *CoRR*, abs/1607.06520, 2016.
[3] J. Chen, L. Song, M. J. Wainwright, and M. I. Jordan. Learning to explain: An information-theoretic perspective on model interpretation. *arXiv preprint arXiv:1802.07814*, 2018.
[4] L. Dixon, J. Li, J. Sorensen, N. Thain, and L. Vasserman. Measuring and mitigating unintended bias in text classification. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pages 67–73. ACM, 2018.
[5] Z. C. Lipton. The mythos of model interpretability. *arXiv:1606.03490*, 2016.
[6] S. M. Lundberg and S.-I. Lee. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems*, pages 4765–4774, 2017.
[7] D. Metzler and W. Bruce Croft. Linear feature-based models for information retrieval. *Inf. Retr.*, 10(3):257–274, June 2007.
[8] G. Montavon, W. Samek, and K. Müller. Methods for interpreting and understanding deep neural networks. *CoRR*, abs/1706.07979, 2017.
[9] M. T. Ribeiro, S. Singh, and C. Guestrin. Why should i trust you?: Explaining the predictions of any classifier. In *Proc. of KDD'16*, pages 1135–1144, 2016.
[10] M. T. Ribeiro, S. Singh, and C. Guestrin. Anchors: High-precision model-agnostic explanations. In *AAAI Conference on Artificial Intelligence*, 2018.
[11] J. Singh and A. Anand. Interpreting search result rankings through intent modeling. *arXiv preprint arXiv:1809.05190*, 2018.
[12] J. Singh and A. Anand. Posthoc interpretability of learning to rank models using secondary training data. *arXiv preprint arXiv:1806.11330*, 2018.
[13] J. Singh and A. Anand. EXS: Explainable Search Using Local Model Agnostic Interpretability. In *Proc. of WSDM '19*, pages 770–773, 2019.