Explaining Black Box Models for Document Retrieval

Karen Tu
Department of Computer Science, Brown University
Providence, RI
karen_tu@brown.edu

In the evaluation of information retrieval models, ranking performance has long been considered the benchmark measure of model functionality. It has become apparent, however, that merely looking at accuracy and precision fails to capture the complexity of model behavior, allowing for the persistence of models that reinforce biases of unfair real-world data. It is often difficult to apply standard fairness measures like statistical parity or to analyze a standard confusion matrix in ranking contexts. This study proposes an alternative method of analyzing the global behavior of ranking models through the aggregation of model agnostic local linear explanations. Using a LambdaMART model trained on a eighteen-feature dataset, we learned locally interpretable model explanations (LIME) for each data point in the held-out test set and used their aggregated weights to make judgements on the overall model's decision making structure. From the LambdaMART model, we found that stream length, the summary statistics associated with term frequencies, and the summary statistics associated with $TF * IDF$ scores were the most consistently influential in the model's decision making.

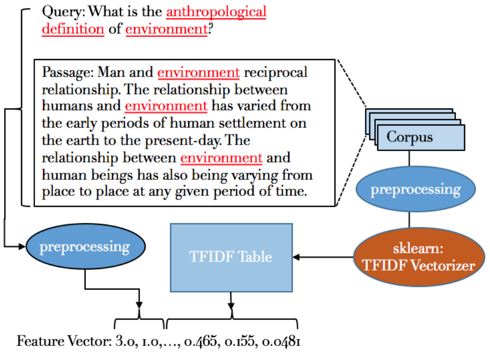
I. INTRODUCTION

In the field of information retrieval, the application of black box machine-learned models has become increasingly ubiquitous. The pervasiveness of these tools in a variety of application settings, from everyday commercial use to medicine and research, warrants a closer look at their potential for misunderstanding and misuse. In particular, models must address both a responsibility to provide the user with the items that they wish to retrieve and a duty to expose relevant items to the user [1].

Information retrieval models are generally trained on a data-set of features related to the frequency with which certain words appear in a query and a document and a range of other metrics of relation between a query and a potentially related document (Figure 1). In routine use, these models learn a complex model and output a prediction, score, or ranking without revealing the internal mechanics of how each feature influences the final rankings.

While each feature in a ranking data-set might have a directly human-interpretable reasoning for its influence in a ranking algorithm, the complexity introduced by combining expansive feature sets via sophisticated nonlinear functions obscures this meaning. Therefore, developers find it progressively more difficult to identify unfairness, bias, or artifacts in their models, even if the original data features were intelligible.

Furthermore, the ability to interpret the reasoning behind a machine learning model has serious practical implications on an engineer's ability to ascertain the functionality of the model. The current model uses some combination of performance metrics like normalized cumulative discounted gain (nDCG)[2], Precision@K [3], or Kendall's Tau [4] and checking a sample of data points to verify functionality. Nevertheless, if a newly developed model does not perform predictably or accurately according to these metrics, it remains difficult to establish precisely where the existing model errs or identify concrete changes that need to be made.


Query: What is the anthropological definition of environment?

Passage: Man and environment reciprocal relationship. The relationship between humans and environment has varied from the early periods of human settlement on the earth to the present-day. The relationship between environment and human beings has also being varying from place to place at any given period of time.

FIG. 1: Sample pipeline to transform query-passage pairs into feature vectors: Both queries and passages are preprocessed using techniques like stopping, stemming, etc. The sample feature vector contains features directly calculated from the processed query and document (referring to the first two values), as well as features generated from TF and IDF scores calculated based off the entire corpus at large (the latter values).

In addition to the technical barriers that unintelligible models pose, the 'black-box' system also raises questions of technological trust, especially in real-world deployment. As unpredictable use cases arise, the ability to regulate, audit, and reproduce ranking results becomes relevant in developing legislation, policy, and even potentially establishing legal liability[5]. There are many potential issues with using the performance of the model on a subset of the original data source to predict performance in real world applications. Although the test set is held out, the possibility of data leakage, either through generated features or through the original data source, remain ever present [6]. The selection of representative data points for point checking also generates potential issues with the completeness and fairness of using those points to extrapolate to a larger test set or to the real world at large. Thus, if an search application wishes to deploy a trustworthy ranking model, the decision-making structure of that model should be carefully evaluated.

The remainder of the paper is structured as follows: Section II presents background information on the definition of explainable models and model agnosticity. Section III introduces relevant work in the field of explaining information retrieval models and presents the specific models (LIME and LambdaMART) that this paper works with. Section IV provides an overview of methods of data generation and the training of the ranking and explanation models. Section V presents the results later discussed in section VI. Lastly, Section VII explores potential further research moving forward.

II. BACKGROUND

An explainable ranker, as described by Singh and Anand, must empower users and developers to answer the following questions:

1. 'What is the intent of the query according to the ranker?'
2. 'Why is one document ranked higher than another?'
3. 'Why is this document relevant to the query?'

This study focused on the latter two questions, examining features that quantify relationship between a query and document and features that compare the relevance of the document of interest compared to the rest of the corpus [7].

In creating an explanation for some model, developers are challenged to balance the fidelity-interpretability tradeoff. That is, while the entire rationale for the existence of an explaining model relies on its ability to effectively emulate model behavior, an explanation that does this perfectly by recreating the model is useless to a user that cannot match that explanation to an interpretable real-world meaning. On the other hand, an explaining model that only reinforces what the user already knows (and is thus maximally interpretable) but performs poorly compared to the actual model will never be used.

So far, a small subset of models are described as inherently "interpretable," including decision trees, rules, additive models, and linear models[8]. These models can be grouped together due to their ability to be closely inspected and interpreted. For example, each node of a decision tree can be inspected and evaluated for what quantitative judgement is made at that point.

This project follows a different approach: model agnostic interpretability [9]. A model agnostic explanation can be applied to any ranking model, regardless of its implementation, so long as the explaining algorithm can access the inputs and outputs of the ranking model. Model agnosticity allows for flexibility in the types of models that could potentially be used to perform a task and in the format of the explanation. Additionally, model agnostic explainability allows for the use of more powerful algorithms that are not limited by the immediate explainability of preserving a white box model.

III. RELATED WORK

The importance of model agnosticity is especially apparent considering the variety in approaches to the learn-to-rank problem. The models RankSVM[10], RankBoost[11], and RankNet[12] all approach ranking problems as a pairwise classification but employ a variety of different training methods: support vector machines, boosting, and neural nets, respectively. Additional layers of complexity, like that introduced by relevance matching [13–15] further improve performance and distance the model from human interpretability. The rapid evolution of the components of top performing models makes post-hoc, model-agnostic explanations a necessity, one recognized by much of the research in the field.

The work around model agnostic explainable models can be split into two approaches: global and local explanations. Global explanation models attempt to capture the behaviour of more complex models by creating models that behave like that more complex model, but are inherently interpretable, like logic rules [16] or decision trees [17]. The simplicity of these models, however, makes it difficult for them to maintain fidelity to the complex global behavior of a the real ranking model, even with more secondary training data and modified learning approaches. Furthermore, studies found that global techniques are sensitive to different ranking techniques, making it difficult for global models to be truly model agnostic [18].

On the other hand, local explanations apply similar methods on the scale of a single data point. While these explanations cannot be generalized to the global model, they attempt to provide some insight into the behavior of the model around the specific portion of the input space where the datapoint of interest lies.

Using the existing, complex black-box model, the explainer perturbs the datapoint and uses the outputs from the complex model to generate a linear model of local behavior. By nature, the linear model is interpretable, as the weights of the linear model can be surfaced and visualized.

LIME (Locally Interpretable Model-Agnostic Explainer) uses this method to explain the output of models built to solve classification problems [9]. Given some black-box model M, LIME trains a linear SVM, E, that minimizes the loss between the predictions output by E and the original model M for some input value i. The linear model is created by generating a training set of data from perturbing i randomly.

Because LIME is built to explain binary classification problems, the score-based ranking output must be converted to a binary classification. We use the following three methods described by Singh and Anand [7].

1. **Top-k:** All passages ranked higher than some cut-off k are classified as relevant:

   $$P(relevant|q,d) = 1$$

   where q, d represents the query and document of interest, respectively. All other passages receive a classification as irrelevant:

   $$P(relevant|q,d) = 0$$

2. **Rank-based:** All passages ranked higher than some cut-off k are classified as relevant.

   $$P(relevant|q,d) = 1$$

   All other passages are classified assigned a probability score defined by the following formula:

   $$P(relevant|q,d) = 1 - \frac{R(q,d_1) - R(q,d)}{R(q,d_1)}$$

   where $R(q,d_1)$ represents the score of the first ranked, most relevant document.

3. **Score-based:** All passages ranked higher than some cut-off k are assigned a probability score defined by the following formula:

   $$P(relevant|q,d) = 1 - \frac{rank(d)}{k}$$

   All other passages are classified as irrelevant:

   $$P(relevant|q,d) = 0$$

Other work, including this paper, builds on the LIME model. LIRME [19] investigated the stability and accuracy of LIME explanations applied to ranking models. LIRME, however, largely focused on sampling methodology and feature size. Likewise, [7] develops a system that uses LIME to explain the local results of a deep relevance matching model (DRMM), tracing ranking decisions back to query and document terms.

IV. METHODS

A. Data Generation

This project used a subset of the MS MARCO (Microsoft Machine Reading Comprehension) Passage Ranking dataset, originally generated from 100,000 anonymized Bing questions and human generated answers [20]. From that original dataset, this project used 3,603,276 unique query-document pairs. The original dataset was formatted in query id, positive id, and negative id triples. Triples were generated automatically from human-labelled passage relevance scores. Each triple consisted of a positively labelled passage and a second negative passage that was randomly-selected from a list of the top-1000 passages (scored using BM25) that were not labelled positively. This structure allowed us to generate two query-passage pairs for each original datapoint. Positive passage examples were assigned a score of 1 and negative passage examples were assigned a score of -1.

The data was structured as described in Table I. The eighteen features were chosen from the list of 46 descriptive statistics selected by Han and Lei as most influential out of a super set of 136 features [21]. The original Microsoft dataset contained document bodies, anchors, titles, and whole documents; thus, the feature size decreased when translated to the MS MARCO dataset which only includes passages. Additionally, some features are privately owned by Microsoft and were excluded from the dataset.

TABLE I: Feature Set Description

Index	Feature Name
1	covered query number
2	covered query ratio
3	stream length
4-8	term frequency (sum, min, max, mean, variance)
9-13	inverse doc. freq. (sum, min, max, mean, variance)
14-18	TF*IDF (sum, min, max, mean, variance)

B. Ranking Model

The dataset was split into training, validation, and test in a $90-5-5$ split. We then used the pyltr implementation of the LambdaMART boosted tree model to train a ranking model on each of the above datasets [22]. Compared to the 0.377-0.402 performance of state-of-the-art models published with the MS MARCO dataset for boosted trees, our model performance of 0.370 mean nDCG was reasonable, considering the loss of various features and smaller training set.

C. Explanation Model

For each of the remaining held-out test data points, we used LIME to learn a local linear SVM that described model behaviour around that point. Additionally, each feature was separated using quartiles, deciles, and entropy-based discretization and results from each method of discretization were collected.

V. RESULTS

Each feature value was categorized into two groups based on the weight that the explanation model for that datapoint assigned that specific feature. Positive weights

TABLE II: For each linear equation, the weights associated with each feature were grouped into positively influential (having a positive values) and negatively influential (having a negative value). Using the Mann-Whitney U-test, we evaluated the influence of the feature value on whether the feature weights would be positive or negative. For this test the null hypothesis H_0 was that the two distributions (positive and negative weights) were equal. For the tests that were statistically significant for an $\alpha = 0.05$ the cells were colored according to the directionality of the difference in distributions. Green cells represent a feature where the distribution of relevant weights was greater than the distribution of irrelevant weights, and vice versa for red cells.

	Quartile		Decile		Entropic	
Feature	Accept H_0	Reject H_0	Accept H_0	Reject H_0	Accept H_0	Reject H_0
Covered Query Number	0.079	—	—	1.211×10^{-7}	0.1244	—
Covered Query Ratio	—	0.029	—	3.28×10^{-6}	—	0.0003
Stream Length	—	0.000	—	0.000	—	0.000
Term Frequency (sum)	—	0.000	—	0.000	0.082	—
Term Frequency (min)	—	0.000	—	0.000	—	0.000
Term Frequency (max)	—	0.000	—	0.000	—	0.000
Term Frequency (mean)	—	0.000	—	0.000	—	0.000
Term Frequency(variance)	—	0.000	—	0.000	—	0.000
Inverse Document Frequency (sum)	0.134	—	—	0.00285	—	0.000
Inverse Document Frequency (min)	—	0.021	—	4.212×10^{-6}	—	0.000
Inverse Document Frequency (max)	0.407	—	0.051	—	—	0.000
Inverse Document Frequency (mean)	0236	—	—	$1.062 \times 10{-}7$	—	0.000
Inverse Document Frequency(variance)	—	0.000	—	0.000×10^{-7}	—	0.000
TF*IDF (sum)	—	0.000	—	0.000	—	0.000
TF*IDF (min)	—	0.000	—	0.000	—	0.000
TF*IDF (max)	—	0.000	—	1.56×10^{-12}	—	0.000
TF*IDF (mean)	—	0.000	—	0.000	—	0.000
TF*IDF (variance)	—	0.000	—	0.000	—	0.000

were labelled as markers of relevance, while negative weights were labelled as irrelevant. Each feature distribution was plotted on a histogram comparing the number of LIME models that classified the feature as a relevant feature to the number of models where that feature is classified as irrelevant.

In general, most features fell into one of two groups. One group of features were equally likely to be classified as relevant or irrelevant regardless of the actual value of the feature (Figure 2a). Other feature distributions were skewed based on the value of the feature (Figure 2b). Of these two groups, the latter provided more useful insight into model behavior, as features showing significant skew in favor of higher or lower values indicated that the value of those features could be predictive of the ultimate classification. Complete results repeated over different methods of discretization are shown in the Appendix.

In order to quantify this difference, we used the Mann-Whitney U-test to compare the two distributions (relevant and irrelevant). We tested a null hypothesis, H_0, that the two distributions are sampled from equal populations. Our alternate hypothesis was that the two populations are in fact unequal and one distribution is greater or lesser than the other. Overall, the influence of stream length and the statistics associated with term frequency and TF*IDF score were consistently distinguishable across all methods of discretization. The results of this test are shown in Table II.

Additionally, we visualized the distribution of the weights of certain variables plotted against the feature values. While many of the weights were relatively normally distributed, as shown in Figure 3a, other scatter plots showed significant skew towards either higher (Figure 3b) or lower (Figure 3c) values.

VI. DISCUSSION

The results shown above affirmed the conclusions of the conventional understanding of the selected features. The features that were most consistently influential (in either direction) were those associated with term frequency and TF*IDF score. Since both are representations of the shared vocabulary between a query and a passage, the influence of term frequency and TF*IDF score statistics in determining a good match between a query and a document was unsurprising. Furthermore, the relative inconsistency seen in inverse document frequency has also been previously documented [21]. While inverse document frequency intends to describe some dimension of passage uniqueness, the connection between these values and the overall relation between the query and passage was unclear. Furthermore, although it was possible that IDF values are not a good indicator of a good match, it may also be possible that the relation between inverse document frequency and relevance is more complex than a simple linear relationship and is not apparent through the analysis presented here.

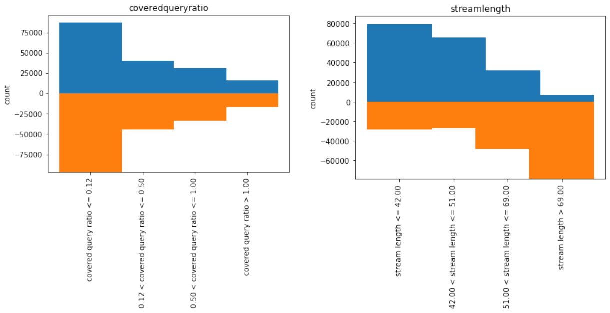


FIG. 2: From right to left: a) Histogram of covered query ratio distribution shows even distribution between relevant (blue) and irrelevant (orange). b) Histogram of stream length distribution shows that relevant weights are skewed towards low stream lengths and irrelevant weights are skewed towards high values of stream length.

It was also interesting to note that the breakdown between the different summary statistics remained consistent between term frequency and TF*IDF. Although this may be attributed to the mathematical relation between the two values, the observation may further support the theory that IDF is not as influential as the term-based values due to the lack of consistency in IDF results across different discretization methods.

Of note, the minimums and means of both values were distributed such that irrelevant weights were generally associated with higher feature values while the maxes and variances were distributed such that relevant weights were associated with higher feature values. These results were somewhat surprising; we would assume that values like a high term frequency or TF*IDF sum, mean, or minimum would indicate a query-document pair with more commonality and therefore represent greater relevance. Similarly, we would assume that high variance is indicative of groups of words that do not share similar levels of uniqueness and are less likely to return useful results. Although these results contradicted many of the assumptions that we might make, they provided some valuable insight into the inner workings of our ranking model. For example, weighting high variances positively may indicate that queries with high term variance concentrate meaning in fewer words and are thus easier to match to relevant documents.

Nevertheless, this style of analysis was limited by its ability to capture the interaction between different variables. For example, a complex model could capitalize on the interaction between the variance and sum of a feature like inverse document frequency across all the terms of the input query such that a query-document pair with a high variance and high sum is labelled irrelevant but a pair with a low variance and high sum is labelled irrelevant. While the aforementioned linear models can capture the behavior of individual features (variance of inverse document frequency, in this case), it failed to capture the interactions between different features.

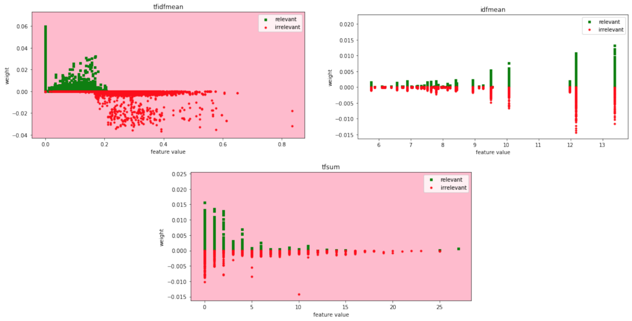


FIG. 3: Left to right and top to bottom: a) Weights evenly distributed as TF*IDF min increases b) Weight increases as idf mean increases c) Weight decreases as tf-sum increases

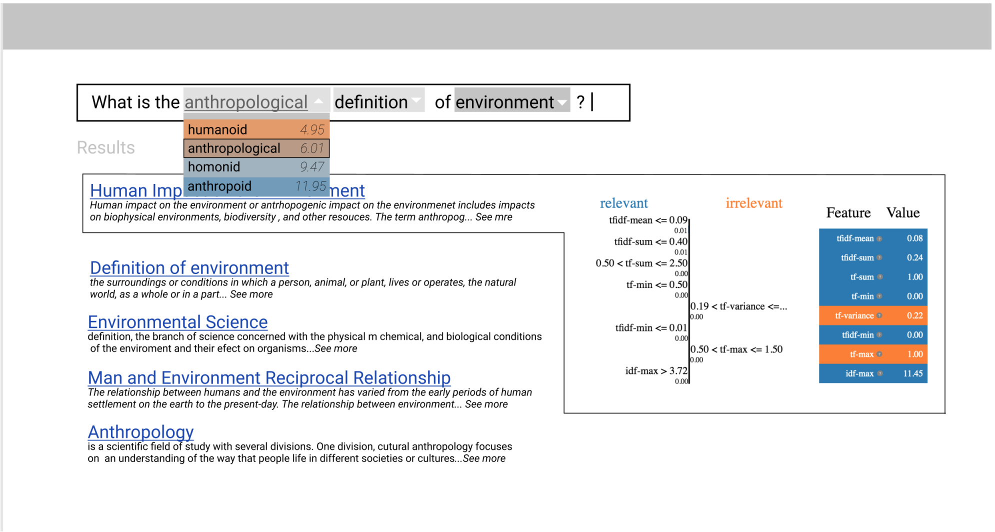
VII. CONCLUSION AND FUTURE WORK

In this paper, we considered a method of model-agnostic explanation that can provide insight into how a non-linear ranking model uses the features that it has been given. We used LambdaMART as a base ranking model and implemented a LIME explanation for test datapoints in order to visualize the local behaviour at various points in the input space. Although the behavior of local models cannot be used to make comprehensive judgements on global model behavior, our results show that there were several features that are consistently influential in determining either relevance or irrelevance that could be potential targets for further investigation.

One potential avenue of further work could center around the implementation of a search engine application to be used in user testing contexts. Two potential use cases are proposed below:

1. **Debugging/Model Improvement:** In order to effectively evaluate model performance and identify potential errors or biases before deployment, users who view these explanations may wish to modify or correct certain model behavior in order to improve performance. In this case, a user study to answers the question of whether or not the model explanation helps users understand what aspects of the model must be changed in order to produce some result on the output.

2. **Understanding Results:** A study surrounding using the visualization to better understand search results would aim to determine whether or not visualizations increased users' trust in the rankings that are produced by the model. Furthermore, it would potentially be interesting to see if users were able to modify their queries, either organically through modifying the raw text of the query or artificially by modifying values in the feature input, in order to obtain better results (Figure 4).



What is the anthropological definition of environment ? |

Human Imp ment
Human impact on the environment or antrhopogenic impact on the environmenet includes impacts on biophysical environments, biodiversity , and other resouces. The term anthropog... See mre

Definition of environment
the surroundings or conditions in which a person, animal, or plant, lives or operates, the natural world, as a whole or in a part... See more

Environmental Science
definition, the branch of science concerned with the physical m chemical, and biological conditions of the enviroment and their efect on organisms...*See more*

Man and Environment Reciprocal Relationship
The relationship between humans and the environment has varied from the early periods of human settlement on the earth to the present-day. The relationship between environment... See more

Anthropology
is a scientific field of study with several divisions. One division, cutural anthropology focuses on an understanding of the way that people life in different societies or cultures...*See more*

Feature	Value
tfidf-mean	0.08
tfidf-sum	0.24
tf-sum	1.00
tf-min	0.00
tf-variance	0.22
tfidf-min	0.00
tf-max	1.00
idf-max	11.45

FIG. 4: Using both aggregate and individual explanations, the above sample shows a possible implementation of a interface system that allows users to perform query expansions based on aggregate model behavior. The search bar drop-down menu for each keyword shows synonyms and a feature of interest (inverse document frequency, in this case) allowing users to modify their search terms and modify the feature array as it relates to IDF statistics. Possible synonyms for query terms are color-coded according to aggregated influence. In this example a higher IDF is indicative of relevance.

Lastly, there has been significant work done on the development of fairness-aware rankings systems, such as FA*IR [23] and Unbiased LambdaMART [24]. A potential future research direction could investigate the differences in the explanations generated around these ”fair” rankers and perhaps contribute developing systems of comparison between various rankers and allow users to better evaluate the differences between different ranking models.

ACKNOWLEDGMENTS

First and foremost, I would like to thank my advisor, Professor Carsten Eickhoff for his advice and support throughout this past year. This project would not exist without his expertise and guidance. I would also like to thank my second reader, Professor Ellie Pavlick, for her indispensable help, especially in the field of natural language processing.

I must also express my deep gratitude to my family and friends in the Computer Science community for everything they have done to support me through this process and for making Brown a home away from home.

Lastly, Claire, Michelle, Nate, and Roy - I could not have asked for a lovelier quarantine pod, thank you for keeping me grounded in the past months of turbulence.

[1] A. Singh and T. Joachims, Fairness of exposure in rankings, CoRR **abs/1802.07281** (2018), arXiv:1802.07281.

[2] K. Järvelin and J. Kekäläinen, Cumulated gain-based evaluation of ir techniques, ACM Trans. Inf. Syst. **20**, 422–446 (2002).

[3] K. Järvelin and J. Kekäläinen, Ir evaluation methods for retrieving highly relevant documents, in *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '00 (Association for Computing Machinery, New York, NY, USA, 2000) p. 41–48.

[4] M. G. Kendall, A New Measure Of Rank Correlation, Biometrika **30**, 81 (1938), https://academic.oup.com/biomet/article-pdf/30/1-2/81/423380/30-1-2-81.pdf.

[5] B. Goodman and S. Flaxman, European union regulations on algorithmic decision-making and a “right to explanation”, AI Magazine **38**, 50–57 (2017).

[6] S. Kaufman, S. Rosset, and C. Perlich, Leakage in data mining: Formulation, detection, and avoidance (2011) pp. 556–563.

[7] J. Singh and A. Anand, Exs: Explainable search using local model agnostic interpretability, (2018).

[8] R. Caruana, Y. Lou, J. Gehrke, P. Koch, M. Sturm, and N. Elhadad, Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission, in *Pro-*

ceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '15 (Association for Computing Machinery, New York, NY, USA, 2015) p. 1721–1730.

[9] M. T. Ribeiro, S. Singh, and C. Guestrin, "Why should I trust you?": Explaining the predictions of any classifier, CoRR **abs/1602.04938** (2016), arXiv:1602.04938.

[10] Large margin rank boundaries for ordinal regression, in *Advances in Large Margin Classifiers*, edited by A. Smola, P. Bartlett, B. Schölkopf, and D. Schuurmans (MIT Press, 2000) pp. 115–132.

[11] Y. Freund, R. Iyer, R. E. Schapire, and Y. Singer, An efficient boosting algorithm for combining preferences, J. Mach. Learn. Res. **4**, 933–969 (2003).

[12] C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, and G. Hullender, Learning to rank using gradient descent, in *Proceedings of the 22nd International Conference on Machine Learning*, ICML '05 (Association for Computing Machinery, New York, NY, USA, 2005) p. 89–96.

[13] Y. Qiao, C. Xiong, Z. Liu, and Z. Liu, Understanding the behaviors of bert in ranking (2019), arXiv:1904.07531 [cs.IR].

[14] J. Guo, Y. Fan, Q. Ai, and W. B. Croft, A deep relevance matching model for ad-hoc retrieval, Proceedings of the 25th ACM International on Conference on Information and Knowledge Management 10.1145/2983323.2983769 (2016).

[15] K. Hui, A. Yates, K. Berberich, and G. de Melo, Pacrr: A position-aware neural ir model for relevance matching (2017), arXiv:1704.03940 [cs.IR].

[16] V. I. S. Carmona, T. Rocktäschel, S. Riedel, and S. Singh, Towards extracting faithful and descriptive representations of latent variable models, in *AAAI Spring Symposia* (2015).

[17] M. W. Craven and J. W. Shavlik, Extracting tree-structured representations of trained networks, in *Proceedings of the 8th International Conference on Neural Information Processing Systems*, NIPS'95 (MIT Press, Cambridge, MA, USA, 1995) p. 24–30.

[18] J. Singh and A. Anand, Posthoc interpretability of learning to rank models using secondary training data, CoRR **abs/1806.11330** (2018), arXiv:1806.11330.

[19] M. Verma and D. Ganguly, Lirme: Locally interpretable ranking model explanation, in *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'19 (Association for Computing Machinery, New York, NY, USA, 2019) p. 1281–1284.

[20] P. Bajaj, D. Campos, N. Craswell, L. Deng, J. Gao, X. Liu, R. Majumder, A. McNamara, B. Mitra, T. Nguyen, M. Rosenberg, X. Song, A. Stoica, S. Tiwary, and T. Wang, Ms marco: A human generated machine reading comprehension dataset (2016), arXiv:1611.09268 [cs.CL].

[21] X. Han and S. Lei, Feature selection and model comparison on microsoft learning-to-rank data sets (2018), arXiv:1803.05127 [stat.AP].

[22] S. S. S. G. C. R. Jerry Ma, Luca Soldaini, pyltr (2017).

[23] M. Zehlike, F. Bonchi, C. Castillo, S. Hajian, M. Megahed, and R. Baeza-Yates, Fa*ir: A fair top-k ranking algorithm, CoRR **abs/1706.06368** (2017), arXiv:1706.06368.

[24] Z. Hu, Y. Wang, Q. Peng, and H. Li, A novel algorithm for unbiased learning to rank, CoRR **abs/1809.05818** (2018), arXiv:1809.05818.

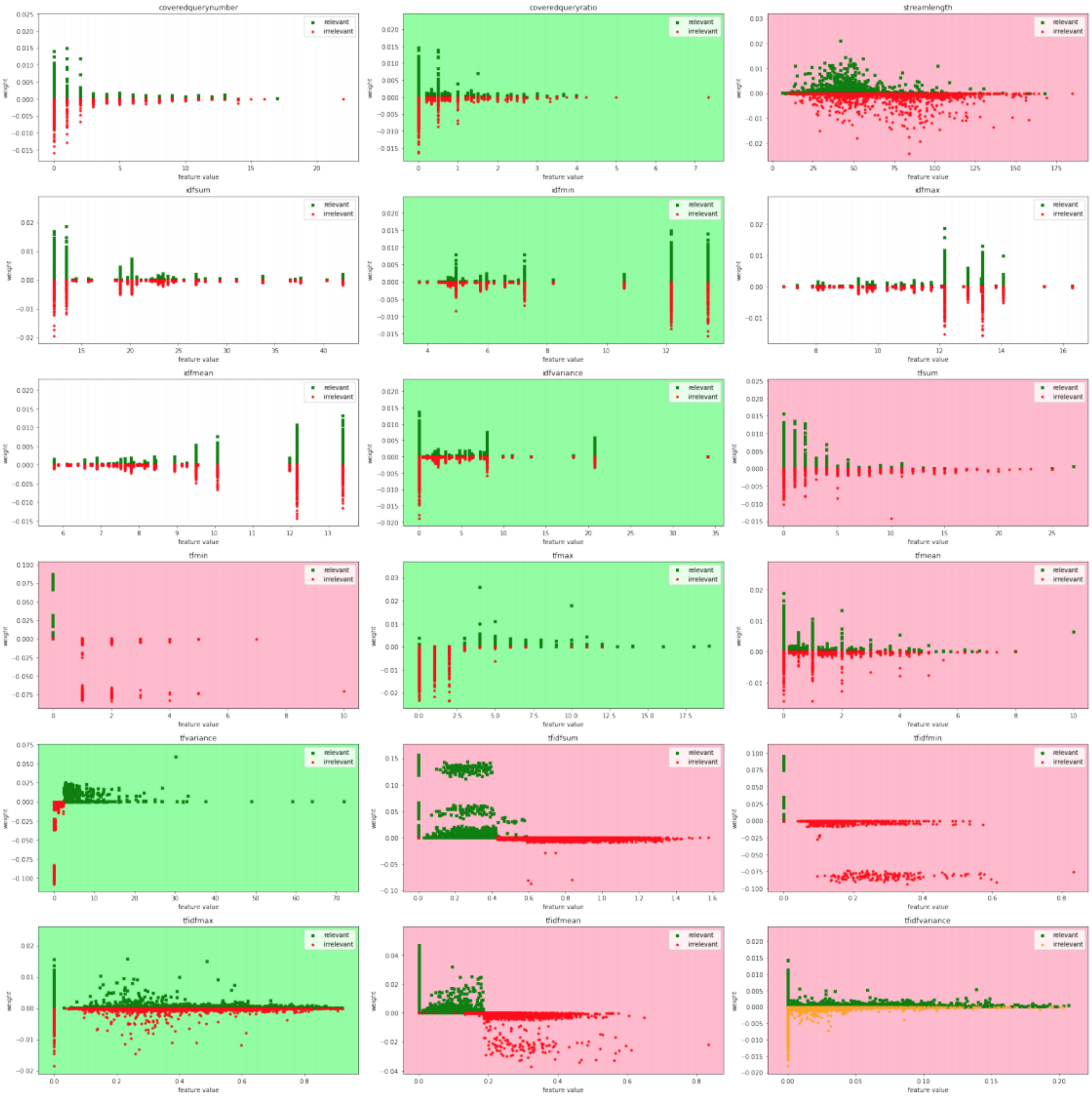


FIG. 5: Quartile Weights: Colored graphs indicate features where the difference between the two distributions was statistically significant. Red indicates that irrelevant weighted features are greater than relevant feature values and green represents the opposite.

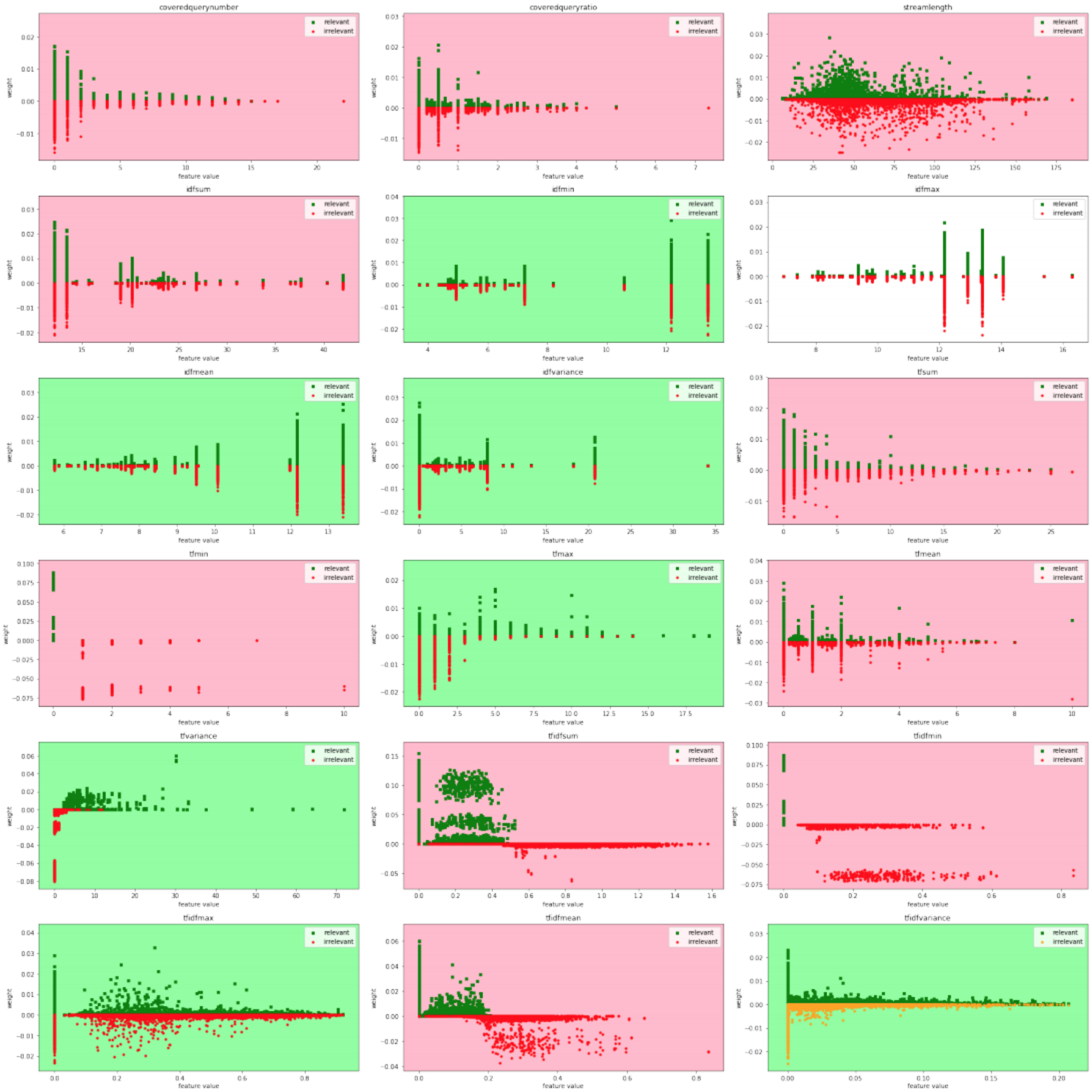


FIG. 6: Decile Weights

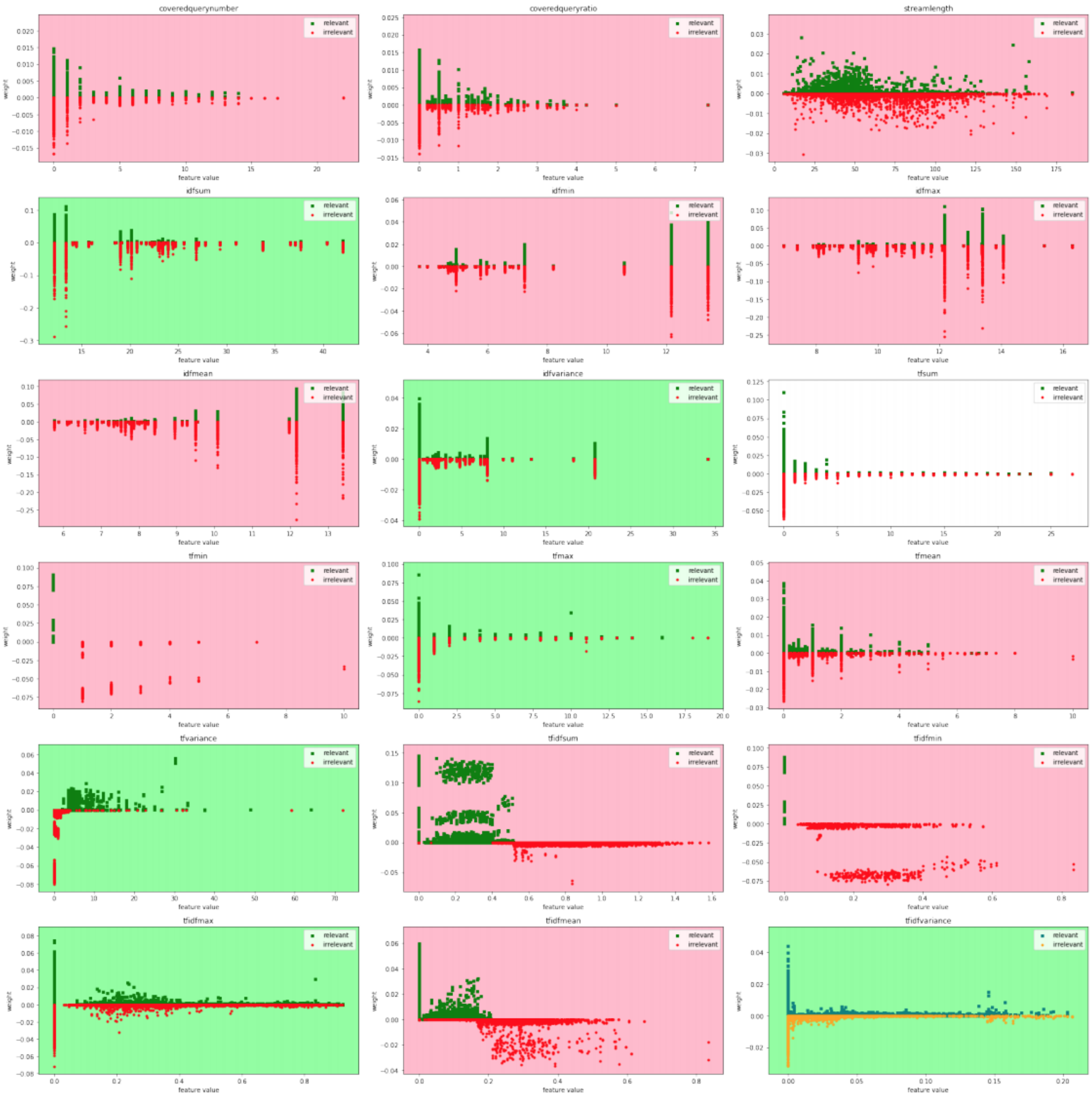


FIG. 7: Entropy Weights