Multi-Objective Ranking Optimization for Product Search Using Stochastic Label Aggregation

David Carmel
Amazon
Haifa, Israel
dacarmel@amazon.com

Elad Haramaty
Amazon
Haifa, Israel
eladh@amazon.com

Arnon Lazerson
Amazon
Haifa, Israel
arnonl@amazon.com

Liane Lewin-Eytan
Amazon
Haifa, Israel
lliane@amazon.com

ABSTRACT

Learning a ranking model in product search involves satisfying many requirements such as maximizing the relevance of retrieved products with respect to the user query, as well as maximizing the purchase likelihood of these products. Multi-Objective Ranking Optimization (MORO) is the task of learning a ranking model from training examples while optimizing multiple objectives simultaneously. Label aggregation is a popular solution approach for multi-objective optimization, which reduces the problem into a single objective optimization problem, by aggregating the multiple labels of the training examples, related to the different objectives, to a single label. In this work we explore several label aggregation methods for MORO in product search. We propose a novel stochastic label aggregation method which randomly selects a label per training example according to a given distribution over the labels. We provide a theoretical proof showing that stochastic label aggregation is superior to alternative aggregation approaches, in the sense that any optimal solution of the MORO problem can be generated by a proper parameter setting of the stochastic aggregation process. We experiment on three different datasets: two from the voice product search domain, and one publicly available dataset from the Web product search domain. We demonstrate empirically over these three datasets that MORO with stochastic label aggregation provides a family of ranking models that fully dominates the set of MORO models built using deterministic label aggregation.

CCS CONCEPTS

• **Information systems → Learning to rank**.

KEYWORDS

product search, multi-objective ranking optimization, stochastic label aggregation

ACM Reference Format:
David Carmel, Elad Haramaty, Arnon Lazerson, and Liane Lewin-Eytan. 2020. Multi-Objective Ranking Optimization for Product Search Using Stochastic Label Aggregation. In *Proceedings of The Web Conference 2020 (WWW '20), April 20–24, 2020, Taipei, Taiwan.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3366423.3380122

WWW '20, April 20–24, 2020, Taipei, Taiwan

ACM ISBN 978-1-4503-7023-3/20/04.
https://doi.org/10.1145/3366423.3380122

1 INTRODUCTION

Product search provided by eCommerce sites is an important service allowing customers to search for products which they can purchase, or upon which they can take some actions such as adding to their shopping cart, caching for future investigation, or exploring their attributes. Recently, product search has also been supported by intelligent voice assistants such as Google home and Amazon Alexa, where customer can use the assistant's voice interface to search for products over online catalogs [19].

Given a user query, the quality of the search results is a leading factor in affecting user satisfaction. When quality signals are associated with training instances, either implicitly (e.g. through user behavioral data) or explicitly (e.g. through manual annotations) a Learning-to-Rank (LTR) approach is typically used to train a ranking model over the search results [17]. The model ranks the products with respect to the query, with the goal of achieving maximum agreement with the rank induced by the given quality signals, where agreement is typically measured by standard IR precision metrics such as NDCG or MRR [4]. When more than one signal is given, an essential question arises, as to how should the signals be combined in order to optimize for search quality.

Multi-Objective Ranking Optimization (MORO) [26] is the common approach for training a ranking model while optimizing multiple objectives simultaneously. Previous studies have investigated MORO for Web search while combining several quality signals [7, 8, 13, 26, 27]. MORO has also been extensively studied in the context of product search [14, 18, 22, 23]. Momma et al. [22] discuss typical objectives that should be considered in product search, including relevance to the query, purchase likelihood, product quality, user rating, return rate, and many more.

Typically, for non-trivial multi-objective optimization problems, there is no feasible solution that maximizes all objective functions simultaneously. Therefore, we look for Pareto optimal solutions; that is, solutions that cannot be improved for one objective without degrading at least one of the other objectives. The set of Pareto optimal solutions is called the *Pareto frontier* [20].

When the quality signals are given as graded labels, label aggregation is a popular multi-objective optimization method which reduces the multi-objective to a single-objective optimization problem. A specific example is linear label aggregation, which generates a new label for each training example by a linear combination of its given labels. Then, a model is trained using a single-objective LTR algorithm based on the new aggregated labels. The advantage of this approach is the ability to exploit existing single-objective LTR frameworks for the case of the MORO problem. The optimality expectation is that an optimal solution for the resulting single-objective optimization problem would be on the Pareto Frontier of the multi-objective optimization problem.

In this work we explore several label aggregation methods for MORO in product search. We propose a novel stochastic label aggregation technique which randomly selects a label per training example according to a given distribution over the labels. We identify a new solution concept, a set of models that cannot be dominated by any combination of models on the Pareto Frontier. We theoretically prove that any existing model in this solution concept can be constructed by a proper parameter setting of the stochastic label aggregation process. Moreover, we prove that this does not hold in the case of deterministic aggregation, by describing a specific MORO problem, with an existing Pareto-optimal solution, that cannot be reached by any deterministic label aggregation method.

Empirically, we experiment with three datasets sampled from product search query logs with multiple quality signals. Two of the datasets are sampled from the product search log of a commercial voice assistant, with two given objectives: the product relevance to the query and its purchase likelihood. The third one is a publicly available product search dataset used to validate our results on data that was not originated from the voice search domain. For all datasets, we first show that the problem is non-trivial, i.e., there is a tradeoff between the two objectives, as optimizing for one objective degrades the model performance with respect to the other objective. We consider a MORO method as superior to another one if its tradeoff curve fully dominates the tradeoff curve of the other method. For all datasets, we show that stochastic label aggregation fully dominates deterministic label aggregation. We would like to emphasize that comparing our solution to any existing MORO approach is out of the scope of this work and is left for future work.

2 RELATED WORK

Product search in the eCommerce domain has attracted much attention in recent years. Many studies investigated the relations between the two objectives we consider, namely relevance and purchase likelihood [2, 15, 18, 25, 28]. Alonso and Mizzaro [2], and recently Carmel et al. [5], presented evidence that when customers are interested in buying products, they apply many criteria in addition to relevance. Long et al. [18] proposed a new ranking framework for product search, based on combining relevance scores with best-selling prediction scores. In contrast, to the best of our knowledge, no previous research studied multi-objective optimization for product search in the voice domain. Recent research on voice-based search [10] has been conducted in the context of mobile devices, showing that users issue much longer queries, and the language of voice queries is closer to natural language than typed queries.

Multi-objective optimization methods have been explored for many years. The survey of Marler and Arora [20] provides a comprehensive review of multi-objective optimization concepts and methods. Many of the methods they present under the category of "methods with a priori articulation of preferences" are used in the works mentioned below.

Many works have focused on the multi-objective optimization problem in the context of information retrieval and recommender systems [1, 7, 8, 13, 24, 26]. Two main approaches can be distinguished in these works. The first is learning how to combine multiple objectives into one global model [1, 7, 26], done by aggregating the labels that relate to the different objectives. The second is learning how to combine multiple models, one built for each objective [13, 23, 24]. Svore et al. [26] optimized multiple objectives based on relevance judgments of human annotators and on click feedback. They improved clicks based measures without decreasing relevance, by requiring that document pairs with the same relevance label are swapped if the lower-ranked document has a higher click through rate. Dai et al. [7] considered both the relevance and freshness of the results, and generated hybrid labels for incorporating both criteria into ranking optimization. Kang et al [13] considered multiple relevance aspects and studied two types of learning-based approaches to estimate the tradeoff between these aspects: a label aggregation method and a model aggregation method. Rodriguez et al. [24] also considered multiple relevance objectives and addressed the problem of multiple objective optimization in the context of recommender systems.

During the last years, the multi-objective ranking optimization has attracted attention specifically in the domain of eCommerce [14, 22, 23, 28]. Karmaker Santu et al. [14] focused on user feedback signals (such as click rate, order rate, revenue and add-to-cart) as objective criteria for training and evaluation. Nguyen et al. [23] considered the consumers objective as well as the suppliers and intermediaries objectives. Given an initial ranking of item recommendations built for the consumer, they re-rank it such that it is also optimized for the secondary objectives while staying close to the initial ranking. Wu et al. [28] utilized implicit user feedback signals to optimize for sales revenue. They considered both clicks and purchases for modeling the two stages of the purchase journey and propose a nested framework to model the interdependence of the two stages.

A work that is highly relevant to our context is of Momma et al. [22], that leverages a multi-objective optimizer for building a multi-objective ranking model for product search. They considered a setting where there are many objectives to optimize such as customer engagement, search defects, diversity, product freshness, product quality, and more. Their solution compromises on the optimality of a primary objective while attaining a threshold limit on the performance of the other sub-objectives, which act as guardrails.

Recently, Lin et al. [16] proposed a Pareto efficient solution for multi-objective ranking in e-commerce. Their linear scalarization-based solution [11] is based on optimizing a weighted aggregation function of the multiple objectives using gradient decent. They showed that the solution belongs to the Pareto frontier, however, they did not show the opposite direction that any Pareto solution is achievable. We theoretically prove in this work that our stochastic label aggregation approach, at the training instance level, guarantees accessibility to any Pareto solution, by setting the right configuration.

3 MULTI-OBJECTIVE LEARNING TO RANK

3.1 Background

3.1.1 Single Objective Ranking Optimization .

We formalize the LTR task for information retrieval [17], and in particular for eCommerce search [14]. At training time, we are given a set of n product related queries $Q = \{q_1, \ldots, q_n\}$; each $q \in Q$ is associated with a set of products $\mathbb{P}_q = \{p_1, \ldots, p_{k_q}\}$. Each product $p_i \in \mathbb{P}_q$ is represented as a query dependent feature vector, with a corresponding relevance label $\ell(q, p_i) \in [0, 1]$ that indicates the relevance of p_i to query q. The 0 label represents an irrelevant product while 1 represents the highest relevance grade. Let $\mathbb{L}_q = \{\ell(q, p_1), \ldots, \ell(q, p_{k_q})\}$ be the labels of all products in $\mathbb{P}_q$. A training instance of the LTR task consists of the tuple $I = (q, \mathbb{P}_q, \mathbb{L}_q)$.

Let $\mathscr{A}$ denote a single-objective LTR algorithm with a fixed set of hyper parameters (e.g., number of trees, trees depth, etc.). Given a sample of training instances $\mathscr{I} = \{I_1, \ldots, I_m\}$, the goal of $\mathscr{A}$ is to create a ranking model, $M(\cdot, \cdot) = \mathscr{A}(\mathscr{I})$, that scores a product p with respect to query q, such that the ranking induced by the M scores over $\mathbb{P}_q$ has maximal agreement with $\mathbb{L}_q$. The loss function $\text{Loss}_\ell(I, M)$ gets as input a training instance I, together with the ranking model M, and estimates the disagreement between the orders induced by M and by $\mathbb{L}_q$ over $\mathbb{P}_q$. The cost function that $\mathscr{A}$ tries to minimize is $\text{Cost}_\ell(\mathscr{I}, M) = \mathbb{E}_{I \in \mathscr{I}}[\text{Loss}_\ell(I, M)]$. When $\mathscr{I}$ is clear from the context we will denote the cost by $\text{Cost}_\ell(M)$.

LambdaMart [3] is a state-of-the art LTR algorithm that is based on a pairwise cross-entropy loss function. The LambdaMart algorithm minimizes the cost function by iterative updates of the ranking model, an ensemble of gradient boosted decision trees, based on the approximation of the gradient of $\text{Cost}_\ell(M)$. While there is no theoretical guarantee for convergence to the global optimum, LambdaMart typically exhibits a good performance in practice [3].

3.1.2 Multi-Objective Ranking Optimization.

Typically for product search, there are several objectives we would like to optimize [22]. We assume, w.l.o.g., that we have two labeling objectives ℓ_1, ℓ_2, and our goal is to devise a ranking algorithm that learns a ranking model that minimizes the costs with respect to both objectives. Given a ranking model M, we define the loss function for each of the objectives in a similar manner to the single objective case, $\text{Loss}_{\ell_i}(I, M)$, $i \in \{1, 2\}$, and $\text{Cost}_{\ell_1,\ell_2}(M) = (\text{Cost}_{\ell_1}(M), \text{Cost}_{\ell_2}(M))$. Thus, there is only a natural partial order over the multiple objective costs, where $(\text{Cost}_{\ell_1}(M), \text{Cost}_{\ell_2}(M)) \le (\text{Cost}_{\ell_1}(M'), \text{Cost}_{\ell_2}(M'))$ if and only if $\text{Cost}_{\ell_1}(M) \le \text{Cost}_{\ell_1}(M')$ and $\text{Cost}_{\ell_2}(M) \le \text{Cost}_{\ell_2}(M')$. Since typically there is no single model minimizing both objectives, many models can be considered "optimal", while each of them represents a different tradeoff between the two objective. Therefore, an optimal solution is based on the *Pareto Frontier* concept, defined as follows.

We denote by $\mathscr{M}$ the set of all models that can be returned by an algorithm $\mathscr{A}$ (e.g., all models that can be returned by LambdaMart with a specific configuration set of hyper parameters). The Pareto Frontier of $\mathscr{M}$, with respect to a given training set $\mathscr{I}$, is defined by:

$$\text{Par}(\mathscr{I}, \mathscr{M}) = \\ \left\{M \in \mathscr{M} \mid \forall M' \in \mathscr{M} : \text{Cost}_{\ell_1,\ell_2}(\mathscr{I}, M') \not< \text{Cost}_{\ell_1,\ell_2}(\mathscr{I}, M)\right\}.$$

Again, when $\mathscr{I}$ is clear from the context we omit it from the expression.

3.1.3 Multi-Objective Approaches.

Multi-objective optimization is usually based on three major approaches: (1) fusion of multiple models tuned independently for each objective [23, 27]; (2) reducing the problem to a single objective by aggregating multiple labels into a single one [26]; (3) the ϵ-constraint method which optimizes the primary objective while considering all other objectives as additional constraints on the solution [20, 22]. In the following we elaborate on these three approaches.

Model Fusion. Model fusion is based on the fusion of two independent ranking models [9]. The first model, M_{ℓ_1}, is trained based on the first objective, while M_{ℓ_2} is trained independently based on the second objective. The final product score used for ranking is a convex combination of the two models scores. In particular, a linear combination is implemented by the formula $M(q, p) = \alpha \cdot M_{\ell_1}(q, p) + (1 - \alpha) \cdot M_{\ell_2}(q, p)$, where the hyper parameter $\alpha \in [0, \ldots, 1]$ controls the tradeoff between the two model scores.

Label Aggregation. This approach aggregates the labels representing multiple objectives to form one single label, which can then be used in an existing LTR framework to find a ranking model that optimizes the consolidated objective function. As specified before, given a query q in our setting, each product $p \in \mathbb{P}_q$ is associated with two different labels denoted by $\ell_1(p, q)$ and $\ell_2(p, q)$. We reduce the problem into a single-objective learning-to-rank problem by aggregating the two labels of each product into one label. We describe two popular approaches for label aggregation:

- *Lexicographic.* We prioritize ℓ_1 and ℓ_2; one is the primary label and the other is the secondary one [26]. We order products according to the primary label, while ties are broken according to the secondary label.
- *Linear.* For each (q, p) pair we use a linear combination for computing a new label, $\ell(p, q) = \alpha \cdot l_1(p, q) + (1 - \alpha) \cdot l_2(p, q)$ [9]. The order between a pair of products is derived from the new label ℓ. We note that the lexicographic approach is a special case of the linear approach, where α is chosen to be very close to 0 or to 1. We also note that when $\alpha \in \{0, 1\}$, the problem is reduced to a single objective optimization.

ϵ-constraint. Assuming l_1 is the primary objective, solve the optimization problem of minimizing $\text{Cost}_{\ell_1}(M)$ subject to $\text{Cost}_{\ell_2}(M) \le \epsilon$. The ϵ-constraint method is quite popular, especially in cases where the significance order between objectives is clear. However, it is easy to show that a solution provided by such a method is not necessarily on the Pareto Frontier. Moreover, existing LTR frameworks cannot be used in this case as the cost function should be re-implemented for supporting the additional constraints. An example for this type of solutions is provided by Momma et al.[22] who describe an ϵ-constraint method for MORO problems using the Augmented Lagrangian inequality constraint approach.

3.2 Stochastic Label aggregation

Our main contribution is a novel MORO approach called *stochastic label aggregation*. For each query we flip a coin with probability α to determine its label, hence, the label of all products in $\mathbb{P}_q$ will be ℓ_1 with probability α, or ℓ_2 with probability $(1-\alpha)$. To this end, we have approximately $\alpha \cdot |\mathscr{I}|$ training instances with results labeled by ℓ_1 and $(1-\alpha) \cdot |\mathscr{I}|$ instances labeled by ℓ_2.

The stochastic label aggregation approach has many advantages over deterministic approaches, as we show both theoretically and empirically in this work. Moreover, while deterministic label aggregation approaches rely on the assumption that all (q, p) pairs are fully labeled according to all objectives, in the stochastic case, even if some of the training examples are only partially labeled, we can still train our system by creating a representative training sample with queries labeled by ℓ_1 and queries labeled by ℓ_2, according to the desired distribution. The scenario of partial labeling is quite common in practice, e.g., when only a small portion of the data is annotated with relevance labels due to the high annotations cost.

3.3 Two-Phase Model Combination

The second MORO approach we propose is a novel integration of label aggregation and model fusion. As in the model fusion approach, we train at first two independent models: M_{ℓ_1} based on the first objective, and M_{ℓ_2} based on the second objective. In the second phase, we train an LTR model that considers $M_{\ell_1}(q, p)$ and $M_{\ell_2}(q, p)$ scores as features of the (q, p) pair. Training in the second phase is based on these two features, and is optimized with respect to a new aggregated label, computed by combining the labels ℓ_1 with ℓ_2 using any of the aggregation methods described in the previous section.

Our experimental results (see Section 5.3) reveal that the two-phase approach provides a very strong optimization mechanism, which can be partially explained due to its richer architecture. Moreover, while a single-phase mechanism aggregates the different objectives by constructing a single label, the two-phase mechanism integrates the different objectives via the internal representation of each of the training instances (as provided to the second phase). Thus, the learning algorithm of the two-phase mechanism may benefit from possible relationships between the objectives that are reflected while comparing between training instances.

4 THEORETICAL ANALYSIS

4.1 Solution concept

In order to present the theoretical advantage of stochastic aggregation over deterministic aggregation approaches, we first extend the concept of a solution for a MORO problem. We use Figure 1 to illustrate the costs of some models and the relations between them. Each axis represents a different objective, and $\text{Cost}_{\ell_1,\ell_2}(M)$ is represented by a point on the graph. Model A is not on the Pareto Frontier of $\mathscr{M}$, since model B has a lower cost on both axes (objectives). Hence, any cost-minimizing decision maker will prefer B over A. Since B is on the Pareto Frontier of $\mathscr{M}$, there is no single model in $\mathscr{M}$ that outperforms B.

4.1.1 Mixed Models. Given two models, each with a lower cost on a different objective (e.g. models C and D in Figure 1), a decision maker may prefer a model CD with a cost equal to the average cost of C and D. Given any two (or more) models, one can easily construct a model that achieves any convex combination of their costs, using a mixed model defined as follows.

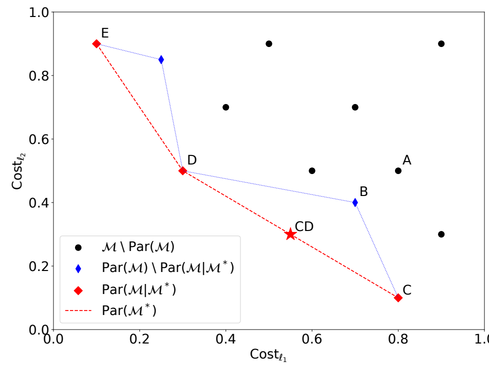


Figure 1: Pareto frontiers: $Par(\mathscr{M})$, $Par(\mathscr{M}^*)$ and $\text{Par}(\mathscr{M}|\mathscr{M}^*)$

A mixed model M^* is defined by a probability distribution D over $\mathscr{M}$. For any query q, M^* randomly selects a model $M \in \mathscr{M}$, with respect to D (independently of the other queries), and orders the products in P_q according to the $M(q, p)$ scores. Naturally, the support of M^* is the set of all models from $\mathscr{M}$ with positive probability according to D. The next proposition shows that a mixed model achieves the expected cost of the models in its support.

Proposition 1. *Let M^* be a mixed model, and let $D(M^*)$ be the probability distribution over $\mathscr{M}$ used to construct M^*. Then, for any objective ℓ,* $\text{Cost}_\ell(M^*) = \mathbb{E}_{M \sim D(M^*)}[\text{Cost}_\ell(M)]$.

Proof. The proof follows directly from the law of total expectation:

$$\text{Cost}_\ell(M^*) = \mathbb{E}_{I \in \mathscr{I}}[\text{Loss}_\ell(I, M^*)]$$
$$= \mathbb{E}_{M \sim D(M^*)}\left[\mathbb{E}_{I \in \mathscr{I}}[\text{Loss}_\ell(I, M)]\right] = \mathbb{E}_{M \sim D(M^*)}\left[\text{Cost}_\ell(M)\right].$$

□

We denote by $\mathscr{M}^*$ the set of all mixed models over $\mathscr{M}$. It is easy to see that $\mathscr{M} \subseteq \mathscr{M}^*$. Note also that since the cost of the mixed model can be directly computed from the costs of the pure models in its support, one does not need to explicitly construct the mixed models in order to compute their costs. A decision maker can always choose a mixed model based on the costs of the pure models in its support. Therefore, based on the costs of models C and D, a decision maker will prefer model CD (in Figure 1) over model B. We also note that in the case of one objective, there is no reason to consider mixed models. In this case, combining the cost-minimizing model with any other model does not bring any value.

4.1.2 Optimal Solutions. The set of optimal solutions for the reduced single-objective problem, $Par(\mathscr{M})$, may contain models that are non-optimal for the original multi-objective problem, due to mixed models that may dominate them (e.g. Model CD in Figure 1 which dominates model B). We characterize the set of optimal solutions for the multi-objective problem, based on the notion of *Pareto Frontier* and *mixed models*. Our goal is to identify all models in $Par(\mathscr{M})$ that are not dominated by any mixed model in $\mathscr{M}^*$.

Formally, we generalize the aforementioned definition of Pareto Frontier by

$$\mathrm{Par}(\mathscr{M}|\mathscr{M}^*) = \left\{M \in \mathscr{M} \mid \forall M^* \in \mathscr{M}^* : \mathrm{Cost}_{\ell_1,\ell_2}(M^*) \not< \mathrm{Cost}_{\ell_1,\ell_2}(M)\right\}.$$

The next proposition shows that any mixed model in $\mathscr{M}^*$ is dominated by a mixed model that is supported by only *two* models from $\mathrm{Par}(\mathscr{M}|\mathscr{M}^*)$, and that can thus be efficiently computed. This demonstrates that $\mathrm{Par}(\mathscr{M}|\mathscr{M}^*)$ is indeed a strong solution concept.

Proposition 2. *For any* $M^* \in \mathscr{M}^*$ *there exists a mixed model* $M^{*\prime} \in \mathscr{M}^*$*, supported by two models* $\{M_1, M_2\} \in \mathrm{Par}(\mathscr{M}|\mathscr{M}^*)$ *such that* $\mathrm{Cost}_{\ell_1,\ell_2}(M^{*\prime}) \leq \mathrm{Cost}_{\ell_1,\ell_2}(M^*)$.

Proof. See Appendix A.1. □

Looking again at model $B \in \mathrm{Par}(\mathscr{M})$ in Figure 1, we argue that a cost-minimizing decision maker will never choose it since it is not on $\mathrm{Par}(\mathscr{M}|\mathscr{M}^*)$. Indeed, for each training instance I, one can order the products according to model C, with probability 0.5, or according to model D, with probability 0.5. The point CD represents the costs of this mixed model which are lower than the costs of model B, with respect to both objectives. Therefore, the decision maker will prefer CD over B, hence $B \notin \mathrm{Par}(\mathscr{M}|\mathscr{M}^*)$. The red curve represents $Par(\mathscr{M}^*)$, and the red points that span it belong to $\mathrm{Par}(\mathscr{M}|\mathscr{M}^*)$. Finally, we note that the decision maker can choose any model on the Pareto Frontier of $\mathscr{M}^*$, since there is no mixed model that outperforms them.

4.2 Optimality

We are now ready to prove that the family of MORO models $\mathscr{M}_{\text{stoch}}$, constructed using stochastic label aggregation, is superior to the family of models $\mathscr{M}_{\text{det}}$ constructed using deterministic label aggregation. In order to establish the proof, we show that $\mathscr{M}_{\text{stoch}}$ equals $\mathrm{Par}(\mathscr{M}|\mathscr{M}^*)$ while $\mathscr{M}_{\text{det}}$ does not.

Let ℓ_α be a stochastic label, i.e., the final label resulting from choosing ℓ_1 with probability α or choosing ℓ_2 with probability $1-\alpha$, where the probabilities are independent over the training instances. Let $\mathrm{Cost}_{\ell_\alpha}(M)$ be the (single-objective) cost function of model M according to ℓ_α. The following proposition defines the relation between $\mathrm{Cost}_{\ell_\alpha}(M)$ and $\mathrm{Cost}_{\ell_1,\ell_2}(M)$.

Proposition 3. *Given* $\vec{\alpha} = (\alpha, 1-\alpha)$*, then* $\mathrm{Cost}_{\ell_\alpha}(M) = \vec{\alpha} \cdot \mathrm{Cost}_{\ell_1,\ell_2}(M)$.

Proof. Following directly from the law of total expectation:

$$\begin{aligned}
\mathrm{Cost}_{\ell_\alpha}(M) &= \mathbb{E}_{I\in\mathscr{I}}[\mathrm{Loss}_{\ell_\alpha}(I, M)] \\
&= \alpha \cdot \mathbb{E}_{I\in\mathscr{I}}\left[\mathrm{Loss}_{\ell_1}(I, M)\right] + (1-\alpha)\mathbb{E}_{I\in\mathscr{I}}\left[\mathrm{Loss}_{\ell_2}(I, M)\right] \\
&= \alpha \cdot \mathrm{Cost}_{\ell_1}(M) + (1-\alpha) \cdot \mathrm{Cost}_{\ell_2}(M) = \vec{\alpha} \cdot \mathrm{Cost}_{\ell_1,\ell_2}(M).
\end{aligned}$$

□

4.2.1 Optimality of Stochastic Label Aggregation.

Theorem 1. $\mathscr{M}_{\text{stoch}} = \mathrm{Par}(\mathscr{M}|\mathscr{M}^*)$.

Proof. We prove a bi-directional set inclusion, that is, $\mathscr{M}_{\text{stoch}} \subseteq \mathrm{Par}(\mathscr{M}|\mathscr{M}^*)$ and $\mathrm{Par}(\mathscr{M}|\mathscr{M}^*) \subseteq \mathscr{M}_{\text{stoch}}$.

$\mathscr{M}_{\text{stoch}} \subseteq \mathrm{Par}(\mathscr{M}|\mathscr{M}^*)$: We consider a model $M_\alpha \in \mathscr{M}_{\text{stoch}}$ that minimizes $\mathrm{Cost}_{\ell_\alpha}$ and show that no other model $M' \in \mathscr{M}^*$ has a lower cost in both objectives, i.e., $\mathrm{Cost}_{\ell_1,\ell_2}(M') \not< \mathrm{Cost}_{\ell_1,\ell_2}(M_\alpha)$. Indeed, if there is such a model M', then $\vec{\alpha} \cdot \mathrm{Cost}_{\ell_1,\ell_2}(M') < \vec{\alpha} \cdot \mathrm{Cost}_{\ell_1,\ell_2}(M_\alpha)$. By Proposition 3, we get $\mathrm{Cost}_{\ell_\alpha}(M') < \mathrm{Cost}_{\ell_\alpha}(M_\alpha)$, in contradiction to the optimality of M_α with respect to $\mathrm{Cost}_{\ell_\alpha}$. Therefore, $M_\alpha \in \mathrm{Par}(\mathscr{M}|\mathscr{M}^*)$.

$\mathrm{Par}(\mathscr{M}|\mathscr{M}^*) \subseteq \mathbf{M}_{\text{stoch}}$: We consider a model $M \in \mathrm{Par}(\mathscr{M}|\mathscr{M}^*)$ and show the existence of $\alpha \in [0,1]$ such that M minimizes $\mathrm{Cost}_{\ell_\alpha}$, therefore $M \in \mathscr{M}_{\text{stoch}}$. The proof of the existence of such α for any model in $\mathrm{Par}(\mathscr{M}|\mathscr{M}^*)$ is given in Appendix A.2. □

4.2.2 Sub-Optimality of Deterministic Label Aggregation.

The next proposition shows that $\mathscr{M}_{\text{det}}$ does not cover $\mathrm{Par}(\mathscr{M}|\mathscr{M}^*)$, and thus is inferior to $\mathscr{M}_{\text{stoch}}$.

Proposition 4. $\mathrm{Par}(\mathscr{M}|\mathscr{M}^*) \not\subseteq \mathscr{M}^*_{\text{det}}$.

Proof. We describe a simple training sample $\mathscr{I}$ and show the existence of a model in $\mathrm{Par}(\mathscr{M}^*)$ such that any model returned by a deterministic aggregation method has a considerably higher cost value in at least one of the objectives.

We consider two queries q_1, q_2, each associated with two products, $\mathbb{P}_{q_1} = \{a, b\}$ and $\mathbb{P}_{q_2} = \{c, d\}$. Each product has two binary labels, ℓ_1, ℓ_2[1]. Our training sample $\mathscr{I}$ consists of 100 training instances: 91 instances associated with query q_1, and 9 instances with query q_2. Table 1 summarizes sample $\mathscr{I}$.

Table 1: Proposition 4: the training sample

	#I	$\boldsymbol{q}$	$\mathbb{P}_{\boldsymbol{q}}$	$\boldsymbol{\ell_1}$	$\boldsymbol{\ell_2}$
(1)	90	q_1	a	1	1
			b	1	0
(2)	1	q_1	a	0	1
			b	1	0
(3)	5	q_2	c	1	1
			d	1	0
(4)	4	q_2	c	0	1
			d	1	0

We mark $\mathscr{M}_{x\rhd y, u\rhd v}$ for models that rank x on top of y and u on top of v. All ranking models consistent with our sample can be classified into four equivalent classes $\mathscr{M}_{a\lhd b, c\lhd d}$, $\mathscr{M}_{a\rhd b, c\rhd d}$, $\mathscr{M}_{a\rhd b, c\lhd d}$ and $\mathscr{M}_{a\lhd b, c\rhd d}$. We assume that $\mathscr{M}$ contains a representative of each of the aforementioned classes. Without loss of generality we assume that the cost of ordering a pair in the wrong order is 1. Consider $M \in \mathscr{M}_{a\lhd b, c\lhd d}$. By looking at all 4 types of labeled instances, M ranks the products in the correct order according to ℓ_1 and in the wrong order according to ℓ_2, thus $\mathrm{Cost}_{\ell_1}(M) = 0$ and $\mathrm{Cost}_{\ell_2}(M) = 1$. Now consider $M \in \mathscr{M}_{a\rhd b, c\rhd d}$. It can easily be seen that a loss according to label 1 occurs only in cases (2) and (4), which together cover 5 out of 100 instances, thus $\mathrm{Cost}_{\ell_1}(M) = 0.05$ and $\mathrm{Cost}_{\ell_2}(M) = 0$. Table 2 summarizes the costs of each model class with respect to both labels, together with the instances in which the loss occurred.

[1] Note the existence of (q, p) pairs with contradicting labels, e.g., there are 5 (q_2, c) pairs with $\ell_1 = 1$ and 4 pairs with $\ell_1 = 0$. This is typical to real datasets where objectives are measured by a noisy process, for example, manual relevance judgments.

Table 2: Proposition 4, costs of all possible models

M	$\text{Cost}_{\ell_1}(M)$	$\text{Cost}_{\ell_2}(M)$	$\text{Loss}_{\ell_1} = 1$	$\text{Loss}_{\ell_2} = 1$
$a \lhd b, c \lhd d$	0	1		(1),(2),(3),(4)
$a \rhd b, c \rhd d$	0.05	0	(2),(4)	
$a \rhd b, c \lhd d$	0.01	0.09	(2)	(3),(4)
$a \lhd b, c \rhd d$	0.04	0.91	(4)	(1),(2)

We show that any learning algorithm with deterministic label aggregation can return only models from $\mathcal{M}_{a \lhd b, c \lhd d}$ or $\mathcal{M}_{a \rhd b, c \rhd d}$, while it cannot return any model from $\mathcal{M}_{a \rhd b, c \lhd d}$. Let $l : \{0,1\} \times \{0,1\} \to \mathbb{R}$ be a deterministic function that aggregates the two labels. We prove the claim by considering the following four cases:

(1) $l(1,1) > l(1,0)$. In this case ranking a above b will be wrong for at most 1 example out of the 91 examples associated with q_1, while ranking a below b will be wrong for 90 cases. Thus, a cost minimizing algorithm will always rank a above b. Similarly, this algorithm will rank c above d since it prefers to use the wrong order only for 4 examples rather than 5 examples. Thus, the only models that can be returned in this case belong to $\mathcal{M}_{a \rhd b, c \rhd d}$.
(2) $l(1,1) < l(1,0)$. Using the same argument as in Case (1), only models from $\mathcal{M}_{a \lhd b, c \lhd d}$ will be returned.
(3) $l(1,1) = l(1,0)$ and $l(0,1) > l(1,0)$. Ordering a above b and c above d is the only way to get zero cost and thus the only models that can be returned in this case are from $\mathcal{M}_{a \rhd b, c \rhd d}$.
(4) $l(1,1) = l(1,0)$ and $l(0,1) < l(1,0)$. Similarly to Case (3), the models returned will be from $\mathcal{M}_{a \lhd b, c \lhd d}$.

Figure 2 presents the costs of all consistent models with the training sample. As shown above, there is no deterministic label-aggregation method that can return a model from $\mathcal{M}_{a \rhd b, c \lhd d}$, which belongs to the Pareto frontier. On the other hand, by Theorem 1, the stochastic label aggregation method can output any model on the Pareto Frontier of problem, including those in $\mathcal{M}_{a \rhd b, c \lhd d}$, completing the proof. □

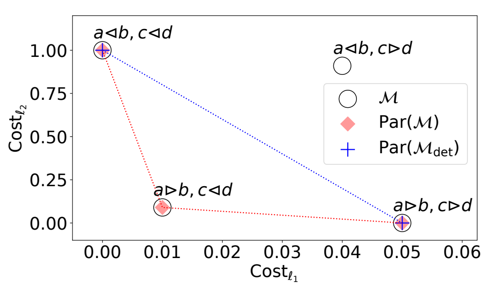


Figure 2: Proposition 4: deterministic vs stochastic label aggregation costs

Discussion: In the example above, for each query, the majority of instances agree on label ℓ_1 (for q_1 the majority is 99%, while for q_2 it is 56%). Note that deterministic aggregation methods do not take into account the level of agreement on the label. This is due to the fact that ordering is determined according to the second label in any case of agreement on the first label, no matter the level of agreement is. A stochastic method however, can output models where ordering is set with respect to the agreement level on the label.

We also would like to note that when the granularity of label values is low (e.g. a binary label), the level of agreement on the label value between instance pairs becomes higher. In Section 5.3.3, we empirically show that for lower granularity levels of label values, deterministic label aggregation methods cover smaller parts of the solution space, thus increasing their inferiority.

5 EXPERIMENTS

5.1 Datasets

We evaluate our MORO approaches on three datasets coming from two different domains: the domain of voice product search and the domain of Web product search.

Voice Product Search. We use two voice shopping datasets based on product search traffic from *headless* devices (devices without a screen): the *aggregated* dataset and the *raw* dataset. Each dataset includes (q, p) pairs associated with two labels: a relevance label R and a purchase label P. The relevance label R denotes the relevance of a product p to query q based on the relevance judgements of at least three human annotators, and is determined according to the majority vote. The purchase label P is derived by the number of purchases of p given q, relative to the number of times it was offered following q (number of impressions), during a time window of six weeks.

The aggregated dataset contains 27K queries with exactly two products per query. In this dataset each (q, p) pair has both a relevance label and a purchase label. Having both R and P labels for each data point allows testing deterministic label aggregation methods which require a complete labeled dataset. However, this requirement considerably limits the size of the dataset, due to the scalability limitation of human annotations. Furthermore, due to privacy considerations, only frequent queries were annotated, further limiting the size of the dataset and creating a bias towards frequent queries.

The raw dataset is based on a log of voice product search traffic collected over a period of two months. The log contains a list of (q, p) pairs. For each pair we have a binary purchase label indicating whether the product was purchased or not. In addition we may have a binary relevance label if the pair was manually annotated. We filtered out queries for which we have neither purchase nor relevance information, resulting with a dataset of 360K queries with roughly 11 products on average per query (3.8M products in total). About 70% of the (q, p) pairs have a purchase label, and about 30% have a relevance label. We note that most (q, p) pairs in the raw dataset have only one of the labels, therefore it can serve only the fusion and stochastic models that do not require having both labels per pair (see 3.1.3).

While both the raw and the aggregated datasets come from the voice domain they have different and complementary characteristics. On one hand, the raw dataset is much larger than the aggregated dataset, as it contains infrequent queries and multiple products per query, and thus provides a better approximation of real traffic. On the other hand, this dataset suffers from a presentation bias since customers are typically exposed to only one or two products[12]. Moreover, as mentioned above, unlike the aggregated dataset, it can be used for evaluating only part of the label aggregation approaches.

Web Product Search. In addition to the voice product search datasets we use a publicly available Web product search dataset[2]. The dataset contains 1K queries and 20K products. Each (q, p) pair is labeled with a relevance score between 1 and 4, where a score of 1 represents an irrelevant product while a score of 4 represents the highest relevance score of a product to the query.

The dataset contains only one label, and thus is ill-fitted for the task of multi-objective optimization. To adapt this dataset for our task we synthetically augment it with a second label, denoted by *Inverse Title Length* (ITL), defined as 1/{#words in the product title}. The motivation behind this label is that in the voice domain it is advantageous to have short titles, as customers have a limited attention span when listening to results by voice. Optimizing for ITL is a way to promote the ranking of products with shorter titles. We note that the single-objective optimization task of ranking products according to ITL is easy. However, the task of optimizing for both objectives is non-trivial.

Training and Evaluation. The datasets were split into three parts. The voice based datasets were partitioned chronologically, while the public web dataset (which contains no date information) was randomly partitioned. The partitions consist of a first phase train set (60%), a second phase train set (20%), and a test set (20%). Note that in the second phase we train a simple model based on two features only, therefore, its train set is allowed to be much smaller than the first phase train set. For models trained by a single phase we do not use the second phase train set. The datasets we used in our evaluation process are summarized in table 3. As in previous sections, ℓ_1 and ℓ_2 are the labels used for the multi-objective optimization task. We report the percentage of (q, p) pairs labeled with $\ell \in \{\ell_1, \ell_2\}$ in columns $\%\ell_1, \%\ell_2$.

Table 3: Datasets description

Dataset	Queries	Prods	ℓ_1, ℓ_2	$\%\ell_1$	$\%\ell_2$
aggregated (voice)	27K	54K	R, P	100%	100%
raw (voice)	360K	3.8M	R, P	30%	70%
public (web)	1K	20K	R, ITL	100%	100%

5.2 Models

We experiment with the models described in Section 3.1.3, that is, both single-phase and two-phase models crossed with the different multi-objective optimization approaches. The two-phase models are based on the combination of single-objective models, trained separately and independently for each objective. In the second phase, the scores of the single objective models are used to represent the training instances for training the final ranking model. Model fusion is obtained by a linear combination of the single objective models scores. We name the model families as follows, according the their characteristics:

- *1phase-stoch*: single-phase stochastic label aggregation.
- *1phase-linear*: single-phase linear label aggregation.
- *2phase-stoch*: two-phase stochastic label aggregation.
- *2phase-linear*: two-phase linear label aggregation.
- *fusion*: linear models score combination.

In our experiments we explore the tradeoff curves of the different model families, representing the tradeoff between the scores obtained for the two objectives[3]. We consider a model family to outperform another model family if its tradeoff curve dominates the other curve.

To generate the tradeoff curve of a model family we evaluate models from that family using $\alpha \in [0, 1]$ values in intervals of 0.002 (500 models per family). We calculate the scores of each model with respect to both objectives. To reduce clutter and de-noise the results we divide the α range to 10 equal buckets and average the model scores in each bucket. Thus, for each model family we get 10 points on the plot, where each point is the average score of a cluster of 50 models.

Our models were trained using the XGBoost[6] implementation of LambdaMART, with the following hyper parameters: 10 rounds, max_depth=2, num_trees = 100, learning_rate=1, and rank=pairwise. The main features we used for representing (q, p) pairs are textual similarity features, such as Jaccard similarity and Levenshtein distance between the query and the product details (title, brand, category); semantic similarity based on the cosine similarity between the average of the FastText[21] pre-trained word embedding vectors; as well as behavioral features based on past transactions. Note that optimizing LTR for the single-objective model is not within the scope of this work, as we focus on the combination of multiple objectives, while using a fixed LTR process for all MORO approaches we consider.

5.3 Results

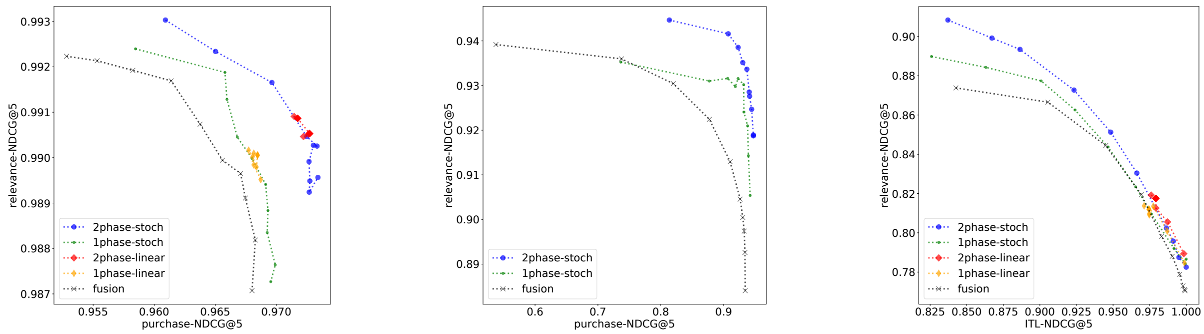
We present the experimental results using the different model families on the various datasets. In all experiments we fix all the hyper parameters of the learning task. The only free parameter is $\alpha \in [0, 1]$ that controls the label aggregation and the fusion processes. We present results based on NDGC@5 scores rather than Cost (as in Sections 3 and 4); the higher the NDGC@5 score, the better the performance is. Experimental results with other precision metrics such P@1, P@3, P@5, MRR@5, revealed very similar trends hence are not reported due to lack of space.

5.3.1 Voice Product Search. Figure 3a depicts the evaluation results on the aggregated voice dataset when optimizing NDCG@5 for both relevance and purchase objectives. Clearly, the *2phase-stoch* model family yields the best tradeoff curve. The *2phase-linear* family, while performing similar in a small range of the tradeoff curve, covers only a very limited part of it. The single-phase families are inferior to the two-phase families. Similarly, the *1phase-linear* family has a similar performance to the *1phase-stoch* family, but it fails to cover most of the tradeoff curve. The limitations of these deterministic model families stem from Theorem 1 and are further discussed in Section 5.3.3. The fusion family, while covering the full range of the tradeoff curve, is dominated significantly by the other families.

Figure 3b shows the tradeoff curves of *1phase-stoch*, *2phase-stoch*, and *fusion* families, on the raw dataset, when optimizing NDCG@5

[2] 'eCommerce search relevance' https://www.figure-eight.com/data-for-everyone/

[3] We differentiate between the tradeoff curve, which represents the output of our experiments, from the Pareto frontier which is the theoretical optimum.



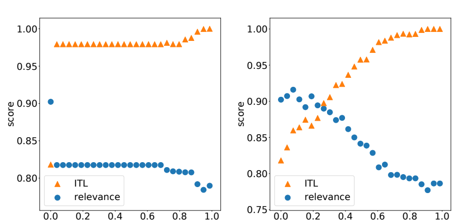
(a) Aggregated Voice Dataset **(b) Raw Voice Dataset** **(c) Public Web Dataset**

Figure 3: The tradeoff curves of the different model families on the three datasets: the *2phase-stoch* family dominates all other families.

for both relevance and purchase objectives. The deterministic label-aggregation families could not be evaluated on this dataset as they require both labels for all (q, p) pairs, while most pairs do not have both (see Section 5.1). As for the aggregated dataset, the *2phase-stoch* family dominates all other curves. The *1phase-stoch* family is better than the fusion family for most of the tradeoff curve.

5.3.2 Web Product Search. Figure 3c presents the evaluation results on the public Web dataset, for the task of optimizing NDCG@5 for both relevance and ITL objectives. The relevance label is binary and equals 1 if the raw relevance score > 3 and 0 otherwise. As in the previous experiments, *2phase-stoch* family outperforms all other families. The linear families capture only a small part of the tradeoff curve. The fusion family, while fully covering the tradeoff curve, is dominated by the stochastic families. As previously seen, the two-phase architecture is superior both in the linear case and in the stochastic case.

We note that while all model families achieve the maximum possible score for the trivial task of maximizing ITL, the stochastic methods have a clear advantage in the more complex task of optimizing both objectives.

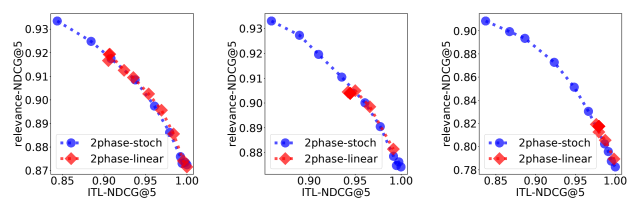


(a) *2phase-linear* **(b) *2phase-stoch***

Figure 4: Scores as a function of α (Web Dataset)

Figure 4 explores the effect of α, which controls the balance between ITL and relevance. Recall that $\alpha \in [0, 1]$, and a larger α emphasizes ITL over relevance. We present the results for the superior two-phase models.

We see that at the extremities ($\alpha = \{0, 1\}$) both the stochastic and the linear models achieve the same scores. This is not surprising since in these cases the models are reduced to similar single-objective models. The scores of the stochastic method are continuous and cover nicely the full range of the score values, showing smooth improvement in ITL score (and smooth decrease in relevance score) while increasing α. In contrast, for the linear deterministic method, the α values have low effect on performance, which is in agreement with the limited coverage of the tradeoff curve of this family. Moreover, it is evident that even setting a negligible α value, causes a sharp drop in the relevance score and a sharp jump in the ITL score. This behavior is explained by the fact that the binary relevance labeling has a high level of agreement between training instances, hence, any deterministic aggregation method is enforced to consider the second objective while comparing instances, in cases of agreement on the first objective, no matter the relative weight between the objectives (more on this behavior in the following subsection).



(a) Continuous **(b) Coarse** **(c) Binary**

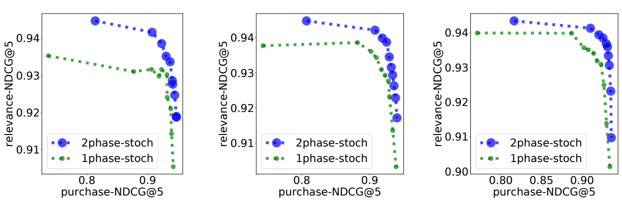
Figure 5: The performance of linear family degrades as the label granularity becomes less fine (Web Dataset).

5.3.3 Effect of Label Granularity. According to Theorem 1, there exist models on the Pareto Frontier that cannot be reached using deterministic label aggregation. What is the portion of the frontier that cannot be reached by the deterministic approach? We hypothesize that this question is related to label granularity. To verify this empirically, we generated three versions of relevance labels for the public dataset, with different levels of granularity. We expect a more significant degradation in performance of the deterministic model families as the level of granularity decreases.

The highest level of granularity, denoted *Continuous*, is achieved by using the raw relevance labels in the dataset, which is the average of independent manual annotation scores for each (q, p) pair. The raw relevance scores comprise about 50 distinct values between 1 and 4. A coarse level of granularity, denoted *Coarse*, is obtained by dividing the raw relevance labels into four bins (1,2,3,4) by rounding them to the closet integer. Finally, the lowest level of granularity, denoted *Binary*, is achieved by dividing the relevance labels into two buckets (relevant/irrelevant) where a (q, p) pair is considered relevant if the raw label is above 3 and irrelevant otherwise. In all cases, the relevance scores were linearly mapped into the $[0, 1]$ range.

We applied the *2phase-stoch* and *2phase-linear* model families on these three variations of the public dataset. Figure 5 depicts the results for the three levels of label granularity. As expected, when the granularity is low, the performance of *2phase-linear* decreases: it covers a smaller part of the trade-off curve compared to *2phase-stoch*, failing to reach high relevance scores. Note that even when using the highest level of granularity, the stochastic family remains superior.

We note that the granularity level is a significant factor, as in many practical use cases the label itself is binary by nature (i.e. purchase/no purchase), discrete, or divided into a small number of bins. Binning is often used as a pre-processing step to de-noise and stabilize the data. In these cases using a stochastic label-aggregation family has a clear advantage.



(a) Depth=2 (b) Depth=3 (c) Depth=4

Figure 6: The single-phase family closes the gap with the two-phase family as the tree-depth grows (Raw Dataset).

5.3.4 Effect of Model Depth. All experiments so far revealed the superiority of the two-phase stochastic family. As we claimed there are two main reasons for this advantage. First, the two-phase method benefits from implicit relationships between the different objectives. Indeed, in all our experiments we saw that optimizing for a single objective using the two-phase method yielded a better score than using the single-phase method. For example, optimizing for purchase on the raw dataset yielded an NDCG@5 score of 0.493, while single-phase optimization yielded only 0.395 [4].

Second, the two-phase method utilizes a richer architecture. To substantiate this we evaluated the relative performance of the two-phase and single phase stochastic families on the raw dataset using LambdaMart with a different tree-depth parameter. The raw dataset was selected for this experiment since applying deeper models to the smaller datasets could have caused over-fitting issues.

[4]This is not exposed in the plots, since the points representing $\alpha = \{0, 1\}$ are absorbed in the first and the last bins.

Figure 6 displays the performance of both stochastic families for tree-depth = {2,3,5}. In each sub-figure both families share the depth parameter (as well as the other hyper parameters). We can see that as the architecture becomes richer (higher depth) the single-phase stochastic family closes its gap to the two-phase family. While it is evident from the experiment that the single-phase family can perform as well as the two-phase family using more complex ranking models, the later better fits small datasets, decreasing the risk of over-fitting.

6 SUMMARY

In this work, we introduced the stochastic label aggregation method for MORO problems which provides a strong mechanism for learning a ranking model that optimizes several objectives simultaneously. We proved that the stochastic aggregation method is able to reveal any solution on the modified Pareto curve, as shown by Theorem 1. Moreover, this method provides decision makers with the ability to precisely set the tradeoff between objectives by choosing the proper α value in the stochastic aggregation process (as shown by Proposition 3).

The advantage of stochastic aggregation was also demonstrated by our experimental results over three datasets. We showed that the tradeoff curve of the stochastic label aggregation approach dominates deterministic (linear) label aggregation approaches which only covers a small part of the curve. Finally, we proposed a two-phase stochastic method for the MORO problem, and analyzed cases where this method is superior to the single phase stochastic method.

In the future, we plan to compare our method with other MORO approaches in an online product search system, validating its superiority as demonstrated in this work, based on users' feedback.

A APPENDIX

A.1 Proof of Proposition 2

Proposition (2, restated). *For any $M^* \in \mathscr{M}^*$ there exists a mixed model $M^{*\prime} \in \mathscr{M}^*$, supported by two models $M_1, M_2 \in$* $\mathrm{Par}(\mathscr{M}|\mathscr{M}^*)$*, such that* $\mathrm{Cost}_{\ell_1,\ell_2}(M^{*\prime}) \leq \mathrm{Cost}_{\ell_1,\ell_2}(M^*)$.

Proof. We say that model M dominates model M' if $\mathrm{Cost}_{\ell_1,\ell_2}(M) \leq \mathrm{Cost}_{\ell_1,\ell_2}(M')$. We say that the domination is strict, if $\mathrm{Cost}_{\ell_1,\ell_2}(M) < \mathrm{Cost}_{\ell_1,\ell_2}(M')$. For $M \in \mathscr{M}^*$, we denote the set of all models in its support by $\mathscr{M}_M$.

Our goal is to show that any model $M^* \in \mathscr{M}^*$ is dominated by a model $M^{*\prime}$ that is supported by two models from $\mathrm{Par}(\mathscr{M}|\mathscr{M}^*)$. We prove it in two stages. In the first stage we prove that M^* is dominated by a model $M^{*\prime\prime}$ that satisfies $\mathscr{M}_{M^{*\prime\prime}} \subseteq \mathrm{Par}(\mathscr{M}|\mathscr{M}^*)$. In the second stage we prove that $M^{*\prime\prime}$ is dominated by a model $M^{*\prime}$ supported by two models from $\mathrm{Par}(\mathscr{M}|\mathscr{M}^*)$.

First stage: Let $M^* \in \mathscr{M}^*$. We construct $M^{*\prime\prime}$ by an iterative process. As we show in the following, if there is a model $M \in \mathscr{M}^*$ such that $\mathscr{M}_M \nsubseteq \mathrm{Par}(\mathscr{M}|\mathscr{M}^*)$, then there must be a model $M' \in \mathscr{M}^*$ that strictly dominates M. Therefore, in each iteration, if $\mathscr{M}_{M^*} \nsubseteq \mathrm{Par}(\mathscr{M}|\mathscr{M}^*)$, we replace it with a model that strictly dominates it. Since at each iteration the model is strictly improved, a convergence is guaranteed to a model $M^{*\prime\prime}$ that dominates M^*, and $\mathscr{M}_{M^{*\prime\prime}} \subseteq \mathrm{Par}(\mathscr{M}|\mathscr{M}^*)$.

We now show that if $M \in \mathscr{M}^*$ and $\mathscr{M}_M \nsubseteq \mathrm{Par}(\mathscr{M}|\mathscr{M}^*)$ then there is a model that strictly dominates it. Let $M_1 \in \mathscr{M}_M \setminus$

$\mathrm{Par}(\mathscr{M}|\mathscr{M}^*)$. Since $M_1 \notin \mathrm{Par}(\mathscr{M}|\mathscr{M}^*)$, there exists a model $M_2 \in \mathscr{M}^*$ that dominates M_1. Since $M_1 \in \mathscr{M}_M$, there is a positive probability $p > 0$ of selecting M_1 according to $D(M)$. Consider the mixed model M', construed according to the distribution $D(M)$, while replacing M_1 (if selected) with a model selected according to $D(M_2)$. By proposition 1, we get $\mathrm{Cost}_{\ell_1,\ell_2}(M') = \mathrm{Cost}_{\ell_1,\ell_2}(M) - p \cdot (\mathrm{Cost}_{\ell_1,\ell_2}(M_1) - \mathrm{Cost}_{\ell_1,\ell_2}(M_2)) < \mathrm{Cost}_{\ell_1,\ell_2}(M)$, as required.

Second stage: Let $M^{*\prime\prime}$ be the model constructed in the first stage. By an additional iterative process, we construct a model $M^{*\prime}$ that dominates $M^{*\prime\prime}$, while being supported by only two models in $\mathrm{Par}(\mathscr{M}|\mathscr{M}^*)$. In each iteration we have a model M, such that $\mathscr{M}_M \subseteq \mathrm{Par}(\mathscr{M}|\mathscr{M}^*)$. If $|\mathscr{M}_M| > 2$, we will show in the following the existence of a model M' that dominates M, $\mathscr{M}_{M'} \subseteq \mathrm{Par}(\mathscr{M}|\mathscr{M}^*)$, and $|M_{M'}| < |M_M|$. Thus, upon convergence of the iterative process, we will have a model $M^{*\prime}$ that dominates $M^{*\prime\prime}$ (and therefore dominates M^*) and is supported by only two models from $\mathrm{Par}(\mathscr{M}|\mathscr{M}^*)$.

Let M be a model such that $\mathscr{M}_M \subseteq \mathrm{Par}(\mathscr{M}|\mathscr{M}^*)^*$ and $|\mathscr{M}_M| > 2$. Let $M_1 \in \mathscr{M}_M$ be the model that minimizes Cost_{ℓ_1} in $\mathscr{M}_M$ and $\mathscr{M}_2$ be the model that maximizes Cost_{ℓ_1} in $\mathscr{M}_M$. By our assumption, there exists a third model $M_3 \in \mathscr{M}_M \setminus \{M_1, M_2\}$ such that $\mathrm{Cost}_{\ell_1}(M_3) \in [\mathrm{Cost}_{\ell_1}(M_1), \mathrm{Cost}_{\ell_1}(M_2)]$. Let $\beta \in [0,1]$ be the solution for the following equation, $\mathrm{Cost}_{\ell_1}(M_3) = \beta \cdot \mathrm{Cost}_{\ell_1}(M_1) + (1-\beta) \cdot \mathrm{Cost}_{\ell_1}(M_2)$. We construct M' as follows.

Let p_1, p_2, p_3 the probabilities of choosing M_1, M_2, M_3 according to $D(M)$ and let $p = \min\left\{\frac{p_1}{\beta}, \frac{p_2}{1-\beta}\right\}$. We define $D(M')$ to be the same as $D(M)$, except that the probabilities of selecting M_1, M_2, M_3 according to $D(M')$ are $p_1 - \beta p$, $p_2 - (1-\beta)p$ and $p_3 + p$, respectively.

We observe that by the choice of p there are only two options: $p_1 - \beta p = 0$ and thus $M_1 \notin \mathscr{M}_{M'}$; or $p_2 - (1-\beta)p = 0$ and thus $M_2 \notin \mathscr{M}_{M'}$. Since the support of M' does not include models outside of $\mathscr{M}_M$, we get that $|\mathscr{M}_{M'}| < |\mathscr{M}_M|$.

In order to show that M' dominates M, we define $M_{1,2}$ to be the mixed model that selects M_1 with probability β and M_2 with probability $1-\beta$. We first show that $M_{1,2}$ is dominated by M_3. By Proposition 1 and our choice of β, $\mathrm{Cost}_{\ell_1}(M_3) = \mathrm{Cost}_{\ell_1}(M_{1,2})$. Thus, there is a dominance relationship between M_3 and $M_{1,2}$ that is determined by Cost_{ℓ_2}. It follows that since $M_3 \in \mathrm{Par}(\mathscr{M}|\mathscr{M}^*)$, it cannot be dominated by $M_{1,2}$ and hence M_3 dominates $M_{1,2}$.

Finally, following Proposition 1, $\mathrm{Cost}_{\ell_1,\ell_2}(M') = \mathrm{Cost}_{\ell_1,\ell_2}(M) - \beta p \mathrm{Cost}_{\ell_1,\ell_2}(M_1) - (1-\beta)p\mathrm{Cost}_{\ell_1,\ell_2}(M_2) + p\mathrm{Cost}_{\ell_1,\ell_2}(M_3) = \mathrm{Cost}_{\ell_1,\ell_2}(M) - p\mathrm{Cost}_{\ell_1,\ell_2}(M_{1,2}) + p\mathrm{Cost}_{\ell_1,\ell_2}(M_3) \leq \mathrm{Cost}_{\ell_1,\ell_2}(M)$. That is, M' dominates M as required. □

A.2 Proof of Theorem 1

Theorem (1, restated). $\mathscr{M}_{\mathrm{stoch}} = \mathrm{Par}(\mathscr{M}|\mathscr{M}^*)$.

Proof. The proof of $\mathscr{M}_{\mathrm{stoch}} \subseteq \mathrm{Par}(\mathscr{M}|\mathscr{M}^*)$ is given in Subsection 4.2.1. We now prove that $\mathrm{Par}(\mathscr{M}|\mathscr{M}^*) \subseteq \mathscr{M}_{\mathrm{stoch}}$.

Let $M \in \mathrm{Par}(\mathscr{M}|\mathscr{M}^*)$. For proving $M \in \mathscr{M}_{\mathrm{stoch}}$ we show the existence of a parameter α such that M minimizes $\mathrm{Cost}_{\ell_\alpha}$. We first define two parameters α_1, α_2 and set α to be their average. Given two models M_1, M_2 let α_{M_1,M_2} be the parameter that satisfies $\mathrm{Cost}_{\ell_\alpha}(M_1) = \mathrm{Cost}_{\ell_\alpha}(M_2)$. Formally,

$$\alpha_{M_1,M_2} = \frac{\mathrm{Cost}_{\ell_2}(M_1) - \mathrm{Cost}_{\ell_2}(M_2)}{\mathrm{Cost}_{\ell_2}(M_1) - \mathrm{Cost}_{\ell_2}(M_2) - \mathrm{Cost}_{\ell_1}(M_1) + \mathrm{Cost}_{\ell_1}(M_2)} \ .$$

One can verify that when $\mathrm{Cost}_{\ell_1,\ell_2}(M_1) \not> \mathrm{Cost}_{\ell_1,\ell_2}(M_2)$ and $\mathrm{Cost}_{\ell_1,\ell_2}(M_1) \not< \mathrm{Cost}_{\ell_1,\ell_2}(M_2)$, then $\alpha_{M_1,M_2} \in [0,1]$ and $\vec{\alpha}_{M_1,M_2} \cdot \mathrm{Cost}_{\ell_1,\ell_2}(M_1) = \vec{\alpha}_{M_1,M_2} \cdot \mathrm{Cost}_{\ell_1,\ell_2}(M_2)$ (recall that $\vec{\alpha} = (\alpha, 1-\alpha)$).

Using the notion of α_{M_1,M_2} we now ready to define α_1, α_2 as $\alpha_1 = \max_{M' \in \mathscr{M}: \mathrm{Cost}_{\ell_1}(M') > \mathrm{Cost}_{\ell_1}(M)} \{\alpha_{M,M'}\}$ and $\alpha_2 = \min_{M' \in \mathscr{M}: \mathrm{Cost}_{\ell_2}(M') > \mathrm{Cost}_{\ell_2}(M)} \{\alpha_{M,M'}\}$. Let $\alpha = \frac{\alpha_1+\alpha_2}{2}$.

In order to show that M minimize $\mathrm{Cost}_{\ell_\alpha}$ we first show that $\alpha_1 \leq \alpha_2$.

Let $M_1 = \mathrm{argmax}_{M' \in \mathscr{M}: \mathrm{Cost}_{\ell_1}(M') > \mathrm{Cost}_{\ell_1}(M)} \{\alpha_{M,M'}\}$ and $M_2 = \mathrm{argmin}_{M' \in \mathscr{M}: \mathrm{Cost}_{\ell_2}(M') > \mathrm{Cost}_{\ell_2}(M)} \{\alpha_{M,M'}\}$.

Let $\beta = \frac{\mathrm{Cost}_{\ell_1}(M) - \mathrm{Cost}_{\ell_1}(M_2)}{\mathrm{Cost}_{\ell_1}(M_1) - \mathrm{Cost}_{\ell_1}(M_2)}$ and M^* be the model that with probability β answers according to M_1 and with probability $1-\beta$ answers according to M_2. By Proposition 1, and following the definition of β,

$$\mathrm{Cost}_{\ell_1}(M^*) = \beta \cdot \mathrm{Cost}_{\ell_1}(M_1) + (1-\beta) \cdot \mathrm{Cost}_{\ell_1}(M_2) = \mathrm{Cost}_{\ell_1}(M) \ .$$

Since $M \in \mathrm{Par}(\mathscr{M}|\mathscr{M}^*)$, $\mathrm{Cost}_{\ell_1,\ell_2}(M) \not> \mathrm{Cost}_{\ell_1,\ell_2}(M^*)$, and thus $\mathrm{Cost}_{\ell_2}(M) \leq \mathrm{Cost}_{\ell_2}(M^*)$. Using Proposition 1, $\mathrm{Cost}_{\ell_2}(M) \leq \beta \cdot \mathrm{Cost}_{\ell_2}(M_1) + (1-\beta)\mathrm{Cost}_{\ell_2}(M_1)$. i.e.

$$\begin{aligned}\mathrm{Cost}_{\ell_2}(M) \leq \ & \frac{\mathrm{Cost}_{\ell_1}(M) - \mathrm{Cost}_{\ell_1}(M_2)}{\mathrm{Cost}_{\ell_1}(M_1) - \mathrm{Cost}_{\ell_1}(M_2)} \cdot \mathrm{Cost}_{\ell_2}(M_1) \\ & + \frac{\mathrm{Cost}_{\ell_1}(M_1) - \mathrm{Cost}_{\ell_1}(M)}{\mathrm{Cost}_{\ell_1}(M_1) - \mathrm{Cost}_{\ell_1}(M_2)} \cdot \mathrm{Cost}_{\ell_2}(M_2) \ .\end{aligned}$$

Rearranging the terms,

$$\begin{aligned} & \frac{\mathrm{Cost}_{\ell_2}(M) - \mathrm{Cost}_{\ell_2}(M_1)}{\mathrm{Cost}_{\ell_2}(M) - \mathrm{Cost}_{\ell_2}(M_1) - \mathrm{Cost}_{\ell_1}(M) + \mathrm{Cost}_{\ell_1}(M_1)} \\ \leq \ & \frac{\mathrm{Cost}_{\ell_2}(M) - \mathrm{Cost}_{\ell_2}(M_2)}{\mathrm{Cost}_{\ell_2}(M) - \mathrm{Cost}_{\ell_2}(M_2) - \mathrm{Cost}_{\ell_1}(M) + \mathrm{Cost}_{\ell_1}(M_2)} \ , \end{aligned}$$

i.e. $\alpha_1 \leq \alpha_2$.

We finally show that for any $M' \in \mathscr{M}^*$, $\mathrm{Cost}_{\ell_\alpha}(M) \leq \mathrm{Cost}_{\ell_\alpha}(M')$, thus M minimizes $\mathrm{Cost}_{\ell_\alpha}$, and as such, $M \in \mathscr{M}_{\mathrm{stoch}}$. By proposition 3, it is sufficient to show that $\vec{\alpha} \cdot \mathrm{Cost}_{\ell_1,\ell_2}(M) \leq \vec{\alpha} \cdot \mathrm{Cost}_{\ell_1,\ell_2}(M')$.

Since $M \in \mathrm{Par}(\mathscr{M}|\mathscr{M}^*)$ there are three options for $\mathrm{Cost}_{\ell_1,\ell_2}(M')$:

(1) $\mathrm{Cost}_{\ell_1}(M) \leq \mathrm{Cost}_{\ell_1}(M')$ *and* $\mathrm{Cost}_{\ell_2}(M) \leq \mathrm{Cost}_{\ell_2}(M')$*:* In this case,

$$\begin{aligned} & \vec{\alpha} \cdot \mathrm{Cost}_{\ell_1,\ell_2}(M) = \ \alpha \cdot \mathrm{Cost}_{\ell_1}(M) + (1-\alpha) \cdot \mathrm{Cost}_{\ell_2}(M) \\ \leq \ & \alpha \cdot \mathrm{Cost}_{\ell_1}(M') + (1-\alpha) \cdot \mathrm{Cost}_{\ell_2}(M') = \ \vec{\alpha} \cdot \mathrm{Cost}_{\ell_1,\ell_2}(M') \ . \end{aligned}$$

(2) $\mathrm{Cost}_{\ell_1}(M) \leq \mathrm{Cost}_{\ell_1}(M')$ *and* $\mathrm{Cost}_{\ell_2}(M) > \mathrm{Cost}_{\ell_2}(M')$ In this case $\alpha_{M,M'} \leq \alpha_1 \leq \alpha$, and hence,

$$\begin{aligned} & \vec{\alpha} \cdot \mathrm{Cost}_{\ell_1,\ell_2}(M) \\ = \ & \vec{\alpha}_{M,M'} \cdot \mathrm{Cost}_{\ell_1,\ell_2}(M) + (\alpha - \alpha_{M,M'})(\mathrm{Cost}_{\ell_1}(M) - \mathrm{Cost}_{\ell_2}(M)) \\ = \ & \vec{\alpha}_{M,M'} \cdot \mathrm{Cost}_{\ell_1,\ell_2}(M') + (\alpha - \alpha_{M,M'})(\mathrm{Cost}_{\ell_1}(M) - \mathrm{Cost}_{\ell_2}(M)) \\ \leq \ & \vec{\alpha}_{M,M'} \cdot \mathrm{Cost}_{\ell_1,\ell_2}(M') + (\alpha - \alpha_{M,M'})(\mathrm{Cost}_{\ell_1}(M') - \mathrm{Cost}_{\ell_2}(M')) \\ = \ & \vec{\alpha} \cdot \mathrm{Cost}_{\ell_1,\ell_2}(M') \end{aligned}$$

(3) $\mathrm{Cost}_{\ell_1}(M) > \mathrm{Cost}_{\ell_1}(M')$ *and* $\mathrm{Cost}_{\ell_2}(M) \leq \mathrm{Cost}_{\ell_2}(M')$ In this case $\alpha_{M,M'} \geq \alpha_2 \geq \alpha$, and the same inequalities as in Item 2 hold.

The result follows. □

REFERENCES

[1] Deepak Agarwal, Bee-Chung Chen, Pradheep Elango, and Xuanhui Wang. 2011. Click Shaping to Optimize Multiple Objectives. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '11)*. ACM, New York, NY, USA, 132–140. https://doi.org/10.1145/2020408.2020435

[2] Omar Alonso and Stefano Mizzaro. 2009. Relevance criteria for e-commerce: a crowdsourcing-based experimental analysis. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*. ACM, 760–761.

[3] Christopher JC Burges. 2010. From ranknet to lambdarank to lambdamart: An overview. *Learning* 11, 23-581 (2010), 81.

[4] Christopher J Burges, Robert Ragno, and Quoc V Le. 2007. Learning to rank with nonsmooth cost functions. In *Advances in neural information processing systems*. 193–200.

[5] 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.

[6] Tianqi Chen and Carlos Guestrin. 2016. XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16)*. ACM, New York, NY, USA, 785–794. https://doi.org/10.1145/2939672.2939785

[7] Na Dai, Milad Shokouhi, and Brian D. Davison. 2011. Learning to Rank for Freshness and Relevance. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '11)*. ACM, 95–104. https://doi.org/10.1145/2009916.2009933

[8] Onkar Dalal, Srinivasan H. Sengemedu, and Subhajit Sanyal. 2012. Multi-objective Ranking of Comments on Web. In *Proceedings of the 21st International Conference on World Wide Web (WWW '12)*. ACM, 419–428. https://doi.org/10.1145/2187836.2187894

[9] Anlei Dong, Yi Chang, Zhaohui Zheng, Gilad Mishne, Jing Bai, Ruiqiang Zhang, Karolina Buchner, Ciya Liao, and Fernando Diaz. 2010. Towards Recency Ranking in Web Search. In *Proceedings of the Third ACM International Conference on Web Search and Data Mining (WSDM '10)*. ACM, 11–20. https://doi.org/10.1145/1718487.1718490

[10] Ido Guy. 2016. Searching by Talking: Analysis of Voice Queries on Mobile Web Search. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '16)*. ACM, New York, NY, USA, 35–44. https://doi.org/10.1145/2911451.2911525

[11] C-L Hwang and Abu Syed Md Masud. 2012. *Multiple objective decision making—methods and applications: a state-of-the-art survey*. Vol. 164. Springer Science & Business Media.

[12] 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 *Proceedings of the 1st International Workshop on Generalization in Information Retrieval (GLARE 2018)*. http://glare2018.dei.unipd.it/paper/glare2018-paper5.pdf

[13] Changsung Kang, Xuanhui Wang, Yi Chang, and Belle Tseng. 2012. Learning to rank with multi-aspect relevance for vertical search. In *Proceedings of the fifth ACM international conference on Web search and data mining*. ACM, 453–462.

[14] Shubhra Kanti Karmaker Santu, Parikshit Sondhi, and ChengXiang Zhai. 2017. On Application of Learning to Rank for E-Commerce Search. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '17)*. ACM, 475–484. https://doi.org/10.1145/3077136.3080838

[15] Rohan Kumar, Mohit Kumar, Neil Shah, and Christos Faloutsos. 2018. Did We Get It Right? Predicting Query Performance in E-commerce Search. In *SIGIR 2018 Workshop on eCommerce (ECOM18)*.

[16] Xiao Lin, Hongjie Chen, Changhua Pei, Fei Sun, Xuanji Xiao, Hanxiao Sun, Yongfeng Zhang, Wenwu Ou, and Peng Jiang. 2019. A Pareto-Efficient Algorithm for Multiple Objective Optimization in e-Commerce Recommendation. In *Proceedings of the 13th ACM Conference on Recommender Systems (RecSys '19)*. Association for Computing Machinery, 20–28. https://doi.org/10.1145/3298689.3346998

[17] Tie-Yan Liu. 2009. Learning to Rank for Information Retrieval. *Found. Trends Inf. Retr.* 3, 3 (March 2009), 225–331. https://doi.org/10.1561/1500000016

[18] Bo Long, Jiang Bian, Anlei Dong, and Yi Chang. 2012. Enhancing product search by best-selling prediction in e-commerce. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management (CIKM '12)*. ACM, 2479–2482.

[19] Yoelle Maarek. 2018. Alexa and Her Shopping Journey. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management (CIKM '18)*. ACM, 1–1. https://doi.org/10.1145/3269206.3272923

[20] R Timothy Marler and Jasbir S Arora. 2004. Survey of multi-objective optimization methods for engineering. *Structural and multidisciplinary optimization* 26, 6 (2004), 369–395.

[21] Tomas Mikolov, Edouard Grave, Piotr Bojanowski, Christian Puhrsch, and Armand Joulin. 2018. Advances in Pre-Training Distributed Word Representations. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*.

[22] Michinari Momma, Alireza Bagheri Garakani, and Yi Sun. 2019. Multi-objective Relevance Ranking. In *Proceedings of ACM SIGIR Workshop on eCommerce (SIGIR 2019 eCom)*.

[23] Phong Nguyen, John Dines, and Jan Krasnodebski. 2017. A multi-objective learning to re-rank approach to optimize online marketplaces for multiple stakeholders. *arXiv preprint arXiv:1708.00651* (2017).

[24] Mario Rodriguez, Christian Posse, and Ethan Zhang. 2012. Multiple Objective Optimization in Recommender Systems. In *Proceedings of the Sixth ACM Conference on Recommender Systems (RecSys '12)*. ACM, 11–18. https://doi.org/10.1145/2365952.2365961

[25] Ning Su, Jiyin He, Yiqun Liu, Min Zhang, and Shaoping Ma. 2018. User Intent, Behaviour, and Perceived Satisfaction in Product Search. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining (WSDM '18)*. ACM, 547–555. https://doi.org/10.1145/3159652.3159714

[26] Krysta M. Svore, Maksims N. Volkovs, and Christopher J.C. Burges. 2011. Learning to Rank with Multiple Objective Functions. In *Proceedings of the 20th International Conference on World Wide Web (WWW '11)*. ACM, 367–376. https://doi.org/10.1145/1963405.1963459

[27] Lidan Wang, Paul N. Bennett, and Kevyn Collins-Thompson. 2012. Robust Ranking Models via Risk-sensitive Optimization. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '12)*. ACM, 761–770. https://doi.org/10.1145/2348283.2348385

[28] Liang Wu, Diane Hu, Liangjie Hong, and Huan Liu. 2018. Turning Clicks into Purchases: Revenue Optimization for Product Search in E-Commerce. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR '18)*. ACM, 365–374. https://doi.org/10.1145/3209978.3209993