Combining Reward and Rank Signals for Slate Recommendation

Imad Aouali*
ENS Paris-Saclay, Criteo AI Lab
Paris, France

Sergey Ivanov
Criteo AI Lab
Paris, France

Mike Gartrell
Criteo AI Lab
Paris, France

David Rohde
Criteo AI Lab
Paris, France

Flavian Vasile
Criteo AI Lab
Paris, France

Diego Legrand
Criteo AI Lab
Paris, France

ABSTRACT
We consider the problem of slate recommendation, where the recommender system presents a user with a collection or slate composed of $K$ recommended items at once. If the user finds the recommended items appealing then the user may click and the recommender system receives some feedback. Two pieces of information are available to the recommender system: was the slate clicked? (the reward), and if the slate was clicked, which item was clicked? (rank). In this paper, we formulate several Bayesian models that incorporate the reward signal (Reward model), the rank signal (Rank model), or both (Full model), for non-personalized slate recommendation. In our experiments, we analyze performance gains of the Full model and show that it achieves significantly lower error as the number of products in the catalog grows or as the slate size increases.

CCS CONCEPTS
• Information systems → Recommender systems; • Computing methodologies → Maximum a posteriori modeling.

KEYWORDS
Slate Recommendation, Bayesian Inference.

ACM Reference Format:
Imad Aouali, Sergey Ivanov, Mike Gartrell, David Rohde, Flavian Vasile, Victor Zaytsev, and Diego Legrand. 2021. Combining Reward and Rank Signals for Slate Recommendation. In KDD ’21 Workshop on Bayesian Causal Inference for Real World Interactive Systems, August 14th-15th, 2021, Singapore. ACM, New York, NY, USA, 6 pages.

1 INTRODUCTION
Slate recommendation, also referred to as banner recommendation, is the task of recommending a collection of $K$ items at once to the user. This problem arises in many real-world applications like search and online advertising. The logs of the recommender system can be used to refine future recommendations by the use of two distinct signals. First, the reward signal that identifies slates that the user interacts with. For example, if we recommend to a user a slate of two items: phone and couscous, and the user interacts with that recommendation, then the slate receives a reward of 1 (and perhaps the user finds the slate appealing as a whole). Second, the rank signal that describes which item was interacted with within the slate. For example, if we recommend to a user a phone and couscous, and the user clicks on the couscous, then the rank is 2 (the user interacted with the 2nd item, and perhaps prefers it to the first). The rank signal is an item-level information that gives an individual ranking characterizing the score of a click on an item in the slate. Non-personalized slate recommendation algorithm can either use the reward signal (the number of clicks & non-clicks on the slate), the rank signal (number of clicks on each item in a slate), or both to decide which slate to display to the users.

The following is an introductory example for this setting. We consider a catalog containing 3 items: phone, couscous, and beer. Ignoring order, there are 3 possible slates with size 2 that we can recommend: [phone, couscous], [phone, beer] or [phone, couscous]. Using historical data summarizing the interactions with these three slates, we consider how to determine the best slate to display to the user. An example of historical data is given in Table 1, where we show each of the 3 slates 700 times. Here, slate [couscous, beer] is the best one. The most direct evidence for this is that it has the lowest number of non-clicks (626) and hence the highest click through rate $(1 - \frac{626}{700} = 0.11)$. There is also indirect evidence using click rank that couscous is preferred to phone (29 clicks vs. 10), beer is preferred to phone (47 clicks vs. 9), and couscous is preferred to beer (46 clicks vs. 28). In aggregate, this ranking information also suggests that [couscous, beer] is the best slate - this suggestion is conditional upon a modeling assumption that there are not virtuous or counterproductive combinations of items in slates - which we will make rigorous shortly.

*Correspondence to i.aouali@criteo.com.
We consider non-personalized slate recommendation. Interaction which allow evaluation of new slate recommendation policies, as non-decreasing function w.r.t items ranking (e.g. [7]).

### Table 1: Example of slate recommendation historical data.

| Slate      | non-clicks | clicks on 1 | clicks on 2 |
|------------|------------|-------------|-------------|
| phone, couscous | 661        | 10          | 29          |
| phone, beer   | 644        | 9           | 47          |
| couscous, beer| 626        | 46          | 28          |

Bandit algorithms are actively being developed for online slate recommendation. In general, bandit algorithms are provably optimal and have strong theoretical guarantees. In the multi-armed bandits setting [11, 15, 16], algorithms rely on the reward signal only. It has been shown that their performance deteriorates in online slate recommendation as the number of possible slates is combinatorially large [7, 14]. In combinatorial bandits/semi-bandits settings, some studies assume access to the reward signal as a function, with certain properties, of the unknown items ranking [3, 4, 7], and others assume direct access to items ranking [9, 10, 12, 18].

In offline settings, numerous reward modeling approaches have been proposed in the context of slate recommendation. In [17], off-policy evaluation and optimization procedures were developed which allow evaluation of new slate recommendation policies, as well finding the one that achieves maximal reward. In that study, reward signal is assumed to be additive w.r.t unknown items ranking. In [8], authors assume access to items ranking, and use them to train conditional variational autoencoders that models items distribution and enables slates generation for recommendation.

In production, and perhaps surprisingly, practical algorithms often ignore the reward signal and rely on ranking items to learn user preferences. An example of such models is a simple extension of the Pop model in [6], where the agent recommends a slate composed of the top K most popular items. Other examples of relevant work include [1, 5, 13]. In our experience, the reward signal is often ignored in recommender systems since it is difficult to integrate it into the target cost function which is based on ranking items correctly. Another reason can be organisational - different teams in the company might focus either on the reward or on items ranking. Such separation is typical in online advertising - bidding team typically focuses on reward prediction, whereas recommendation team is only interested in ranking candidates correctly.

In this paper, we formulate three intuitive Bayesian models that use either the reward signal (Reward model), the rank signal (Rank model), or both (Full model). These algorithms learn from offline historical data similar to the example presented in Table 1, and allow consistent estimation of the underlying reward model. We demonstrate empirically that the Full model outperforms the other two approaches highlighting the benefits of combining both, the reward and rank signals.

## 2 BAYESIAN FORMULATION OF FULL, REWARD & RANK MODELS

### 2.1 Setting

We consider non-personalized slate recommendation. Interaction between items in a slate is ignored, meaning that the best slate is the one composed of the overall best $K$ items. In addition, the order of items in a slate doesn’t matter, meaning that recommending [item1, item2] is the same as recommending [item2, item1]. The statistics of slate interaction with users is summarized in the following $K + 1$ variables, the number of non-clicks on the slate $nc$, and the number of clicks on each item of the recommended slate of size $K$, which we denote $c_i$ for $i \in [K]$. Other useful variables, which can be derived from the ones defined previously, are the number of clicks on the recommended slate $c = \sum_{i \in [K]} c_i$, and the number of impressions $I = c + nc$. Table 3 in Appendix A summarizes the remaining variables as well as the ones we have already mentioned.

### 2.2 Bayesian Formulation & Learning

#### 2.2.1 Formulation

We present three intuitive and simple Bayesian approaches that allow consistent estimation of the underlying reward model in non-personalized slate recommendation. We chose the Bayesian framework because it is highly flexible, since it allows us to incorporate prior information. In addition, it is suited to many MCMC sampling methods that allow consistent estimation of the posterior distribution of the parameters. We start by introducing two important parameters $\phi$ and $\theta$. $\phi$ is a real-valued random variable that quantifies the overall magnitude of a non-click on the slates (i.e. magnitude of the reward signal). A large value of $\phi$ means that users tend to not click on slates very often. $\theta = [\theta_1, \ldots, \theta_N]$, with $N$ as the catalog size is a random vector where each coordinate $\theta_i$ represents the score of a click on item $i$ in the catalog independently of the slate in which it appears.

The Full model makes use of both reward and rank signals. In this model, we put Gamma priors over the magnitude of a non-click on the slates $\phi$ and the scores of a click on each item in the catalog $\theta$. Conditioned on $\phi$, $\theta$, the recommended slate $a$, and the number of impressions $I$, we model the number of non-clicks $nc$ and the number of clicks on each item $c_i, i \in [K]$ using a multinomial distribution with $K + 2$ parameters $I, q, p_1, \ldots, p_K$ expressed as follows:

$$nc, c_1, \ldots, c_K | I, \phi, \theta, a \sim \text{Multinomial} (I, q, p_1, \ldots, p_K),$$

with $q$ (probability of a non-click on the slate $a$) and $p_i, i \in [K]$ (probability of a click on the $i$-th item of the slate $a$) are obtained by normalizing the scores $\phi$ and $\theta$ across slate $a$. The Reward model ignores items ranking (number of clicks on each item in the slates), and only uses the reward signal (i.e. number of clicks on the slates). First, Gamma priors are put over $\phi$ and $\theta$. Conditioned on relevant random variables, we model the number of non-clicks on the slate $nc$ and the number of clicks on the slate $c$ by a multinomial distribution with parameters $I, q, p$:

$$nc, c | I, \phi, \theta, a \sim \text{Multinomial} (I, q, p),$$

with $q$ (probability of a non-click the slate $a$) and $p$ (probability of a click on the the slate $a$) are obtained by normalizing scores $\phi$ and $\theta$ across slate $a$. The Rank model takes into account items ranking only (the number of clicks on each item in the slates). First, Gamma prior is put over the scores $\theta$. Conditioned on the number of clicks on the slate $I_k$ and other relevant random variables, we model the number of clicks on each item in the slate $c_i$ by a multinomial distribution...
3 EXPERIMENTS

3.1 Experimental Setup

We use synthetic data to compare our three methods. We generate \( n \) samples of user interactions with each slate, using a multinomial distribution with known parameters \( \phi = 100 \) and \( \theta \) containing values evenly spaced from 1 to 6. This generative process leads to a dataset similar to the one presented in Table 1 and Section 2.2.2. We then fit our models to this data, and evaluate the ability of each model to estimate the true parameters of the generative process. Since all of our models estimate the parameter \( \theta \), we use this parameter to evaluate the performance of all models. More precisely, we compute the \( L_1 \) distance between \( \hat{\theta}_a \) (the vector of estimated probabilities of a click on item 1 in each recommended slate \( a \)) and \( \theta_a \) (the vector of true probabilities of a click on item 1 in each recommended slate \( a \)).

\[
L_1(p_{\hat{\theta}}, p_{\theta}) = \sum_{a} \left| \frac{\hat{\theta}_a}{\sum_{j \in [K]} \theta_{aj}} - \frac{\theta_a}{\sum_{j \in [K]} \theta_{aj}} \right|
\]  

For experiments with varying slate and catalog sizes, the number of samples per slate \( n_a \) is fixed and set to 1000 (i.e. each slate appears 1000 times in the data). We set the slate size to 2 for experiments with varying number of samples and varying catalog sizes. Catalog size is set to 50 for experiments with varying slate size. In Figures 1a, 1b, 1c, we run all models 50 times, and report the empirical mean and standard deviation of \( L_1 \) errors over these 50 runs. In violin plots 1d and 2 in Appendix C, we use Stan to generate a set of samples \( \hat{\theta}_i \) from the posterior distribution. Eq. 1 is then used to calculate \( L_1 \) distance between \( p_0 \) and \( p_{\hat{\theta}_i} \) for all generated samples \( \hat{\theta}_i \). This process leads to a set of \( L_1 \) errors that we visualize with violin plots. Table 4 in Appendix B summarizes the parameters of
all these experiments. As an additional experiment, we compare Full and Reward models ability to estimate the probability of a non-click on the slates. Precisely, we compare Full and Reward models using the $L_1$ distance between $\hat{q}$ (the vector of estimated probabilities of a non-click for all recommended slates $a$) and $q$ (the vector of true probabilities of a non-click for all recommended slates $a$).

$$L_1(\hat{q}, q) = \sum_{\text{all slates } a} \left| \frac{\hat{q}}{\phi + \sum_{j \in \mathcal{K}} \hat{\theta}_{aj}} - \frac{q}{\phi + \sum_{j \in \mathcal{K}} \theta_{aj}} \right|$$

Clearly, Rank model isn't involved in this comparison as it doesn't estimate the magnitude of a non-click $\phi$. Table 7 in Appendix D shows the results for this experiment.

### 3.2 Results

Figure 1 shows the results for our three models, with varying catalog sizes, slate sizes, and number of samples. In particular, the Full model achieves better $L_1$ error when the catalog size increases (Figure 1a). For instance, with 80 items in the catalog, the Reward and Rank models have 54% and 36% higher relative $L_1$ error than the Full model. In real-world settings, with partners having millions of items in their catalogs, the gap between the Full model and the two other models can become significant. Results for a catalog of size 50 and different slate sizes are shown in Figure 1b. The gap in performance between the Full and Rank models does not change when slate size increases, while the $L_1$ error for the Reward model grows at a much higher rate when the slate size increases. Recall that we have $n$ samples per slate in our data. Since the Reward model only exploits the reward signal, it only uses $n$ samples to estimate items scores, independently of the slate size. In contrast, the other two models use individual ranking. Therefore, the number of samples used to estimate items scores increases as the slate size increases. For instance, a single item would appear in many slates, meaning that samples from all of these slates will be used to estimate that item’s score. Figure 1c shows that the Full model outperforms the other two models for any number of samples. Additionally, we see that the gap in performance between models in Figure 1c seems to be constant. Figure 1d shows a violin plot of the $L_1$ errors obtained by sampling from the posterior for a catalog with 20 items. From Figure 1d we see that the $L_1$ error is more concentrated on the mean in the Full model compared to the Rank and Reward models. We invite the reader to see Tables 5, 6 in Appendix C for additional numerical results from our experiments.

### 4 CONCLUSION

In this paper we have formulated and compared Bayesian models for non-personalized slate recommendation. We have confirmed that the Full model, which utilizes both types of signals, reward and rank, is more favorable than any of the models that are based only on one type of signal. As such, we verified that as the catalog or slate size grows, the performance gains provided by the Full model increase as well.

The impact of the reward signal on the ranking quality demonstrated in this paper is of interest for industrial recommender systems. Although all three models provide an estimate of items scores, the difference between models in practice can be significant since the catalog for applications such as online advertising can contain hundreds of millions of items, and the error grows with the catalog size. A typical industrial personalized recommendation system is based on collaborative filtering and does not have room for including both types of signals considered in this paper. It usually favors the ranking signal over the reward one and is optimized for ranking items correctly on the training dataset. The Full model has the benefit of combining both signals and is optimized for predicting the actual outcome of the user interaction with the slate.

For future work, we plan to extend this framework to the personalized slate recommendation scenario, where the results of recommendations depend on user features. We also plan to test our framework on real-world slate recommendation datasets.
A SUMMARY OF NOTATIONS & DEFINITIONS
Following table summarizes definitions and notations of quantities used in the paper.

| Notation | Definition |
|----------|------------|
| $N$      | Catalog size. |
| $I$      | Total number of impressions. |
| $I_c$    | Total number of clicks. |
| $K$      | Slate size. |
| $a = [a_1, \ldots, a_K]$ | Recommended slate. |
| $nc$     | Number of non-clicks on a recommended slate $a$. |
| $c$      | Number of clicks on a recommend slate $a$. |
| $\phi$   | Score of a non-click. |
| $\theta_i, i \in [N]$ | Scores of a click on item $i$. |
| $c_i, i \in [K]$ | Number of clicks on the $i$-th item in the recommended slate $a$. |

Table 3: Notations and Definitions

B EXPERIMENTAL SETTING DETAILS.
Following table provides details about all parameters used in our experiments.

| Figure | slate size | catalog size | $N$ samples |
|--------|------------|--------------|-------------|
| 1a     | 2          | varying      | 1000        |
| 1b     | varying    | 50           | 1000        |
| 1c     | 2          | 80           | varying     |
| 1d     | 2          | 20           | 1000        |
| 2      | 2          | 80           | 1000        |

Table 4: Parameters values for different experiments.

C EXPERIMENTAL RESULTS.
Figure 2 is a violin plot for a catalog with 80 items.

Figure 2: Catalog with 80 items.

Figure 3: Violin plot of $L_1$ errors distributions for a catalog with 80 items. In this plot, we generate samples $\hat{\theta}_i$ from the posterior and calculate the $L_1$ distance (Eq. 1) between vectors $p_{\theta}$ and $p_{\hat{\theta}_i}$ for all samples $\hat{\theta}_i$. This process results in a set of $L_1$ errors that we visualize using the violin plot.

Tables 5 and 6 show the results used in Figures 1b, 1a, 1c.

D ADDITIONAL EXPERIMENT
Table 7 shows the results for the additional experiment.

| Model | catalog size |
|-------|--------------|
| Full  | 0.03 0.09 0.25 0.48 0.77 1.08 1.42 1.78 2.14 |
| Reward| 0.05 0.14 0.39 0.67 1.07 1.52 1.99 2.55 3.05 |

Table 7: $L_1$ errors (Eq. 2) for varying catalog size.
| Model | 5  | 10 | 20  | 30  | 40  | 50  | 60  | 70  | 80  |
|-------|----|----|-----|-----|-----|-----|-----|-----|-----|
| **Full** | 0.23 | 0.69 | 2.02 | 3.91 | 6.15 | 8.71 | 11.36 | 14.34 | 17.44 |
| **Rank** | 0.28 | 0.94 | 2.70 | 5.31 | 8.37 | 11.60 | 15.29 | 19.31 | 23.83 |
| **Reward** | 0.59 | 1.70 | 3.77 | 6.28 | 9.72 | 13.64 | 17.62 | 22.64 | 26.97 |

| Model | 2  | 3  | 4  |
|-------|----|----|----|
| **Full** | 8.48 | 19.88 | 42.93 |
| **Rank** | 11.73 | 22.61 | 45.88 |
| **Reward** | 13.21 | 42.07 | 134.97 |

Table 5: $L_1$ errors for varying catalog size and slate size.

| Model | 5  | 10  | 50  | 100 | 200 | 300 | 400 | 500 | 600 | 700 | 800 | 900 | 1000 | 5000 | 10000 |
|-------|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-------|-------|--------|
| **Full** | 250.69 | 171.30 | 77.67 | 54.77 | 39.15 | 32.11 | 27.48 | 24.20 | 22.87 | 20.90 | 19.98 | 18.32 | 17.59 | 7.85 | 5.50 |
| **Rank** | 360.00 | 241.73 | 107.76 | 75.80 | 53.42 | 44.08 | 36.72 | 33.32 | 31.32 | 28.72 | 27.14 | 24.96 | 24.27 | 10.65 | 7.45 |
| **Reward** | 419.56 | 287.23 | 124.00 | 90.08 | 65.22 | 52.00 | 45.49 | 38.83 | 35.20 | 31.89 | 30.54 | 28.40 | 26.90 | 12.18 | 8.49 |

Table 6: $L_4$ errors for varying number of samples.