Improving Relevance Quality in Product Search using High-Precision Query-Product Semantic Similarity

Alireza Bagheri Garakani, Fan Yang, Wen-Yu Hua, Yetian Chen,
Michinari Momma, Jingyuan Deng, Yan Gao, Yi Sun
Amazon
Seattle, WA, USA
{alirezg,fnam,wenyuhua,yetichen,
michi,jingyua,yanngao,yisun}@amazon.com

Abstract

Ensuring relevance quality in product search is a critical task as it impacts the customer's ability to find intended products in the short-term as well as the general perception and trust of the e-commerce system in the long term. In this work we leverage a high-precision cross-encoder BERT model for semantic similarity between customer query and products and survey its effectiveness for three ranking applications where offline-generated scores could be used: (1) as an offline metric for estimating relevance quality impact, (2) as a re-ranking feature covering head/torso queries, and (3) as a training objective for optimization. We present results on effectiveness of this strategy for the large e-commerce setting, which has general applicability for choice of other high-precision models and tasks in ranking.

1 Introduction

Search is one of the primary means used by customers to find products in e-commerce and therefore it is critical to ensure the relevance quality of search results. A search result may be considered to have low relevance quality (search defect) if it mismatches the customer's query intent. Such defects may range from the mild case of mismatch in brand or color (i.e., substitutes) to the more egregious case of a completely irrelevant result of a different product type. Addressing search defects is a critical task as it can damage customer trust and perception of the e-commerce system, and in general hinder the ability to sell products.

As a simplified view, product search may consist of two distinct phases. First, given a search query, a set of candidate products are determined based on various matchset generation techniques (e.g. lexical/semantic matching, historical associations from past query reformulations, and others). Next, a ranking model is used to generate a score for each (query,product) pair upon which a descending sort determines a ranked list. In the ideal case, the construction of the matchset would be strictly restricted to products that are only relevant to the customer's query, however, this can be challenging to enforce without potentially limiting recall, which itself presents issues that negatively impact search experience. In practice, products in the matchset may still contain complementary or related items to the customer-intended one due to partial matches or from noisy historical associations.

The ranking phase can be used to mitigate the impact of search defects that may exist in the matchset by demoting such results out of the first several pages. In contrast to matchset restrictions, demotion in ranking can be seen as a softer approach since a product will not be entirely eliminated from search results but simply moved beyond the top-results. Given a dataset with relevance labels for query-product pairs, relevance quality can be optimized within ranking to demote products estimated to be less relevant to the customer's query. However, this strategy will have a dependence on similarity measures that often need to favor online efficiency over a higher-precision computationally-expensive counterpart. For example, common ranking features may include query-product lexical/semantic match features and behavioral features that incorporate historical customer interactions; for all these cases, efficient computation over the entire matchset will be required, whether by simple online computations (e.g. TF-IDF, cosine similarity) or by retrieving pre-computed offline (intermediate) results or a combination of both.

In this work we leverage a high-precision model bounded by offline computational resources for addressing our ranking-based task. Specifically, we develop a high-precision cross-encoder BERT model for semantic similarity between customer query and products that is optimized for predicting relevance quality and we survey its effectiveness for three applications where offline-generated scores could be used: (1) as an offline metric for estimat-

Proceedings of The Fifth Workshop on e-Commerce and NLP (ECNLP 5), pages 44 - 48

ing relevance quality impact, (2) as a re-ranking feature covering head/torso queries, and (3) as a training objective for optimization. We present results on effectiveness of this strategy for the large e-commerce setting, which has general applicability for choice of other high-precision models (i.e. other than BERT) and tasks in ranking (other than relevance quality).

2 Related Work

Generating textual representations that are effective for downstream tasks is an active area of research that has seen significant improvement in the last several years (Peters et al., 2018; Cer et al., 2018; Devlin et al., 2019). BERT (Devlin et al., 2019) is one such model based on a multi-layer bidirectional Transformer encoder (Vaswani et al., 2017) that has shown state-of-the-art performance on various NLP tasks. In this work, we leverage BERT for two-sentence classification with cross-encoders that is known to have strong predictive performance while at the same time presenting computational challenges for large-scale product search (Humeau et al., 2020; Reimers and Gurevych, 2019). Various strategies can be used towards improving scalability of model inference such as model distillation/compression (Hinton et al., 2015; Sanh et al., 2019; Bucila et al., 2006; Liu et al., 2021; Gordon et al., 2020) and factorizing inputs into separate model paths (i.e. two-tower or bi-encoder models) (Huang et al., 2013; Reimers and Gurevych, 2019; Humeau et al., 2020), but these optimizations typically come at the expense of model performance. For example, using a factorized model enables opportunities to cache pre-computed embeddings and use efficient distance functions on embeddings vectors (e.g. cosine similarity), but this architecture will sacrifice interactions between inputs within early model layers that tend to be helpful. Indeed, the trade-off between precision and inference speed is not limited to BERT nor transformer-based models; in general, higher precision for a given task may be achievable when there is flexibility to use more complex models (e.g. more parameters, ensemble methods), and consume more costly features (e.g. embeddings generated from a secondary model and fed as input). Given its applicability and known computational challenges for our task, in this work we use BERT directly as our high-precision model (i.e. without any computational efficiency optimizations) and apply it for several applications where offline-computed scores are permitted.

3 Query-Product Semantic Similarity

For our task we would like to develop a measure of semantic similarity $g : (Q, A) \rightarrow \mathbb{R}$, given an arbitrary query $q \in Q$ and product $a \in A$, where we assume the cardinality of Q and A may be infinite. We select g by favoring a model that maximizes predictive performance bounded by offline resources as opposed to meeting stricter online inference requirements. As mentioned in the previous section, we adopt BERT and frame our problem as two-sentence classification (Devlin et al., 2019) to incorporate textual inputs for the query and product.

For two-sentence classification using BERT as a cross-encoder the input sequence is prepared by prefixing a 'CLS' token followed by the query textual representation, a special separator token ('SEP'), and finally the product textual representation via product title. This input sequence is segmented into sub-words using WordPiece algorithm (Wu et al., 2016) and fed through BERT-base pre-trained model (12 transformer blocks, 768 hidden units, and 12 self-attention heads). We further pre-train the BERT-base model using both Masked LM and Next Sentence Prediction tasks as described in (Devlin et al., 2019) on product metadata such as title and description. For building a classification model, the output embedding for 'CLS' token is passed to a final linear classification layer. All model weights, including transformer block layers, are trained jointly using binary cross-entropy loss. The labeled dataset consisting of judgments tuples (query,product,label) is based on historical query-product samples with relevance judgments (relevant vs irrelevant), which is split into train/validation/test datasets. Table 1 shows a few examples of query-product pairs with their respective model score. Further model improvements (e.g. use of pairwise loss for finetuning, extension beyond BERT to include product images) are applicable for our problem setting but left as future work.

For evaluating relevance quality, we use NDCG metric (Normalized Discount Cumulative Gain) that achieves the highest value of 1 when the rank order respects the ideal relevance label ordering, which is then averaged over all queries. Table 2 shows the performance of our BERT-based classification model benchmarked against a competitive

Table 1: Example query and product inputs with respective BERT-based predictions with higher scores indicating stronger relevance.

Search Query	Product Title (truncated)	Label	Score
blankets for winter double bed soft	... Microfiber Single Comforter ...	exact	0.911
kitchen small storage boxes	... Plastics Polka Container Set ...	exact	0.967
girl jacket for winter	... Women's Slim Fit Joggers ...	irrelevant	0.011

Table 2: Relevance quality (NDCG@16) over competitive GBDT baseline model, including evaluation over query-frequency segments. Table includes BERT model without additional pre-training (sem-noPT), after pre-training (sem), and added as additional feature on top of GBDT baseline.

	Relevance Quality		
Model	Overall	Head+Torso	Tail
sem-noPT	-0.25%	-0.60%	0.18%
sem	-0.13%	-0.59%	0.44%
sem+GBDT	**0.75%**	**0.64%**	**0.89%**

model based on Gradient Boosted Decision Tree (GBDT) (Friedman, 2000) using existing search ranking features (lexical, semantic, and behavioral based) that are either computed in real-time or as part of an offline build. We observe that the BERT-based predictor after pre-training (sem) has significant improvement in the tail-query segment ($+0.44\%$) but sub-par for head+torso queries (-0.59%); this is expected as behavioral features as part of the GBDT baseline model tend to perform well for frequent traffic segments where historical customer behavior signals are available but will be sparse or noisy otherwise. Finally, we build a GBDT-based predictor in a similar manner as the baseline except including our BERT model score as an additional input feature used for feature selection. This final model demonstrates the value brought over the existing set of online-efficient features ($+0.75\%$ overall improvement), where gains are seen even for the head-torso query segment suggesting the new feature works in a complimentary way with existing features.

Using this high-precision predictor, we explore several applications in search ranking where we can leverage this model using offline-generated scores for our end task. These are discussed in the next section.

Table 3: Measure of correlation metrics between offline and online measurements for relevance improvement, where offline estimator is baseline or our high-precision predictor.

estimator	Pearson	Kendall
baseline	0.58029	0.20589
sem+GBDT	0.83764	0.59298

4 Applications within Ranking

4.1 Offline Estimation of Relevance Quality

Given that relevance quality is an important metric across search, it is useful to monitor it alongside other important metrics (e.g. revenue, latency) for each experiment that impacts ranking of products and search experience. Here we propose using our semantic predictor for estimating online relevance impact and seek to measure which offline estimator (our high-precision predictor or baseline) better reflects the changes observed for the online metric. We represent our observations as pairs (x_i, y_i) where, for of a given model (treatment) over production (control), x_i is the estimated improvement given an estimator and offline dataset, and y_i is the actual observed improvement as measured by human-judged labels. For our study, our dataset consists of 19 pairs collected from 5 experiments within a 6-month span impacting ranking for a particular marketplace. Our offline estimate is measured on a ranking evaluation dataset of query-product pairs by measuring the improvement in exact probability (output of our semantic predictor) among top results based on the treatment's rank over the control's rank, averaged over all queries. Similar estimation is done for our baseline model, which is a GBDT model trained similar to baseline in section 3. Using this dataset of offline-online impact pairs, we measure Pearson and Kendall correlation coefficients. Table 3 indicates that while neither estimator perfectly reflects online observed values, our high-precision semantic predictor shows significantly improvement over our baseline and, hence, can be used for more effective model selec-

Table 4: Online relevance quality (NDCG@16) of re-ranking models using high-precision semantic model (sem+GBDT) and refreshed baseline over production model.

Model	Relevance Quality
baseline	-0.15% (p=0.39)
sem+GBDT	0.46% (p=0.017)

tion and metric monitoring.

4.2 Feature for Search Re-ranking

We explore using our high-precision model as an input feature for search re-ranking. Re-ranking is a second ranking phase on the top-K results from the preceding (main) ranking phase. Given that our semantic predictor cannot be trivially applied online for the entire matchset due to computational and latency costs, we instead pre-computed semantic scores for more frequent queries and their respective top results and index these scores for fast online retrieval. Specifically, coverage is limited to queries having a predefined number of searches S within the last D days and for their top-P ranked results. For our experiment we re-rank top-16 results where feature coverage exists for head/torso queries (influenced by S and D parameters) and their respective products (by selecting $P >= 16$), however, feature will lack coverage for infrequent queries.

We prepare GBDT predictor models similar to section 3, except the semantic feature used within 'sem-GBDT' is modified to reflect the expected feature coverage online. Results are shown in Table 4, where we observe from online results that we are able to significantly improve NDCG by 0.46% while avoiding regression to revenue, latency, or other guardrails (not shown).

4.3 Objective for Optimization

In this section we prepare primary-phase ranking models and introduce an objective for optimization of search quality. We take the optimization-based approach in ranking, as opposed to feature-based as used for re-ranking usecase, given that offline score pre-computation is no longer practical as it would need to cover the entire matchset (or at least a sizable portion) to be useful. Instead, by using an optimization-based approach we can use full-coverage estimates using our high-precision predictor at training time. Tail queries (e.g. a query never seen before) were strictly not covered in the feature-based re-ranking usecase, but the optimization route can be useful even for this segment by allowing the model to learn associations between other existing ranking features and the task-specific labels generated via our high-precision predictor.

Table 5: Online relevance quality (NDCG@16) of multi-objective ranking model using high-precision semantic predictor as an objective over comparable baseline. Results include query frequency segments. All measurements are statistically significant ($p < 0.05$).

	Relevance Quality		
Exp.	Overall	Head+Torso	Tail
1	0.29%	0.19%	0.44%
2	0.49%	0.52%	0.42%

We use the constraint-based optimization algorithm, AL-LambdaMART (Momma et al., 2020), for the multi-objective formulation:

$$\min_s C^p(s) \quad s.t. \; C^s(s) \leq b \tag{1}$$

where the cost terms are NDCG-weighted pairwise loss as similarly defined in LambdaMART (Burges, 2010)). In our problem we assume two objectives – relevance quality and revenue – and our modeling goal is to maximize relevance quality $[\min_s C^p(s)]$ while remaining at least flat on revenue relative to the existing production model $[C^s(s) \leq b]$. For the latter, the upperbound value b is set accordingly to achieve this goal using the approach outlined in (Momma et al., 2020). The motivation for having both objectives, despite being generally aligned, is that relevance quality may be one of several factors important for a customer's shopping mission.

We ran 2 online experiments for a particular marketplace against the existing production model. To isolate impact, each experiment included a comparable treatment optimized similarly but without the high-precision semantic score. Results in Table 5 show that in each experiment our semantic treatment is able to achieve higher online relevance impact over the baseline ranking model, while keeping flat on revenue (not shown). Each experiment also showed improvements across all query segments, including tail queries.

5 Conclusion and Future Work

To improve relevance quality in search ranking we applied a high-precision BERT cross-encoder model for semantic similarity in search. We demonstrated three applications where offline-generated

scores can be leveraged to improve the end task. This can be viewed as a complementary approach alongside efforts for developing online-efficient models, where the advantages include leveraging higher-precision models and with potentially less development overhead. A follow-up study including benchmarking on public datasets is left as future work.

References

Cristian Bucila, R. Caruana, and Alexandru Niculescu-Mizil. 2006. Model compression. In *KDD '06*.

Chris J.C. Burges. 2010. From ranknet to lambdarank to lambdamart: An overview. Technical Report MSR-TR-2010-82.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Brian Strope, and Ray Kurzweil. 2018. Universal sentence encoder for English. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 169–174, Brussels, Belgium. Association for Computational Linguistics.

J. Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*.

Jerome H. Friedman. 2000. Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 29:1189–1232.

Mitchell Gordon, Kevin Duh, and Nicholas Andrews. 2020. Compressing bert: Studying the effects of weight pruning on transfer learning. In *Proceedings of the 5th Workshop on Representation Learning for NLP*, pages 143–155, Online. Association for Computational Linguistics.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network.

Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. 2013. Learning deep structured semantic models for web search using clickthrough data. ACM International Conference on Information and Knowledge Management (CIKM).

Samuel Humeau, Kurt Shuster, Marie-Anne Lachaux, and Jason Weston. 2020. Poly-encoders: Architectures and pre-training strategies for fast and accurate multi-sentence scoring. In *International Conference on Learning Representations*.

Peiyang Liu, Xi Wang, Lin Wang, Wei Ye, Xiangyu Xi, and Shikun Zhang. 2021. *Distilling Knowledge from BERT into Simple Fully Connected Neural Networks for Efficient Vertical Retrieval*, page 3965–3975. Association for Computing Machinery, New York, NY, USA.

Michinari Momma, Alireza Bagheri Garakani, Nanxun Ma, and Yi Sun. 2020. *Multi-Objective Ranking via Constrained Optimization*, page 111–112. Association for Computing Machinery, New York, NY, USA.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

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)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. Neural Information Processing Systems.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *CoRR*, abs/1609.08144.