Pairwise Review-Based Explanations for Voice Product Search

Gustavo Penha*
g.penha-1@tudelft.nl
Delft University of Technology
Delft, Netherlands

Eyal Krikon
krikon@amazon.com
Amazon
Seattle, USA

Vanessa Murdock
vmurdock@amazon.com
Amazon
Seattle, USA

ABSTRACT

Explanations describe product recommendations in a human interpretable way in order to achieve a goal, e.g. persuade users to buy. Unlike web product search, where users have access to diverse information as to why the products might be suitable for their needs, in the voice product search domain the amount of information that can be disclosed is inherently limited. Users in general evaluate a maximum of two products and usually buy low consideration products when using the voice channel [3]. In order to enable decision making in voice product searches we propose here a framework for generating pointwise and pairwise review-based explanations that disclose further information about the products. The POINTWISE method selects a helpful sentence from the top review of the recommended product based on a BERT-based model and uses the extracted sentence to fill a response template. The PAIRWISE method first selects a diverse pair of products—in terms of their review-based representations—from the top-k ranked products for a query, then chooses a helpful review sentence for each product in the pair, and finally fills a template with the sentences. Besides further describing the product, the PAIRWISE method gives a reference point to the users and enables a comparison of the recommendations based on two diverse products for the same information need. Our crowd-sourced evaluation of explanations based on queries from a widely used e-commerce platform shows that the proposed pairwise explanations provide statistically significant improvements compared to the POINTWISE and BASELINE methods for two goals: *Effectiveness*, i.e. helping users to make good decisions, and *Transparency*, i.e. explaining how the system works. The gains of PAIRWISE over POINTWISE and BASELINE are consistent for different subsets of data based on the diversity of the selected pairs, average product price associated with the query and the query ambiguity.

ACM Reference Format:
Gustavo Penha, Eyal Krikon, and Vanessa Murdock. 2022. Pairwise Review-Based Explanations for Voice Product Search. In *Proceedings of the 2022 ACM SIGIR Conference on Human Information Interaction and Retrieval (CHIIR '22), March 14–18, 2022, Regensburg, Germany*. ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3498366.3505828

*Work done during an internship at Amazon.

CHIIR '22, March 14–18, 2022, Regensburg, Germany

ACM ISBN 978-1-4503-9186-3/22/03.
https://doi.org/10.1145/3498366.3505828

1 INTRODUCTION

Explanations describe machine learning model's output in a human understandable manner. They are becoming increasingly important as black-box machine learning models—where the processes between input and output are opaque—have a direct impact in society, as seen by a number of workshops and conferences related to explainability and transparency[1]. In the information retrieval domain, explanations can serve different goals, for example in a recommendation scenario it can improve trust, persuade users to buy and increase the transparency of the system [22].

A number of methods have been proposed to explain the output of a model for both recommendation [4, 7, 11, 13, 14, 20, 22] and search [1, 5, 17, 18, 21, 23], which typically use one of the three following sources of information: attributes, reviews and relevance information [25].

The voice channel is a natural way to interact with assistants and it is becoming ubiquitously available through multiple devices. In voice-based shopping customers who engage with devices without a screen cannot see product images and are exposed to fewer products and less information in general. Unlike web product search, users of voice assistants typically do not go beyond two product recommendations, issue queries differently, buy products which are less expensive and in a higher ratio for certain categories such as groceries and do less navigational actions [8].

While providing explanations for product recommendations seems appropriate for addressing challenges of voice product search, little attention has been given to this topic. It is still unknown how to provide explanations for voice product search and their effect in user behavior is unclear.

In order to address this gap, we propose here a framework to generate review-based explanations for voice product search, and propose a POINTWISE variant that automatically selects a helpful review sentence for the recommended product to disclose and a PAIRWISE variant that first selects a diverse pair of products and then discloses a review sentence for each product of the pair. We compare the proposed methods with a BASELINE that mimics the response of Alexa [12]. Unlike previous work on comparative explanations for recommender systems [10] that selects reference products to compare against previous user purchases, we focus on the cold-start scenario where we only have access to a query.

An example of explanation obtained from each method can be seen in Table 1 for the query '*lemon cake decorations*'. A user that receives the PAIRWISE explanation can gather that while the first product is cheap, cute and sturdy, the second product is suited for a bridal shower themed party. Unlike the BASELINE and POINTWISE explanations, the pair of products provided as part of the PAIRWISE

[1]For example https://facctconference.org/index.html and https://human-centered.ai/explainable-ai/.

Table 1: Examples for the query '*lemon cake decorations*' generated by different explanation methods. In bold are the review sentences selected for the product(s).

Method	Example
BASELINE	"*I found LILIPARTY 24 Pcs Glitter Lemon Cupcake Toppers Fruits Theme Party, Lemonade Party Decor. It's 9.99. With delivery in 3 days*"
POINTWISE	"*I found LILIPARTY 24 Pcs Glitter Lemon Cupcake Toppers Fruits Theme Party, Lemonade Party Decor. It's 9.99. With delivery in 3 days.* ***A reviewer said that 'They were really cute and sturdy and added just the right touch'.***"
PAIRWISE	"*I found two products. The first is LILIPARTY 24 Pcs Glitter Lemon Cupcake Toppers Fruits Theme Party, Lemonade Party Decor. It's 9.99 . With delivery in 3 days. The second is Lemon Bridal Shower Party Decoration Set. It's 16.99 . With delivery in 3 days.* ***A reviewer said the following about the first product 'They were really cute and sturdy and added just the right touch', while the second product received the following comment 'These were the perfect toppers for our key lime pie shooters for a bridal shower theme.***"

method give a reference point for the user to decide and compare upon, and allows them to make a more informed decision.

Our crowd-sourced evaluation with real queries from a large e-commerce search system reveal that PAIRWISE explanations improves with statistically significance over both the BASELINE and the POINTWISE on the *Effectiveness* and *Transparency* goals—*helping users make good decisions* and *explaining how the system works* respectively. Moreover, we see that the gains provided by the PAIRWISE method are consistent with different subsets of data, based on the diversity of the selected pairs, average product price associated with the query and the ambiguity of the query.

2 METHOD

In this section we first describe the baseline, followed by the review-based pointwise explanations before defining our pipeline for generating review-based pairwise explanations. In order to generate explanations we rely on filling template-based structures, for which we describe in Backus-Naur Form [9].

2.1 Baseline explanation (BASELINE)

The baseline explanation is based upon Alexa's [12] existing large-scale voice product search system. The recommended product is the first result a ranker[2] provides for a query. The information disclosed about the recommended product is the product title, the price and delivery date. The template is as follows:

[2]Throughout this paper, we treat the product ranker as a black-box and the sole assumption we make is that the top results are relevant with respect to a given query.

⟨*baseline*⟩ ::= I found ⟨*product_title*⟩. It's ⟨*price*⟩ dollars. With delivery in ⟨*delivery*⟩ days.

2.2 Pointwise explanation (POINTWISE)

The majority of explanation methods in recommender systems extract information from reviews. Here we extract a ⟨*review_sentence*⟩ from the review of the recommended product that received the highest number of helpful votes and use it to fill in the following template:

⟨*pointwise_explanation*⟩ ::= ⟨*baseline*⟩. A reviewer said that ⟨*review_sentence*⟩.

In order to extract the most helpful sentence out of the review with the most helpful votes we use a regression-based BERT model that receives as input a review sentence and outputs a helpfulness score. Formally, we fine-tune BERT on a corpus of review sentences and their respective helpfulness scores. We make predictions as follows: $helpfBERT(s) = FFN(BERT_{CLS}(s))$, where s is the review sentence, $BERT_{CLS}$ is the pooling operation that extracts the representation of the [*CLS*] token from the last layer and FFN is a feed-forward network that outputs one logit with the predicted helpfulness score of the sentence. We use the mean-square loss for training. The selection of the review sentence is then $s_p^* = argmax_{s \in topReview(p)} helpfBERT(s)$, where $topReview(p)$ is the review with the highest count of '*found this review helpful*' for the product p, and the template is filled by setting ⟨*review_sentence*⟩ as s_p^*.

2.3 Pairwise explanation (PAIRWISE)

The pipeline to generate review-based pairwise explanations is shown in Figure 1. Based on the list $\mathcal{R}_q$ with k products for the query, we select a pair of products π_A, π_B, select one sentence from a review for each product of the pair and finally fill the following template with the selected sentences:

⟨*pairwise_explanation*⟩::=⟨*products_pair*⟩. A reviewer said the following about the first product⟨*review_sentence_A*⟩ while the second product received the following comment⟨*review_sentence_B*⟩.

⟨*products_pair*⟩::= I found two products. The first is ⟨*product_title_A*⟩. It's ⟨*price_A*⟩. With delivery in ⟨*delivery_A*⟩ days. The second product is⟨*product_title_B*⟩. It's ⟨*price_B*⟩. With delivery in ⟨*delivery_B*⟩ days.

In order to select products from the ranked list provided by the ranker, we rely on the following greedy algorithm. Formally, the first product π_A is the top-ranked product for the query: $\pi_A = \pi_1$. The second product is the one from the remaining products of the top-k ranked list with maximum f: $\pi_B \leftarrow argmax_{p \in \mathcal{R}_q \setminus \pi_A} f(p, \pi_A)$.

The function f is a linear combination of the helpfulness of the most helpful sentence from the product p with the novelty of

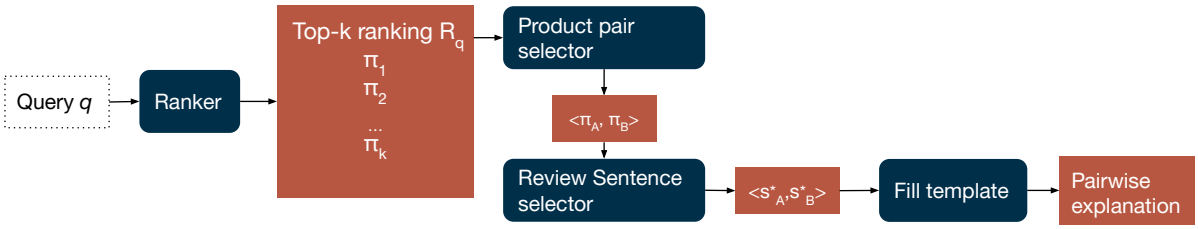

Figure 1: Pipeline for generating PAIRWISE explanations for voice product search.

product p with respect to first product π_A as estimated by negative cosine similarity between their review sentences. Formally, it can be written as:

$$f(p, \pi_A) = \alpha * helpfBERT(s_p^*) - (1 - \alpha) * CosSim(sBERT(s_p^*), sBERT(\pi_A)) \quad (1)$$

In order to obtain a representation of the sentences to calculate their similarity we use sentenceBERT (*sBERT*) [16]. After choosing the pair π_A, π_B, we use the same sentence selection process described for the POINTWISE method, applied for both products leading to s_A^*, used to fill the template in ⟨*review_sentence_A*⟩, and s_B^* used to fill the template in ⟨*review_sentence_B*⟩.

3 EXPERIMENTAL SETUP

In this section we first describe the datasets and models used to generate and evaluate the explanations, followed by how we evaluate the quality of the explanations generated by each method.

3.1 Datasets and models

The dataset used to generate explanations consists of a set of queries Q, where $|Q| = 274$, issued on a widely used e-commerce product search system and their respective ranked lists $\mathcal{R}_q$. The queries were selected randomly from the set of all queries issued in the year of 2021. For each $q \in Q$ we have the set of ranked documents that the search system in production showed to the user $\mathcal{R}_q$, for which we limit it to the top-10 ranked products, $|\mathcal{R}_q| = 10$.

The information of products, e.g. reviews, price, that appear at least once in a ranked list R_q are extracted from their respective product web pages, except for the title, for which we use shorter voice-friendly versions as used by Alexa.

In order to train the *helpfBERT* model used at § 2.2 and 2.3 we resort to the publicly available dataset provided by [6] which contains a total of 20.000 training instances: review sentences and their respective helpfulness scores. After fine-tuning *helpfBERT* for one epoch with a batch size of 14, we obtain a mean squared error of 0.048 on the test set, which is in line with their results. We use the HuggingFace library [24] implementation and resort to their default hyperparameters, usng the pre-trained `bert-base-cased` release. Before applying the *helpfBERT* model on review sentences (for both POINTWISE and PAIRWISE methods), we use only reviews with either 4 or 5 stars and filter sentences with negative sentiment using a distilled BERT model, fine-tuned for sentiment analysis on the SST-2 dataset[3].

[3] https://huggingface.co/distilbert-base-uncased-finetuned-sst-2-english

In order to represent the review sentences (*sBERT*) and calculate their similarity for the PAIRWISE pair selection method, we use the fine-tuned `paraphrase-mpnet-base-v2` model from [16], which has the highest quality overall[4]. The α parameter is set to 0.5 and we apply min-max scaling to the outputs of *helpfBERT* before combining it with the novelty at Equation 1.

3.2 Crowd-sourced evaluation

In order to evaluate the quality of the generated explanations we rely on the human evaluation through post-task questionnaires, which is the most commonly used methodology to evaluate explanations [15]. Each crowd-worker received a query, the respective explanation for the query and was asked to answer a set of questions regarding the quality of the explanations.

The specific questions asked in the post-questionnaire were adapted from [2] to reflect a product search scenario instead of a movie recommendation one, and are described in Table 2. Each question was rated on a 4-point Likert scale: (1) not at all, (2) moderately, (3) slightly and (4) a great deal. We asked questions for the following goals: *Persuasiveness* (convince users to buy), *Effectiveness* (help users make good decisions), *Transparency* (explain how the system works) and *Scrutability* (allow users to tell the system is wrong). We chose such goals as Balog and Radlinski [2] found them to be highly correlated with the remaining goals defined by [22]. We added *Persuasiveness* to the set of three goals recommended by [2] due to the fact that convincing to buy is one of the most important goals in a product search system.

We filtered workers who did not answer correctly the test question regarding the number of products being recommended (one for BASELINE and POINTWISE vs two for PAIRWISE). Additionally, we filtered workers who did not spend at least 1 minute answering each question. We hired level 2 workers which are a smaller group of more experienced, higher accuracy contributors of the Appen platform[5]. Each worker contributed to a maximum of 20 different judgements and each of the queries and explanation method were judged 3 times, leading to a total of 2446 unique judgements. In our analysis, we take the average value of the 3 judgements for an explanation method and goal combination. We calculate the agreement between annotators for each query and average the agreements over all queries using the Pearson Correlation in the same manner as [19] and obtain a satisfactory correlation of 0.55 for our subjective task, which are comparable for example with the correlations observed by [6] for their review helpfulness task.

[4] Amongst models from https://www.sbert.net/docs/pretrained_models.html.
[5] https://appen.com/

Table 2: Questions asked for each of the explanation goals.

Goal	Definition [22]	Question asked, adapted from [2]
Persuasiveness	Convince users to buy	The response makes me want to buy one of the recommended products.
Effectiveness	Help users make a good decision	The response helps me determine how well I will like this product.
Transparency	Explain how the system works	The response helps me understand what the product recommendation is based on.
Scrutability	Allow users to tell the system is wrong	The response allows me to understand if the system made an error in interpreting my request.

Table 3: Quality of the explanation methods for different explanations goals (mean and confidence intervals). Bold values indicate the best value for each goal and superscripts † and ‡ indicate statistical significant improvements over the `BASELINE` and `POINTWISE` respectively using Student's t-test with confidence level of .95.

	Persuasiveness	*Effectiveness*	*Transparency*	*Scrutability*
`BASELINE`	3.43 ± .07	3.37 ± .07	3.34 ± .07	2.34 ± .10
`POINTWISE`	3.48 ± .06	3.48 ± .06†	3.46 ± .06†	**2.37 ± .10**
`PAIRWISE`	**3.56 ± .06**†	**3.57 ± .06**†‡	**3.56 ± .06**†‡	2.34 ± .10

4 RESULTS

In this section we first discuss the quality of the explanations generated by the proposed method followed by an analysis of errors per different types of queries.

4.1 Explanation methods

In order to evaluate the quality of the explanations of the proposed method, we compare the average scores received by each method across queries for each of the explanation goals. The results are displayed in Table 3.

First, we find that the `POINTWISE` method that takes into account review sentences for providing explanations leads to statistically significant improvement over the `BASELINE` explanation that does not use review sentences for both *Effectiveness* and *Transparency*. This shows that using helpful review sentences has a positive effect on helping decision making in voice product search, and allows users to feel like they understand how the system works even though the review text might not necessarily be used to rank products by the underlying search system.

For the `PAIRWISE` method we observe that it outperforms the `BASELINE` method for *Persuasiveness*, *Effectiveness* and *Transparency*, and the `POINTWISE` method for *Effectiveness* and *Transparency*. This indicates that our proposed method which selects pairs and provides a reference product is more persuasive, more helpful in the decision making process of customers and also leads to a perception of better understanding of how the underlying system works.

We found that no method had a significant improvement for the *Scrutability* goal. We hypothesize that a method that identifies the search intent words that were used by the ranker such as [18] might be preferred over review-based methods for this specific goal.

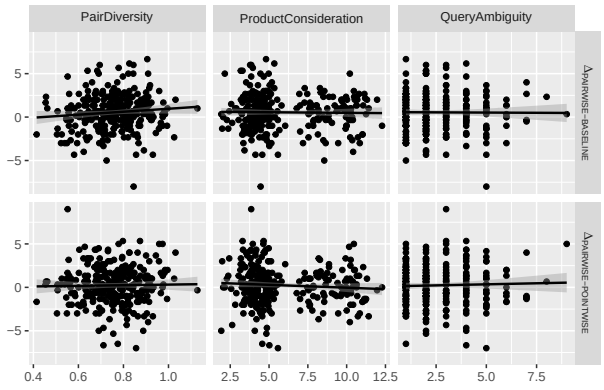

Figure 2: Difference between `PAIRWISE` and other methods ($\Delta_{\texttt{PAIRWISE−BASELINE}}$ and $\Delta_{\texttt{PAIRWISE−POINTWISE}}$) in terms of the sum of the goals (*Persuasiveness*, *Effectiveness*, *Transparency*, *Scrutability*) and its correlation with `PairDiversity`, `ProductConsideration` and `QueryAmbiguity`.

4.2 Error analysis

In order to understand when `PAIRWISE` performs better than the `POINTWISE` and `BASELINE` methods we check how three features (`PairDiversity`, `ProductConsideration` and `QueryAmbiguity`) correlate with the gains provided by the proposed method.

`PairDiversity` is the distance between the selected products' representations ($1 - CosSim(sBERT(s^*_A), sBERT(s^*_B))$).

`ProductConsideration` is the log of the average price of the top-10 products retrieved for the query ($\log_2 \frac{\sum price_{p\ p\in\mathcal{R}_q}}{|\mathcal{R}_q|}$).

`QueryAmbiguity` is the number of different product categories, e.g. Laptop Accessories, in the ranked list for the query $|C_q|$.

We plot the results on Figure 2, where each column is one of the three features represented by their x-axis, and the y-axis is the difference in the sum of goals ($\Delta_{\texttt{PAIRWISE−BASELINE}}$ and $\Delta_{\texttt{PAIRWISE−POINTWISE}}$). We see that the regression slopes are not steep, and the Pearson correlation between the gains provided by the `PAIRWISE` method and the variables analyzed are small (<0.10). This indicates that the method is not particularly more prone to errors when (I) the diversity is small (small `PairDiversity`), (II) the product is cheaper (small `ProductConsideration`), and (III) when the query is more specific (small `QueryAmbiguity`).

We also find that `PAIRWISE` outperforms both `BASELINE` and `POINTWISE` if we split the data according to such features into two equal sized buckets, e.g. `PairDiversity`<0.8 and `PairDiversity`>=0.8. The results indicate that the gains of the proposed method are consistent across different levels of pair diversity, price and query ambiguity.

5 CONCLUSION

In this paper we propose a pipeline to generate review-based explanations for voice product search. We compare two variants, a

POINTWISE explanation that recommends one product and selects a helpful sentence to explain the recommendation and a PAIRWISE explanation that recommends two diverse products and selects two helpful sentences to explain the recommendations.

Our results reveal that PAIRWISE explanations are significantly better at achieving the goals of *Effectiveness* and *Transparency* when compared to the BASELINE and POINTWISE explanations, and the performance gains hold across different subsets of data based on the diversity of the pairs, the products average product price associated with the query and the query ambiguity.

Directions for future work include automatically identifying when each explanation method is better for a given query and methods that optimize for *Scrutability*.

REFERENCES

[1] Qingyao Ai, Yongfeng Zhang, Keping Bi, and W Bruce Croft. 2019. Explainable product search with a dynamic relation embedding model. *ACM Transactions on Information Systems (TOIS)* 38, 1 (2019), 1–29.

[2] Krisztian Balog and Filip Radlinski. 2020. Measuring recommendation explanation quality: The conflicting goals of explanations. In *Proceedings of the 43rd International ACM Sigir Conference on Research and Development in Information Retrieval*. 329–338.

[3] David Carmel, Elad Haramaty, Arnon Lazerson, Liane Lewin-Eytan, and Yoelle Maarek. 2020. Why Do People Buy Seemingly Irrelevant Items in Voice Product Search. In *Proceedings of the Third ACM International Conference on Web Search and Data Mining (WSDM'20). ACM.*

[4] Zhongxia Chen, Xiting Wang, Xing Xie, Tong Wu, Guoqing Bu, Yining Wang, and Enhong Chen. 2019. Co-Attentive Multi-Task Learning for Explainable Recommendation.. In *IJCAI*. 2137–2143.

[5] 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*. 1005–1008.

[6] Iftah Gamzu, Hila Gonen, Gilad Kutiel, Ran Levy, and Eugene Agichtein. 2021. Identifying Helpful Sentences in Product Reviews. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*. Association for Computational Linguistics, 678–691. https://doi.org/10.18653/v1/2021.naacl-main.55

[7] Diana C Hernandez-Bocanegra, Tim Donkers, and Jürgen Ziegler. 2020. Effects of argumentative explanation types on the perception of review-based recommendations. In *Adjunct Publication of the 28th ACM Conference on User Modeling, Adaptation and Personalization*. 219–225.

[8] Amir Ingber, Arnon Lazerson, Liane Lewin-Eytan, Alexander Libov, and Eliyahu Osherovich. 2018. The challenges of moving from web to voice in product search. In *Proc. 1st International Workshop on Generalization in Information Retrieval (GLARE 2018). http://glare2018. dei. unipd. it/paper/glare2018-paper5. pdf.*

[9] Donald E Knuth. 1964. Backus normal form vs. backus naur form. *Commun. ACM* 7, 12 (1964), 735–736.

[10] Trung-Hoang Le and Hady W Lauw. 2021. Explainable Recommendation with Comparative Constraints on Product Aspects. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*. 967–975.

[11] Lei Li, Yongfeng Zhang, and Li Chen. 2020. Generate neural template explanations for recommendation. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 755–764.

[12] Yoelle Maarek. 2019. Alexa, Can You Help Me Shop?. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval* (Paris, France) *(SIGIR'19)*. Association for Computing Machinery, New York, NY, USA, 1369–1370. https://doi.org/10.1145/3331184.3331443

[13] Cataldo Musto, Pasquale Lops, Marco de Gemmis, and Giovanni Semeraro. 2019. Justifying recommendations through aspect-based sentiment analysis of users reviews. In *Proceedings of the 27th ACM Conference on User Modeling, Adaptation and Personalization*. 4–12.

[14] Jianmo Ni, Jiacheng Li, and Julian McAuley. 2019. Justifying Recommendations using Distantly-Labeled Reviews and Fine-Grained Aspects. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Association for Computational Linguistics, Hong Kong, China, 188–197. https://doi.org/10.18653/v1/D19-1018

[15] Ingrid Nunes and Dietmar Jannach. 2017. A systematic review and taxonomy of explanations in decision support and recommender systems. *User Modeling and User-Adapted Interaction* 27, 3 (2017), 393–444.

[16] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Association for Computational Linguistics, Hong Kong, China, 3982–3992. https://doi.org/10.18653/v1/D19-1410

[17] 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*. 770–773.

[18] Jaspreet Singh and Avishek Anand. 2020. Model agnostic interpretability of rankers via intent modelling. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*. 618–628.

[19] Rion Snow, Brendan O'connor, Dan Jurafsky, and Andrew Y Ng. 2008. Cheap and fast–but is it good? evaluating non-expert annotations for natural language tasks. In *Proceedings of the 2008 conference on empirical methods in natural language processing*. 254–263.

[20] Panagiotis Symeonidis. 2008. Justified recommendations based on content and rating data. In *WebKDD Workshop on Web Mining and Web Usage Analysis*. ACM.

[21] 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).

[22] Nava Tintarev and Judith Masthoff. 2015. Explaining recommendations: Design and evaluation. In *Recommender systems handbook*. Springer, 353–382.

[23] Manisha Verma and Debasis Ganguly. 2019. LIRME: locally interpretable ranking model explanation. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1281–1284.

[24] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771* (2019).

[25] Yongfeng Zhang and Xu Chen. 2020. .