A Ranked Bandit Approach for Multi-stakeholder Recommender Systems*

TAHEREH ARABGHALIZI, University of Pittsburgh, USA
ALEXANDROS LABRINIDIS, University of Pittsburgh, USA

Recommender systems traditionally find the most relevant products or services for users tailored to their needs or interests but they ignore the interests of the other sides of the market (aka stakeholders). In this paper, we propose to use a Ranked Bandit approach for an online multi-stakeholder recommender system that sequentially selects top k items according to the relevance and priority of all the involved stakeholders. We presented three different criteria to consider the priority of each stakeholder when evaluating our approach. Our extensive experimental results on a movie dataset showed that the contextual multi-armed bandits with a relevance function make a higher level of satisfaction for all involved stakeholders in the long term.
Keywords: Multi-stakeholder Recommender Systems; Multi-armed Bandits; Ranked Bandit;

CCS Concepts: • **Information systems** → **Recommender systems**.

1 INTRODUCTION

With information technology permeating all aspects of our modern life, recommender systems have become an integral part of personalizing the user experience. Conventional recommender systems only consider the needs and interests of the end users and overlook the preferences of the other sides of the marketplace, also known as stakeholders, who might benefit from the recommendation selections.

1.1 Motivating Example

In this paper, we address the problem of how to best recommend coupons, offered by local businesses such as coffee shops, to passengers awaiting the arrival of a bus at a nearby bus stop. The coupons are offered when the next bus is expected to be full in order to encourage the bus passengers to take advantage of the recommended incentive rather than taking a full bus [3]. The goal of this coupon recommender system is to take the preferences of all involved stakeholders into account and be able to prioritize different stakeholders at different times. It is worth noting that the preferences of local businesses towards the bus passengers are based on their marketing purposes (e.g., special coupons for college students).
Although our motivational application targets people arriving at bus stops, our proposed solution can be used in a wide range of applications in which people receive recommendations and there are multiple stakeholders (beyond just the people) who have preferences that need to be considered.

1.2 Background and Related Work

A traditional user-centric recommender system recommends items based on the interests and preferences of the user. However, the user is not the sole stakeholder in many real-world applications. In those cases, the recommendations could benefit other individuals or organizations [1, 2].
Moreover, online recommendation tasks, which ingest data one observation at a time, are not well served by traditional offline recommendation techniques such as Collaborative Filtering [17] which rely on historical data. It has recently come to the attention of online recommendation tasks to study multi-armed bandits (MAB), a classic reinforcement learning problem. As a result of this

Presented at the MORS workshop held in conjunction with the 16th ACM Conference on Recommender Systems (RecSys), 2022, in Seattle, USA.

approach, a solution can be provided for the dilemma between exploration and exploitation that maximizes the expected cumulative payoff over the long term [5].
Depending on whether side information (aka context) is taken into account, bandit algorithms fall into two categories: *context-free* and *contextual*. In context-free bandits, the observed reward depends on the selected arm while it is both the selected arm and its context that determine the observed reward in contextual bandits [13]. While contextual bandits have been used extensively for online user-centric recommender systems [11, 20] and much research has been conducted on multi-objective multi-armed bandit algorithms [7, 21, 23], multi-stakeholder resommender systems have received less attention. Most recently, Mehrotra et al. [14] proposed a multiple-objective contextual bandit approach to maximize the long-term payoffs for different objectives such as diversity for a multi-stakeholder recommender system on a music streaming platform. Although, there are several existing works that consider multiple stakeholders in their recommendation generation [15, 19], to the best of our knowledge, there is no work that considers the contextual information of users, preferences and priority of all stakeholders in an online recommender system.

1.3 Contributions

In this work, we aim to provide long-term, acceptable levels of satisfaction for all the stakeholders involved in a recommender system, according to the given context (if available) and priorities. We propose a bandit-based approach that sequentially selects top candidate items and then accepts the best candidates that maximize the total payoff, given relevance and priority of each stakeholder. Our contributions are as follows:

- we propose to use a Ranked Bandit approach to recommend multiple items to a user and provide acceptable levels of satisfaction for all stakeholders in a recommender system over time.
- we introduce three different criteria including Deterministic, Probabilistic and Multi-sided relevance Function to consider the priority of each stakeholder when evaluating our approach.
- we define and use different metrics to evaluate the satisfaction of stakeholders and compare different benchmarks.
- we use real-world and synthetic datasets and multiple sensitivity analyses to experimentally evaluate the performance of our approach.

2 PROBLEM STATEMENT

In this paper, we address *the problem of recommending items to users in an online multi-stakeholder platform, considering the contextual information of users and items (if available), relevance and priority of involved stakeholder in order to provide them with an acceptable level of satisfaction in the long term.* In the application of coupon recommendation, the recommender system needs to recommend top k coupons (from the finite set of available coupons at each bus stop) to a bus passenger whose up-coming bus is going to be full. This system should take the preferences of all stakeholders such as bus passengers, coupon suppliers and minority-owned businesses (i.e., another stakeholder group that we want to prioritize) into account and offer the relevant coupons based on the given priority (weight) of stakeholders. This recommender system aims to maximize the total payoffs and makes a good balance between the satisfaction of all stakeholders over time.

3 PROPOSED SOLUTION

To address this problem, we propose to utilize a Ranked Bandit approach that sequentially selects top k items and accepts the best candidates that maximize the payoff, given relevance and priority of each stakeholder.

In *Ranked Bandit* algorithm, which was first introduced by [16, 18], one multi-armed bandit algorithm is instantiated for each slot of a ranked list with k slots, to learn the greedy-optimal solution. If a user clicks on slot i, that slot gains a reward of 1, and all other slots j where $j < i$ receive a reward of 0. In a multi-stakeholder platform, however, the algorithm should consider the relevance and priority (if applicable) of all involved stakeholders to obtain a reward of 1 for each slot. To this end, we introduce three different criteria including deterministic, probabilistic, and multi-sided relevance function (see Section 4.1), apply them after displaying the selected items to a user and receive a reward accordingly.

Our proposed approach is described in Algorithm 1. As you can see, at each round, the Ranked Bandit algorithm plays a multi-armed bandit instance M_i for each ranking slot i. If the arm s_i was already selected at a higher ranking slot, an arbitrary unselected arm is chosen instead. This process is being sequentially repeated for top k ranking slot (Lines 3-9). Then the algorithm receive a feedback for each multi-armed bandit instance M_i through running a criterion. If the arm in ranking slot i is relevant to the involved stakeholders based on the used criterion and the given priority (weight) of stakeholders, then bandit M_i receives a reward of 1 and all higher bandits M_j ($j < i$) receive a reward of 0 (Lines 12-17). Each bandit instance updates its reward afterwards.

Algorithm 1 Ranked Bandit for Multi-stakeholder Recommender Systems

1: **Inputs**: M: base bandits, A: arms, T: rounds, k: number of slots, C_u: user context (if any), F_a: arm features (if any), C($w_1, w_2, ..., w_n$): criterion (Deterministic, Probabilistic or Multi-sided relevance function) where w_n is the weight of the n^{th} stakeholder.
2: **for** $t = 1, 2, \ldots, T$ **do**
3: **for** $i = 1, 2, \ldots, k$ **do**
4: $s_i(t) \leftarrow M_i$.selectArm(A, u_t, C_{u_t}, F_{a_t})
5: **if** $s_i(t) \in \{s_1(t), s_2(t), ..., s_{i-1}(t)\}$ **then**
6: $s_i(t) \leftarrow$ *arbitrary unselected arm*
7: **end if**
8: $S(t) \leftarrow \bigcup_i s_i(t)$
9: **end for**
10: Display $S(t)$ to user and receive feedback for M_i:
11: **for** $i = 1, 2, \ldots, k$ **do**
12: Apply criterion C($w_1, w_2, ..., w_n$)
13: **if** $s_i(t)$ is the first relevant arm to stakeholders (based on the used criterion) **then**
14: $reward_{s_i(t)} = 1$
15: **else**
16: $reward_{s_i(t)} = 0$
17: **end if**
18: M_i.UpdateReward(u_t, C_{u_t}, F_{a_t}, $s_i(t)$, $reward_{s_i(t)}$)
19: **end for**
20: **end for**

4 EVALUATION

The training process in online algorithms, such as bandits, occurs incrementally over time. However, offline evaluation methods can be used to evaluate the performance of such algorithms using historical datasets. One of the most popular offline evaluation methods is called *Replay*. In this methodology, the bandit algorithm selects an arm for each record in the historical data. If this arm has been seen by the user before, the replay methodology accepts the arm, otherwise the arm is denied by the replay method [12]. This method works well in offline evaluation of bandits

employed in the user-centric recommender systems which focus on user satisfaction obtained based on user clicks. In a multi-stakeholder recommender system such as coupon recommendation, however, the replay evaluation should be able to consider the relevance of a selected arm to multiple stakeholders given the priority of those stakeholders. To this end, we propose three criteria called *Deterministic, Probabilistic, and Multi-sided Relevance Function* to address the relevance of a selected arm to multiple weighted stakeholders in the replay evaluation.

4.1 Selection/Evaluation Criteria

Deterministic: based on this criterion, the replay method accepts an arm if it is relevant to the most prioritized stakeholder which is the one with the highest weight.
Probabilistic: this criterion determines the winner arm by first partitioning the [0,1] interval to different ranges according to the weights of the different stakeholders and then selecting a random number ∈ [0,1] which would map to one of the ranges and therefore one of different stakeholders. The replay method would then accept the arm which is relevant to that stakeholder.
Multi-sided Relevance Function: the multi-sided relevance function can be defined as the weighted sum of the relevance score of each stakeholder to each selected arm a. The relevance scores can be calculated based on the feedback of users and other stakeholders about the quality of the recommended items. The relevance function is defined as below:

$$r^a(w, s) = w_1 s_1^a + ... + w_n s_n^a \tag{1}$$

where $r^a \in [0, 1]$ is the estimated multi-sided relevance score, n is the number of stakeholders, $s_1^a, s_2^a, ..., s_n^a$ are the relevance scores of arm a for each stakeholder where each score is either 1 (relevant) or 0 (non-relevant), w_i is the given priority or weight of each stakeholder where $w_i \in [0, 1]$, $\sum_{i=1}^{n} w_i = 1$. We also define a relevance threshold, δ, to let the replay evaluation accepts an arm if the multi-sided relevance score is greater than this threshold [4].

4.2 Evaluation Metrics

We use three different types of metrics to evaluate the performance of our proposed approach:

- **Average Rewards**: we accumulate the reward values of the accepted arms in each round and return the average of the rewards for total rounds. The ultimate goal of a bandit algorithm is to maximize the rewards in the long term.
- **Average NDCG** : Normalized Discounted Cumulative Gain (NDCG) measures the usefulness of an item based on its position in the recommendation list [9]. We compute the Average NDCG at k for each stakeholder over time as the representation of their satisfaction level.
- **Balance-Relevance Rate**: this metric computes the product of the balance and relevance rates which are defined as follows:
  - **Balance Rate**: to measure how similar the weights of the stakeholders and their satisfaction levels (i.e. Average NDCG) are, we compute the cosine similarity between the vector of weights and their corresponding Average NDCG.
  - **Relevance Rate**: this is a measure that calculates the number of times (out of total rounds) that the selected arms are relevant to stakeholders (using one of the evaluation criteria) and are accepted by the replay evaluation.

5 EXPERIMENTAL EVALUATION

In this section, we conduct several experiments on a movie recommendation data to evaluate the performance of our proposed approach.

5.1 Dataset

Since there are currently no public datasets available for a multi-stakeholder platform in the coupon recommendation domain, we merged two real-world datasets including MovieLens (1m) [8] with IMDB (81k+) [10] and also generated a synthetic dataset. In our experiments, we use a scenario where there are three stakeholders involved in the coupon recommendation system. According to the data, movies are used as coupons, users as bus passengers (first stakeholder), movie production companies as local businesses/coupon suppliers (second stakeholder) and movies with a specific genre as a minority-owned business (third stakeholder). We created the following datasets for training and offline evaluation of the bandit algorithms:
Movies: we use top m movies with the highest number of ratings as arms.
Movies' Features: the genres of movies are used as the context information of the arms.
Users' Features: users' demographic features such as age range and gender along with the the normalized average of the genres of the movies that each user rated, are used as the context information of the users.
Users Data: the ratings of the users to the movies (integers between 1 and 5) are used as the feedback of users in bandit evaluation. If the rating of a selected movie is greater than 3.0, the movie is likely relevant to the preferences of that user and they will click on it. It should be noted that we used *Singular Value Decomposition (SVD)* technique to fill in the missing data in the user ratings.
Suppliers Data: as there aren't any ratings data for production companies towards the users in the MovieLens or IMDB datasets, we generated a synthetic dataset. First we clustered all users using *KMeans* algorithm and then for each production company, we used a *Truncated Normal Distribution* with mean = 3 and standard deviation = 1, to generate random integer numbers between 1 and 5 as the ratings of that production company towards the users in the same cluster. If a production company has given a rating greater than 3.0 to the current user, that user is likely relevant to that company's preferences.
Minority Data: we consider 'Sci-Fi' movies as minority-owned businesses (since only about 7% of all movies in the MovieLens data are 'Sci-Fi'). If a selected movie is Sci-Fi, it is assumed to be relevant to minority-owned businesses.
Observation Data: we use a stream of users who rated the top movies (arms) in the past to evaluate the bandits.

5.2 Base Bandit Algorithms

In this section, we describe the base multi-armed bandit algorithms that are used in our approach:

- **Hybrid LinUCB**: as mentioned earlier, contextual bandits is a variant of the multi-armed bandit problems that utilizes contextual information. In many applications, the arms are not distinct from each other and have shared features. For example, in a coupon recommendation system where the coupons are arms that are recommended to users, there may be some similarities between coupons, and thus they share similar features. Li et al. [11] introduced LinUCB with Hybrid Linear Model where the payoff of each arm is a linear function of shared and non-shared components. We use the contextual information of users along with the arms features for this algorithm. LinUCB has a hyper-parameter, α, which should be tuned in advance.
- **Thompson Sampling**: it is a Bayesian multi-armed bandit algorithm where a posterior distribution is used to summarize reward values inferred from past data. Under this distribution, the algorithm selects the arm proportionately to its likelihood of being optimal [22].

- **Epsilon-greedy**: in each round, this algorithm selects a random arm with probability ϵ, and chooses the arm with the highest empirical mean with probability $1 - \epsilon$ [6]. ϵ is a hyper-parameter and needs to be tuned properly.
- **UCB1**: an upper confidence bound has to be calculated for each arm for the algorithm to be able to choose an arm in each round [5].
- **Random**: it randomly selects an arm to pull at each round.

5.3 Experimental Results

In this section we present and discuss the results of our experiments.

Results with Different Criteria: we conducted multiple experiments for all base bandit algorithm with default settings (see Table 1) to compare the impact of our proposed evaluation criteria in terms of Balance-Relevance Rate. As you can see in Table 2, where we show the results for the Hybrid LinUCB algorithm, the multi-sided relevance function outperforms the other two criteria (i.e. Deterministic and Probabilistic) in almost all experiments with different weight combinations. This shows that given the priority of stakeholders, the multi-sided relevance function can provide a better trade-off between the satisfaction of all stakeholders in the long term. For this reason, we carried out the next experiments using the the multi-sided relevance function as the evaluation criterion.

Table 1. Default experimental settings

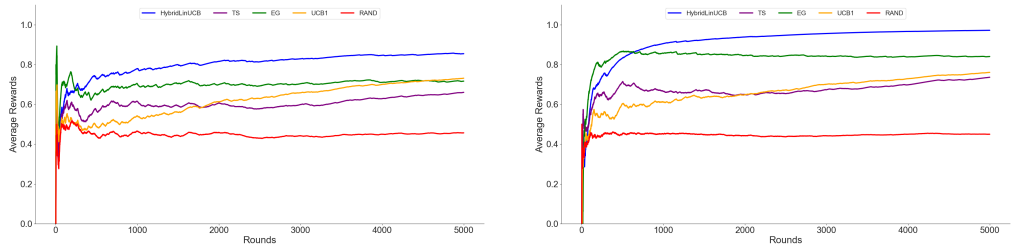
Number of Rounds (T)	5000
Number of Arms ($\|A\|$)	100
Number of Slots (k)	3
Stakeholders' Weights (w_i)	All triplet combinations with 0.6, 0.3, 0.1
Relevance Threshold (δ)	0.5
LinUCB Hyper Parameter (α)	0.5
EG Hyper Parameter (ϵ)	0.15

Table 2. Comparing evaluation criteria for Hybrid LinUCB in terms of balance-relevance rate

Stakeholders' Weights	Prioritized Stakeholder	Deterministic	Probabilistic	Multi-sided Relevance Function
(0.33, 0.33, 0.33)	-	0.442	0.755	**0.900**
(0.6, 0.3, 0.1)	user	0.840	0.725	**0.851**
(0.6, 0.1, 0.3)	user	0.818	0.695	**0.841**
(0.3, 0.6, 0.1)	supplier	0.965	0.789	**0.967**
(0.1, 0.6, 0.3)	supplier	0.917	0.834	**0.921**
(0.3, 0.1, 0.6)	minority	0.967	0.823	**0.967**
(0.1, 0.3, 0.6)	minority	**0.954**	0.862	0.948

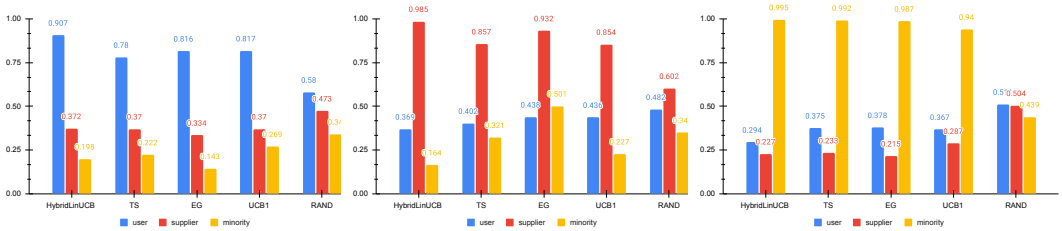
Results with Multi-sided Relevance Function: in this section we describe and illustrate the results of several experiments with the default settings (see Table 1) and compare the base bandit algorithms in terms of average rewards and average NDCG while prioritizing different stakeholders. As it is shown in Figure 1, Hybrid LinUCB has the highest average rewards compared to the other context-free bandit algorithms when we prioritize users (Figure 1a) and suppliers (Figure 1b). This indicates that utilizing the contextual features of users and items can make higher average rewards for all stakeholders when applying the Multi-sided Relevance Function as the evaluation criterion.

It should be mentioned that our results show that the Hybrid LinUCB algorithm outperforms other base bandits when we also prioritize minority (as the third stakeholder).
Furthermore, Figure 2 illustrates the Average NDCG for different stakeholders when running different bandit algorithms with multi-sided relevance function. As one can see, Hybrid LinUCB provides higher Average NDCG for the prioritized stakeholder and makes a good balance rate



(a) weights: (user: 0.6, supplier: 0.3, minority: 0.1) (b) weights: (user: 0.3, supplier: 0.6, minority: 0.1)

Fig. 1. Average rewards for 100 arms and 3 ranking slots when prioritizing users (a) and suppliers (b)



(a) weights: (0.6, 0.3, 0.1) (b) weights: (0.3, 0.6, 0.1) (c) weights: (0.1, 0.3, 0.6)

Fig. 2. Average NDCG when prioritizing users (a), suppliers (b) and minority (c)

between the Average NDCG and the weights of stakeholders, in all scenarios when we prioritize users (a), suppliers (b) or minority (c).
Sensitivity Analysis: we performed multiple sensitivity analyses to see how Balance-Relevance Rate changes with different set of stakeholders' weights, different number of arms and different number of ranking slots:

- Stakeholders' Weights: we set the weights of stakeholders to different values (e.g., 0.5, 0.25, 0.25), where sum is equal to 1. The results of these experiments showed a similar pattern to the the results of experiments with the default weights.
- Number of Arms: we set the number of arms to 100 (default), 200 and 300 for different bandits while prioritizing different stakeholders. As the figures in Table 3 show, Hybrid LinUCB with with multi-sided relevance function outperforms the other bandit algorithms in terms of balance-relevance rate even when we change the number of arms.
- Number of Slots: we change the number of ranking slots (offered items) to 1, 3 (default) and 5 while prioritizing one stakeholder at a time for 100 arms. As the results of these experiments in Table 4 show, Hybrid LinUCB with multi-sided relevance function works better than other algorithms in terms of balance-relevance rate for various number of slots. There are only

two scenario where Thompson Sampling works a little better than Hybrid LinUCB but the difference between their balance-relevance rates is negligible (<0.006). It is worth noting that balance-relevance rates grow when the number of offered items (slots) increases which is reasonable because when the recommender system offers higher number of items to users, there is a higher probability that the offered items are relevant to stakeholders.

Table 3. Comparing bandits in terms of balance-relevance rate when changing number of arms

Stakeholders' Weights	Prioritized Stakeholder	Number of Arms	Hybrid LinUCB	TS	EG	UCB1	RAND
(0.6, 0.3, 0.1)	User	100	**0.851**	0.657	0.715	0.723	0.429
		200	**0.835**	0.527	0.698	0.575	0.348
		300	**0.834**	0.525	0.704	0.466	0.293
(0.3, 0.6, 0.1)	Supplier	100	**0.967**	0.724	0.801	0.758	0.424
		200	**0.930**	0.716	0.821	0.653	0.444
		300	**0.934**	0.678	0.870	0.588	0.434
(0.1, 0.3, 0.6)	Minority	100	**0.948**	0.942	0.888	0.852	0.262
		200	**0.966**	0.955	0.891	0.780	0.250
		300	**0.953**	0.942	0.887	0.694	0.201

Table 4. Comparing bandits in terms of balance-relevance rate when changing number of slots

Stakeholders' Weights	Prioritized Stakeholder	Number of Slots	Hybrid LinUCB	TS	EG	UCB1	RAND
(0.6, 0.3, 0.1)	User	1	**0.808**	0.557	0.593	0.64	0.316
		3	**0.851**	0.657	0.715	0.723	0.429
		5	**0.864**	0.65	0.757	0.732	0.473
(0.3, 0.6, 0.1)	Supplier	1	**0.953**	0.631	0.701	0.664	0.270
		3	**0.967**	0.724	0.801	0.758	0.424
		5	**0.967**	0.741	0.836	0.773	0.466
(0.1, 0.3, 0.6)	Minority	1	0.935	**0.938**	0.84	0.792	0.154
		3	**0.948**	0.942	0.888	0.852	0.262
		5	0.949	**0.955**	0.955	0.857	0.307

6 CONCLUSIONS

In this paper, we addressed the problem of recommending coupons in a multi-stakeholder platform where stakeholders can be prioritized. We proposed to use a Ranked Bandit approach that sequentially selects top k items and accepts the best candidates that maximize the payoff, given relevance and priority of each stakeholder. We introduced three different criteria including Deterministic, Probabilistic and Multi-sided Relevance Function to consider the priority of each stakeholder when evaluating our approach. We also defined different metrics to evaluate the satisfaction of stakeholders and compare different benchmarks. Our experimental results on real-world and synthetic datasets showed that contextual multi-armed bandits (i.e. Hybrid LinUCB) with a multi-sided relevance function outperforms context-free bandits with any evaluation criterion.

ACKNOWLEDGMENTS

This work is part of the PittSmartLiving project which is supported by NSF award CNS-1739413.

REFERENCES

[1] Himan Abdollahpouri, Gediminas Adomavicius, Robin Burke, Ido Guy, Dietmar Jannach, Toshihiro Kamishima, Jan Krasnodebski, and Luiz Pizzato. 2020. Multistakeholder recommendation: Survey and research directions. *User Modeling and User-Adapted Interaction* 30, 1 (2020), 127–158.

[2] Himan Abdollahpouri and Robin Burke. 2022. Multistakeholder recommender systems. In *Recommender systems handbook*. Springer, 647–677.

[3] Tahereh Arabghalizi and Alexandros Labrinidis. 2020. Data-driven bus crowding prediction models using context-specific features. *ACM Transactions on Data Science* 1, 3 (2020), 1–33.

[4] Tahereh Arabghalizi and Alexandros Labrinidis. 2022. Context-aware Multi-stakeholder Recommender Systems. In *The International FLAIRS Conference Proceedings*, Vol. 35.

[5] Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. 2002. Finite-time analysis of the multiarmed bandit problem. *Machine learning* 47, 2 (2002), 235–256.

[6] Nicolo Cesa-Bianchi and Paul Fischer. 1998. Finite-Time Regret Bounds for the Multiarmed Bandit Problem.. In *ICML*, Vol. 98. 100–108.

[7] Madalina M Drugan and Ann Nowe. 2013. Designing multi-objective multi-armed bandits algorithms: A study. In *The 2013 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 1–8.

[8] F Maxwell Harper and Joseph A Konstan. 2015. The movielens datasets: History and context. *Acm transactions on interactive intelligent systems (tiis)* 5, 4 (2015), 1–19.

[9] Kalervo Järvelin and Jaana Kekäläinen. 2002. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems (TOIS)* 20, 4 (2002), 422–446.

[10] Stefano Leone. 2019. IMDb movies extensive dataset. https://www.kaggle.com/stefanoleone992/imdb-extensive-dataset

[11] Lihong Li, Wei Chu, John Langford, and Robert E Schapire. 2010. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*. 661–670.

[12] Lihong Li, Wei Chu, John Langford, and Xuanhui Wang. 2011. Unbiased offline evaluation of contextual-bandit-based news article recommendation algorithms. In *Proc of the 4th ACM intl conference on Web search and data mining*. 297–306.

[13] Tyler Lu, Dávid Pál, and Martin Pál. 2010. Contextual multi-armed bandits. In *Proceedings of the Thirteenth international conference on Artificial Intelligence and Statistics*. JMLR Workshop and Conference Proceedings, 485–492.

[14] Rishabh Mehrotra, Niannan Xue, and Mounia Lalmas. 2020. Bandit based Optimization of Multiple Objectives on a Music Streaming Platform. In *Proc of the 26th ACM SIGKDD Intl Conference on Knowledge Discovery & Data Mining*. 3224–3233.

[15] 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).

[16] Filip Radlinski, Robert Kleinberg, and Thorsten Joachims. 2008. Learning diverse rankings with multi-armed bandits. In *Proceedings of the 25th international conference on Machine learning*. 784–791.

[17] Francesco Ricci, Lior Rokach, and Bracha Shapira. 2011. Introduction to recommender systems handbook. In *Recommender systems handbook*. Springer, 1–35.

[18] Matthew Streeter, Daniel Golovin, and Andreas Krause. 2009. Online learning of assignments. *Advances in neural information processing systems* 22 (2009).

[19] Özge Sürer, Robin Burke, and Edward C Malthouse. 2018. Multistakeholder recommendation with provider constraints. In *Proceedings of the 12th ACM Conference on Recommender Systems*. 54–62.

[20] Liang Tang, Yexi Jiang, Lei Li, and Tao Li. 2014. Ensemble contextual bandits for personalized recommendation. In *Proc. of the 8th ACM Conf. on Recommender Systems*. 73–80.

[21] Cem Tekin and Eralp Turğay. 2018. Multi-objective contextual multi-armed bandit with a dominant objective. *IEEE Transactions on Signal Processing* 66, 14 (2018), 3799–3813.

[22] William R Thompson. 1933. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika* 25, 3-4 (1933), 285–294.

[23] Saba Q Yahyaa, Madalina M Drugan, and Bernard Manderick. 2014. The scalarized multi-objective multi-armed bandit problem: An empirical study of its exploration vs. exploitation tradeoff. In *2014 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2290–2297.