Debiasing the Cloze Task in Sequential Recommendation with Bidirectional Transformers

Khalil Damak*
khalil.damak@louisville.edu
Knowledge Discovery & Web Mining
Lab, Dept. of Computer Science &
Engineering, University of Louisville
Louisville, Kentucky, USA

Sami Khenissi*
sami.khenissi@louisville.edu
Knowledge Discovery & Web Mining
Lab, Dept. of Computer Science &
Engineering, University of Louisville
Louisville, Kentucky, USA

Olfa Nasraoui
olfa.nasraoui@louisville.edu
Knowledge Discovery & Web Mining
Lab, Dept. of Computer Science &
Engineering, University of Louisville
Louisville, Kentucky, USA

ABSTRACT

Bidirectional Transformer architectures are state-of-the-art sequential recommendation models that use a bi-directional representation capacity based on the Cloze task, a.k.a. Masked Language Modeling. The latter aims to predict randomly masked items within the sequence. Because they assume that the true interacted item is the most relevant one, an exposure bias results, where non-interacted items with low exposure propensities are assumed to be irrelevant. The most common approach to mitigating exposure bias in recommendation has been Inverse Propensity Scoring (IPS), which consists of down-weighting the interacted predictions in the loss function in proportion to their propensities of exposure, yielding a theoretically unbiased learning. In this work, we argue and prove that IPS does not extend to sequential recommendation because it fails to account for the temporal nature of the problem. We then propose a novel propensity scoring mechanism, which can theoretically debias the Cloze task in sequential recommendation. Finally we empirically demonstrate the debiasing capabilities of our proposed approach and its robustness to the severity of exposure bias.

CCS CONCEPTS

• **Information systems** → **Collaborative filtering**; **Recommender systems**; **Information retrieval**; • **Computing methodologies** → **Machine learning**.

KEYWORDS

exposure bias; cloze task; sequential recommender system; transformers

---

---

1 INTRODUCTION

Sequential recommendation is a recommendation setting in which the goal is to predict the next best interaction or interactions given a sequence of previous interactions through time [36]. Most successful recent work relies on deep learning models including Recurrent Neural Networks (RNNs) [5, 13–15, 21], Convolutional Neural Networks (CNNs) [18, 32], and more recently, self-attention modules [7, 17, 30, 35]. Recent research has also addressed different biases in recommendation [2]. In particular, exposure bias stems from the partial exposure of items to the users [2], making items with relatively low exposure often considered to be irrelevant in building predictive models. Ideally, recommender systems should capture the true relevance of the items to the users, regardless of their propensities of exposure. However, this is far from true on real life recommendation platforms. Exposure bias can be mitigated during the training of recommender systems [2], mainly by making the models aware of the items' exposure propensities. One of the most common approaches consists of building propensity-weighted loss functions that are unbiased estimates of the desirable relevance-based objectives [27, 28]. This approach, called Inverse Propensity Scoring (IPS), showed success in recommendation settings with user profiles [31]. Despite the progress in this area, to the extent of our knowledge, no previous work has addressed the problem of exposure bias in *sequential* recommendation. In this paper, we mitigate exposure bias in bi-directional transformer-based recommender systems, which are considered state-of-the-art sequential recommender systems [30], and more specifically, the widely-used BERT4Rec model [30]. More broadly however, our work covers any *sequential* recommender system that is trained to optimize the *Cloze* task [7, 33]. Our contributions are summarized as follows:

- We theoretically formulate the problem of exposure bias in the Cloze task, and argue and prove that traditional Inverse Propensity Scoring (IPS) based debiasing frameworks do not extend to sequential recommendation.
- We propose an ideal Cloze task loss function that aims to capture the relevance of items within a sequence context.
- We propose a novel framework for debiasing the Cloze task in sequential recommendation, called Inverse Temporal Propensity Scoring (ITPS), and use it to propose a novel loss function that produces an unbiased estimator for the ideal Cloze task loss.
- We make our implementation available to the public[1].

---

[1]https://github.com/KhalilDMK/DebiasedBERT4Rec

• We conduct experiments that demonstrate the debiasing capabilities of our ITPS-based estimator, and empirically validate our theoretically proven claims.

2 BACKGROUND

Exposure bias occurs when user interactions are dependent upon the exposure of the items. Thus, recommender systems trained on collected data would assume that interaction represents relevance; and hence, non-interacted items would be considered irrelevant regardless of whether they had a chance to be exposed or not. Previous work addressing exposure bias varied in whether they treat bias during the *training* or *evaluation* [2]. The common approach to mitigating exposure bias in the evaluation of recommender systems relies on incorporating Inverse Propensity Scoring (IPS) in the ranking evaluation metrics. More specifically, items are downweighted by their popularities in the evaluation metrics [38]. On the other hand, a variety of techniques were introduced to mitigate exposure bias in the training phase. Some of these techniques are based on integrating a measure of confidence into the unobserved interactions when considering them as irrelevant. Among these techniques, a few [8, 16] considered a uniform weight for all negative items that is lower than one; while others [23, 24] utilized user activity, such as the number of interacted items, to weight the negative interactions. Other approaches used item popularity [12, 39] and user-item similarity [19] instead. Another line of work proposed IPS-based unbiased estimators for the ideal pointwise [28] and pairwise [6, 27] losses, and estimated the propensity of an interaction using the relative item popularity. Departing from the previously mentioned methods, some methods proposed new negative sampling processes to mitigate exposure bias during training. This is usually performed by exploiting *side* information such as social network information [4] or item-based knowledge graphs [37]. Another approach consists of integrating the ability to learn the exposure probability within the model by making assumptions on the probability distribution of exposure [3, 4, 20].

The above methods share the limitation of recommendation with user profiles, where the goal is to predict items to users regardless of the *temporal* context of the previous interactions. To the extent of our knowledge, no previous work has validated these techniques in sequential recommendation. Furthermore, only a few studies [26, 41] have addressed exposure bias in sequential recommendation. However, these approaches treated sequential recommendation in a seq2seq adversarial setting, and use a different formulation of exposure bias which consists of a discrepancy between the training data distribution and the data distribution generated by the model [25], rather than a discrepancy between relevance and interaction.

We address the aforementioned gaps by first studying the limitations of Inverse Propensity Scoring for mitigating exposure bias in sequential recommender systems, and then proposing a debiasing framework that is tailored to sequential recommendation.

3 PROBLEM FORMULATION AND MOTIVATION

We start by formulating the sequential recommendation setting before presenting the Cloze task in bidirectional transformer-based models. Next, we discuss the exposure bias problem in the Cloze task, and how the traditional Inverse Propensity Scoring (IPS) framework does not generalize to sequential recommendation.

3.1 Sequential Recommendation

Let S be a sequential recommendation dataset comprised of $|S|$ sequences. Each sequence S_s is a succession of consecutive item interactions by a user during a certain period of time. An interaction could be defined as a click, rating, review, or consumption, and the time span of the sequence could be short or long. Also, consider a set of items I. The sequence S_s can be represented by its item interactions, for example $S_s = [I_1, I_5, I_9, I_2, I_3]$. We assume that all the sequences are normalized to the same number of time steps T to fit the input requirements of transformer-based models. To do so, sequences that are longer than T time steps are truncated to the most recent T interactions, and sequences that are shorter than T time steps are padded with a padding item 0 at the beginning. Hence, the dataset S is converted to a matrix $S \in I \cup \{0\}^{|S| \times T}$, where element $S_{s,t}$ represents the item, belonging to I, in sequence S_s at time step t. The goal of sequential recommendation is to build a model that is able to accurately predict the next item interaction given a context of previous interactions in a sequence. We represent the trained model by the function f_Ω, with parameters Ω, such that $f_\Omega : [1, |S|] \times [1, T] \times [1, |I|] \to \mathbb{R}; (s, t, i) \mapsto f_\Omega(S_{s,t}, I_i)$. The model f_Ω outputs a prediction of the relevance of item I_i for sequence $S_{s,t}$ at time step t. More specifically, in our work, f_Ω is the bidirectional transformer-based model BERT4Rec [30]. Because the use of Transformers has become common, and because our focus is on debiasing the Cloze task rather than the model itself, we omit an exhaustive background description of transformers, and the BERT4Rec model architecture. Instead, we refer the reader to [30]. That said, we note that all the findings described in this paper are model-agnostic, as long as the model is trained for the Cloze task, and is capable of modeling sequential data.

3.2 The Cloze Task in Sequential Recommendation

The Cloze task [33] consists of randomly masking a percentage ρ of the tokens, in our case items in the sequence, and training the machine learning model to predict those masked tokens. This approach, also called "Masked Language Modeling" (MLM) [7], allows for learning a bidirectional context in the training sequence without any information leakage [30] from the future. This ability of modeling a bidirectional context through the Cloze objective is what gives BERT4Rec its prediction power compared to other models, such as uni-directional self-attention based recommender systems [35]. Consider a training dataset $S^m \in I \cup \{0, \langle mask \rangle\}^{|S| \times T}$. S^m is a masked version of the ground truth dataset S where a fraction ρ of the items is replaced with the token $\langle mask \rangle$ in each sequence. The goal of the Cloze task is to train the hypothesis f_Ω to reconstruct the ground truth dataset S from the masked training dataset S^m. Hence, the loss function associated with the Cloze task is defined as the negative log-likelihood of the predicted probability of correctly predicting the masked tokens, which we formulate as follows:

DEFINITION 1 (CLOZE TASK LOSS FUNCTION).

$$L_{Cloze} = \frac{-1}{|S||I|T} \sum_{s=1}^{|S|} \sum_{t=1}^{T} \sum_{i=1}^{|I|} \mathbb{1}_{\{S_{s,t}^m = \langle mask \rangle\}} Y_{S_s,I_i,t} \times log \, softmax(f_\Omega(S_{s,t}^m, I_i)) \tag{1}$$

where $softmax(f_\Omega(S_{s,t}^m, I_i)) = \frac{e^{f_\Omega(S_{s,t}^m, I_i)}}{\sum_{k=1}^{|I|} e^{f_\Omega(S_{s,t}^m, I_k)}}$ approximates the predicted probability $P(S_{s,t} = I_i | S_s^m)$ of the ground truth item in sequence S_s at time t being I_i given the masked sequence S_s^m. $Y_{S_s,I_i,t}$ is a binary random variable that equals 1 when $I_i \in I$ is interacted with in sequence $S_s \in S$ at time step $t \in [1, T]$, and 0 otherwise.

3.3 Exposure Bias in the Cloze Task

The Cloze loss function, in Definition 1, considers the interacted ground truth item $S_{s,t}$ as the desirable and relevant target item for the input $S_{s,t}^m$. However, as shown in [27–29], interaction does not necessarily signify relevance. In other words, an item could be interacted because it was the most relevant item among the items that the user was exposed to within the item sequence at the corresponding time step. Moreover, non-interacted items could be relevant to some extent, and it could be that the user did not interact with them because they were not exposed to the user. It is this estimation of the relevance of an item with the interaction that engenders the exposure bias. Hence, we can define the ideal Cloze task loss by replacing the interaction random variable $Y_{S_s,I_i,t}$ by the relevance of the item that the user chose to interact with in sequence S_s at time step t, assuming that the user is aware of all items. The awareness of the user of all items completely eliminates the exposure bias because it infers that all items were exposed to the user. Moreover, weighting the interaction by the relevance allows the loss to capture the true relevance of the item. Hence, we consider a Bernoulli random variable $R_{S_s,I_i,t} \sim Ber(\gamma_{S_s,I_i,t})$, where $\gamma_{S_s,I_i,t} = P(R_{S_s,I_i,t} = 1)$ represents the probability of item I_i being relevant in sequence S_s at time step t (i.e., $R_{S_s,I_i,t}$ equals 1). Moreover, we define a Choice random variable that simulates the user behaviour when choosing to interact with item I_i within sequence S_s at time step t. We assume that this choice is contingent upon its relevance compared to all the other items given the sequence context. Hence, we can model the Choice random variable $C_{S_s,I_i,t}$ by a Categorical (Generalized Bernoulli) distribution as follows:

$$C_{S_s,t} \sim Cat(|I|, [\gamma_{S_s,I_1,t}, .., \gamma_{S_s,I_{|I|},t}]). \tag{2}$$

The outcome of the random variable is a vector of $|I|$ zeroes except for a 1 for the item the user chooses to interact with. This means that the user chooses one of the $|I|$ items based on their relevance to the context $S_{s,t}$. We denote the outcome of $C_{S_s,t}$ for item I_i by $C_{S_s,I_i,t}$ and define the ideal Cloze task loss as follows:

DEFINITION 2 (IDEAL CLOZE TASK LOSS FUNCTION).

$$L_{Cloze}^{ideal} = \frac{-1}{|S||I|T} \sum_{s=1}^{|S|} \sum_{t=1}^{T} \sum_{i=1}^{|I|} \mathbb{1}_{\{S_{s,t}^m = \langle mask \rangle\}} C_{S_s,I_i,t} \times \gamma_{S_s,I_i,t} log \, softmax(f_\Omega(S_{s,t}^m, I_i)). \tag{3}$$

The discrepancy between the interaction random variable $Y_{S_s,I_i,t}$ and the product $C_{S_s,I_i,t} \gamma_{S_s,I_i,t}$ causes the Cloze task loss to be biased against the ideal loss, as stated in the following Proposition:

PROPOSITION 1 (EXPOSURE BIAS OF THE CLOZE TASK LOSS FUNCTION). *The Cloze task loss function is biased against the ideal Cloze task loss, such that* $\mathbb{E}[L_{Cloze}] \neq L_{Cloze}^{ideal}$. *See Appendix A.1 for proof.*

3.4 Inverse Propensity Scoring in the Cloze Task and Its Limitations

The common solution to debiasing a maximum likelihood-based loss function for recommendation is Inverse Propensity Scoring (IPS) where an IPS-based estimator of the ideal pointwise loss is obtained by weighting every item prediction for a user by the reciprocal of its exposure propensity for that user [28]. The IPS framework is suitable for debiasing loss functions for recommendation with user profiles. However, we argue that it does not extend to sequential recommendation for the following two reasons:

(1) Inadequacy of the interaction random variable representation: The IPS-based framework for recommendation with user profiles [28] models the interaction random variable $Y_{u,i}$, that represents whether user u interacted with item i, by the product of the relevance and the exposure of the item to the user. The framework relies on two random variables, $O_{u,i} \sim Ber(\theta_{u,i})$ and $R_{u,i} \sim Ber(\gamma_{u,i})$, of exposure and relevance respectively, and models the interaction using $Y_{u,i} = O_{u,i}R_{u,i}$. This means that an item is interacted with by a user if and only if it is both observed by, and relevant to the user. If we extend this modeling of the interaction to sequential recommendation by mapping users to sequences and introducing the temporal component, we would obtain for a sequence S_s, an item I_i and a time step t: $Y_{S_s,I_i,t} = O_{S_s,I_i,t}R_{S_s,I_i,t}$, where $R_{S_s,I_i,t}$ is the relevance random variable and $O_{S_s,I_i,t}$ is a Bernoulli exposure random variable that takes value 1 if item I_i was exposed in sequence S_s at time step t, such that $O_{S_s,I_i,t} \sim Ber(\theta_{S_s,I_i,t})$. θ is the probability of exposure such that $\theta_{S_s,I_i,t} = P(O_{S_s,I_i,t} = 1)$. This modeling of the interaction random variable is inadequate for sequential recommendation. In fact, in traditional recommendation, it is safe to assume that any item that is exposed and relevant to a user is interacted. However, when introducing the temporal component into the equation, the assumption does not hold anymore. This is because a user can only interact with one item at a time. Multiple items can be relevant for the same sequence at the same time step, but only one of them can be interacted with. For this reason, the IPS-based framework for recommendation with user profiles does not extend to sequential recommendation.

(2) Ignoring the temporal component: The IPS estimator for the ideal pointwise loss function down-weights every interaction $Y_{u,i}$ by the propensity of exposure of item i to user u, $\theta_{u,i}$. In order to define an IPS-based Cloze loss for sequential recommendation, we assimilate the users to sequences and consider the propensity of exposure of an item I_i in a sequence S_s as $\theta_{S_s,I_i} = P(O_{S_s,I_i} = 1)$, where $O_{S_s,I_i} \sim Ber(\theta_{S_s,I_i})$ is a Bernoulli random variable that takes the value 1 when item I_i is exposed in sequence S_s. We define the IPS-based Cloze loss as follows:

DEFINITION 3 (INVERSE PROPENSITY SCORING-BASED CLOZE LOSS FUNCTION).

$$L_{Cloze}^{IPS} = \frac{-1}{|S||I|T} \sum_{s=1}^{|S|} \sum_{t=1}^{T} \sum_{i=1}^{|I|} \mathbb{1}_{\{S_{s,t}^m = \langle mask \rangle\}} \frac{Y_{S_s,I_i,t}}{\theta_{S_s,I_i}} \times log \, softmax(f_\Omega(S_{s,t}^m, I_i)). \tag{4}$$
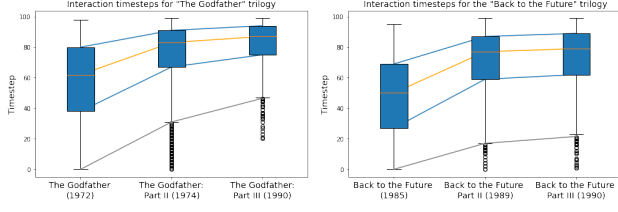
Figure 1: Boxplots of the interaction timesteps for "The Godfather" and "Back to the Future" trilogies. The interaction distributions vary through time, meaning that the exposure propensities must not be considered static.

The IPS-based Cloze loss function can only be completely unbiased if the propensity of every item I_i in every sequence S_s at time step t, $\theta_{S_s,I_i,t}$, is equal to the "static" propensity, θ_{S_s,I_i}, of item I_i in sequence S_s. We state this in the following proposition:

PROPOSITION 2 (UNBIASEDNESS CONDITION OF THE IPS-BASED CLOZE LOSS FUNCTION).

$$\mathbb{E}[L_{Cloze}^{IPS}] = L_{Cloze}^{ideal} \Leftrightarrow \theta_{S_s,I_i,t} = \theta_{S_s,I_i}, \forall (S_s, I_i, t) \in S \times I \times [1..T].$$
(5)

The proof is in Appendix A.2. This unbiasedness condition of the IPS estimator is unlikely and hard to satisfy as the propensities of exposure tend to vary with the temporal context. We demonstrate this in Figure 1 where we show boxplots of the interaction time steps for two movie trilogies in the Movielens 1M dataset [11]. The boxplots show that there are movies that tend to be watched later than others in the sequence; for instance, sequels tend to be watched after the original movies. We chose movies that are older than the dataset to ensure that the differences in observation time are not related to the release dates of the movies, but rather to the temporal context within the trilogies. Hence, given that the interaction distribution tends to vary with time, it is safe to assume that the exposure propensities also vary with time. Thus, in contrast to the IPS framework, they should not be considered static in sequential recommendation. The latter observation additionally shows how the IPS framework does not extend to sequential recommendation. This consequently calls for proposing a new framework that is specifically tailored for debiasing the Cloze task in sequential recommendation, which is the subject of the next section.

4 INVERSE TEMPORAL PROPENSITY SCORING FOR AN UNBIASED CLOZE TASK

The Inverse Propensity Scoring technique fails to capture the temporal component of the sequential recommendation setting, and hence fails to provide an unbiased estimation of the ideal Cloze task loss. We propose a debiasing framework that is tailored to the Cloze task in sequential recommendation, and that we call **Inverse Temporal Propensity Scoring (ITPS)**. In ITPS, we address the two main limitations of IPS that prevent it from generalizing to sequential recommendation. First, to address the issue of the inadequacy of the interaction random variable representation, we include the outcome of the Choice random variable for item I_i in the interaction model for the following formulation:

DEFINITION 4 (INTERACTION RANDOM VARIABLE REPRESENTATION IN THE ITPS FRAMEWORK).

$$Y_{S_s,I_i,t} = C_{S_s,I_i,t} \; O_{S_s,I_i,t} \; R_{S_s,I_i,t}$$
(6)

The latter formulation of the interaction allows for only one item to be interacted within a sequence at a given time step, which is adequate for sequential recommendation. Now, an item I_i is interacted by a user ($Y_{S_s,I_i,t} = 1$) in a sequence S_s at time step t if and only if the item is exposed ($O_{S_s,I_i,t} = 1$), relevant ($R_{S_s,I_i,t} = 1$), and chosen by the user based on its relevance ($C_{S_s,I_i,t} = 1$). Finally, to account for the temporal component in sequential recommendation in ITPS, we weight the prediction of every item I_i in every sequence S_s at every time step t by the temporal propensity $\theta_{S_s,I_i,t}$, as opposed to the static propensity θ_{S_s,I_i} of IPS. Thus, we define the ITPS-based Cloze task loss function as follows:

DEFINITION 5 (INVERSE TEMPORAL PROPENSITY SCORING-BASED CLOZE LOSS FUNCTION).

$$L_{Cloze}^{ITPS} = \frac{-1}{|S||I|T} \sum_{s=1}^{|S|} \sum_{t=1}^{T} \sum_{i=1}^{|I|} \mathbb{1}_{\{S_{s,t}^m = \langle mask \rangle\}} \frac{Y_{S_s,I_i,t}}{\theta_{S_s,I_i,t}} \\ \times log \, softmax(f_\Omega(S_{s,t}^m, I_i))$$
(7)

This new ITPS-based loss is an unbiased estimator of the ideal Cloze task loss, as stated in the following proposition:

PROPOSITION 3. *The ITPS-based Cloze task loss is unbiased for the ideal Cloze task loss, meaning that* $\mathbb{E}[L_{Cloze}^{ITPS}] = L_{Cloze}^{ideal}$.

The proof is in Appendix A.3.

5 EXPERIMENTAL EVALUATION

We perform experiments to assess the validity of our theoretical claims of unbiasedness and the applicability of our approach in real recommendation settings. We use semi-synthetic and real world datasets. The semi-synthetic data, used in Section 5.1, provides a full visibility of the data properties, allowing us to evaluate the debiasing capabilities of our proposed approach. Moreover, it allows us to control the data properties in order to evaluate the robustness of our approach to varying bias levels. The real datasets, used in Section 5.2, allow us to evaluate the applicability of our approach in real recommendation settings. Additionally, we simulate a feedback loop to evaluate the longitudinal effects of the proposed debiasing.

5.1 Experiments on Semi-Synthetic Data

We perform experiments to answer three research questions:

RQ1: How well does the proposed ITPS estimator capture the true relevance?

RQ2: How robust is the proposed ITPS estimator to increasing levels of exposure bias?

RQ3: How important is an unbiased evaluation in assessing exposure debiasing?

5.1.1 Data. Semi-synthetic experiments are necessary due to the unavailability of any open or public unbiased sequential recommendation dataset. In fact, only an exposure-unbiased testing dataset would allow us to truly compare the debiasing capabilities of the different approaches - a claim that we validate in RQ3. We use

Table 1: Statistics of the real (ml-100k) and semi-synthetic (ss-ml-100k) Movielens 100K datasets.

Dataset	# sequences	# items	# ratings	Avg. length	Sparsity
ml-100k	943	1,349	99,287	105.28	92.19%
ss-ml-100k	943	229	94,104	99.79	56.42%

the Movielens 100K (ml-100k)[2] dataset because it is a benchmark dataset that can be used for sequential recommendation since it includes interaction timestamps. This data is described in the first row of Table 1. The choice of this dataset is justified due to its relatively low number of sequences (users) and items, compared to other sequential datasets. In fact, our first task is to generate all data properties, including relevance, exposure, and interaction for all sequence, item and timestep tuples; a task that is resource-expensive, especially in memory requirements. Considering a dataset with $|S|$ sequences, $|I|$ items and T time steps, the number of parameters that need to be predicted and kept into memory for each controlled property is $|S| \times |I| \times T$. Hence, given the ml-100k dataset statistics, we would be predicting over 127 Million values for every property. For this reason, using other benchmark datasets with tens of thousands of sequences or items, is simply prohibitive with our current resources. Moreover, similar conclusions could be drawn regardless of the dataset, assuming a high reconstruction quality. Our goal is to use the available ratings to infer all the data properties, namely the relevance, exposure, and interaction of all items $I_i \in I$, in all sequences $S_s \in S$, and at all time steps $t \in [1, T]$. This is done in the following steps:

(1) We normalize the dataset to $T = 100$ time steps.

(2) We train a Tensor Factorization (TF) model [1, 40] on the available (sequence, item, timestep, rating) tuples to reconstruct the missing ratings. We train the model on the Mean Squared Error (MSE) loss for rating prediction. Finally, we use the trained TF model to reconstruct the rating tensor by predicting the missing ratings. Given that the rating represents an explicit measure of satisfaction of a user with an item, we can approximate the probability of relevance of an item I_i in a sequence S_s at a time step t by normalizing the predicted rating with the sigmoid function as follows: $\gamma_{S_s, I_i, t} \approx \sigma(\hat{r}_{s,i,t})$. Here, $\hat{r}_{s,i,t} = \sum_{k=1}^{d} P_{s,k} Q_{i,k} W_{t,k}$ is the predicted rating, where P, Q, and W are respectively the sequence, item, and time latent factor matrices, which all have d latent features.

(3) We train another Tensor Factorization model to predict the probabilities of exposure. We convert every rating in the dataset to a positive exposure, and sample a portion of non-interacted tuples as negative exposures. We assume that an item has a higher probability of not being exposed than of being exposed, which is a realistic assumption given the abundance of items in recommendation platforms. Thus, we sample 3 negative exposure tuples for every positive exposure tuple. We train the TF model using the Binary Cross Entropy loss for exposure classification. Similarly to step (2), we approximate the propensity of exposure of an item I_i in a sequence S_s at a time step t by the predicted exposure as follows: $\theta_{S_s, I_i, t} \approx \hat{o}_{s,i,t}$. Here, $\hat{o}_{s,i,t}$ is the predicted exposure probability of item i in sequence s at time step t, obtained by: $\hat{o}_{s,i,t} = \sigma(\sum_{k=1}^{d} P_{s,k} Q_{i,k} W_{t,k})$.

[2]https://grouplens.org/datasets/movielens/100k/

(4) Following [28], we introduce a hyperparameter p that controls the skewness of the exposure distribution, and hence the level of exposure bias, as follows:

$$\theta_{S_s, I_i, t} \approx \hat{o}_{s,i,t}^{p}. \tag{8}$$

The higher the value of p, the higher the level of exposure bias introduced. We will control the value of p to study RQ2.

(5) We generate the interaction random variable for every sequence S_s, item I_i, and timestep t combination by following the probabilistic model presented in Equation 6, such that:

$$O_{S_s, I_i, t} \sim Ber(\theta_{S_s, I_i, t}) \tag{9}$$

$$R_{S_s, I_i, t} \sim Ber(\gamma_{S_s, I_i, t}) \tag{10}$$

$$C_{S_s, I_i, t} \sim Cat(|I|, [\gamma_{S_s, I_1, t}, .., \gamma_{S_s, I_{|I|}, t}]) \tag{11}$$

$$Y_{S_s, I_i, t} = C_{S_s, I_i, t} \, O_{S_s, I_i, t} \, R_{S_s, I_i, t}. \tag{12}$$

In our experiments, we obtain $C_{S_s, I_i, t}$ by considering a rational user interacting with the exposed item ($O_{S_s, I_i, t} = 1$) with highest relevance $\gamma_{S_s, I_i, t}$.

(6) Finally, we filter the interacted instances to construct the semi-synthetic sequential dataset. The statistics of a sample generated semi-synthetic dataset are presented in the second row of Table 1.

5.1.2 Evaluation Process. Our estimators should be evaluated in terms of their capacity to capture the true relevance of the test interactions. However, our sequence interactions are obtained with the interaction probabilistic model in Equation 6, which requires all interactions to be exposed. Hence, sampling the test and validation interactions from the semi-synthetic sequences would not allow for an evaluation in terms of the true relevance. This is because the most relevant items are not necessarily exposed to the user. We cope with this issue using the following evaluation process: We start by splitting the data into training, validation and test sets by considering the last item interaction in each sequence for testing and the second to last for validation. Then, we replace every item interaction in the validation and test sets by the item I_i with the highest relevance $\gamma_{S_s, I_i, t}$ in the corresponding sequence S_s and at the corresponding timestep t. This way, the model is evaluated on its ability to predict the most relevant item, which translates to its ability to capture the true relevance of the items. This being done, we compare the ranking of the test and validation instances to 100 randomly sampled items. Note that negative sampling does not introduce any bias because, regardless of their exposure, all the negative items are less relevant than the test and validation items. Thus, our evaluation process is unbiased and evaluates the models in terms of their capacity to capture the true relevance of the items. We use Normalized Discounted Cumulative Gain ($NDCG@k$) and Recall ($R@k$) for the ranking evaluation.

5.1.3 Models Compared. We compare the following models:

- **BERT4Rec:** This is the original BERT4Rec model trained to optimize the Cloze task loss in Equation 1. It relies solely on the interaction information and does not incorporate any exposure debiasing.
- **IPS-BERT4Rec:** This is the BERT4Rec model trained with the IPS-based Cloze loss function in Equation 4. We estimate

Table 2: Model comparison in terms of capturing the true relevance: Average Recall@k and NDCG@k results over 5 replicates. The best results are in bold and second to best results are underlined. A value with * is significantly higher than the next best value (p-value < 0.05).

Model	R@10	NDCG@10	R@5	NDCG@5
BERT4Rec	0.7992	0.6065	0.6917	0.5716
IPS-BERT4Rec	0.7890	0.5961	0.6868	0.5628
ITPS-BERT4Rec	<u>0.8027</u>*	<u>0.6110</u>*	<u>0.6997</u>*	<u>0.5777</u>*
Oracle	**0.8218***	**0.6247***	**0.7083***	**0.5880***

the "static" exposure propensities by averaging the temporal exposures, such that $\theta_{S_s,I_i} = \frac{1}{T}\sum_{t=1}^{T}\theta_{S_s,I_i,t}, \forall (S_s, I_i) \in S \times I$.

- **ITPS-BERT4Rec:** This is the BERT4Rec model trained with our ITPS-based Cloze task loss in Equation 7. The loss relies on the temporal propensities $\theta_{S_s,I_i,t}$ to provide an unbiased estimation of the ideal Cloze task loss.
- **Oracle:** This is the BERT4Rec model trained with the ideal Cloze task loss in Equation 3. The loss has access to the true relevance of the items $\gamma_{S_s,I_i,t}$ in the training, and hence, is able to provide a completely unbiased representation of the user preferences. Hence, this model provides an upper bound on capturing the true relevance.

Because the goal of the experiments is to assess the impact of the different debiasing frameworks, we leave the comparison to additional baselines for future work.

5.1.4 Hyperparameter Tuning. We tune all the models presented in Section 5.1.3, along with the Tensor Factorization models presented in steps 2 and 3 of Section 5.1.1 as described below.

Tuning the BERT4Rec models: Using random search, we tune the number of hidden units within the set {8, 16, 32, 64}, the number of transformer blocks within {1, 2}, the number of attention heads within {1, 2}, the batch size within {8, 16, 32}, the dropout rate within {0, 0.1, 0.2, 0.4}, and finally, the masking probability ρ of the Cloze task within {0.1, 0.15, 0.2, 0.4, 0.6}. We try 30 random combinations, and compare the average $NDCG@10$ results over 3 replicates on the validation set.

Tuning the Tensor Factorization models: We randomly split the data into training, validation and test sets with the respective ratios 80%, 10% and 10%. We adopt a grid search by trying all combinations of number of latent features within {50, 100, 200}, and batch size within {64, 128, 256}. We replicate every experiment 3 times and compare the average performances on the validation set. The rating-based TF model from step 2 is tuned in terms of Mean Squared Error (MSE) for rating prediction, while the exposure-based TF-model from step 3 is tuned in terms of Area Under the ROC Curve (AUC) for exposure classification.

5.1.5 RQ1: How well does the proposed ITPS estimator capture the true relevance? To answer this research question, we evaluate the models in terms of their capacity to capture the true relevance using the evaluation process described in Section 5.1.2. We summarize the results in Table 2. The best performer on all metrics is the Oracle model, owing to its explicit optimization using the relevance
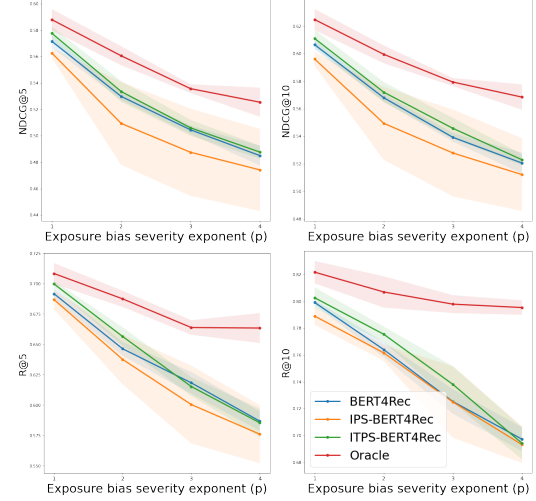

Figure 2: Robustness of the ranking performance of the different models to increasing levels of exposure bias. All the values are averages over 5 replicates and the 90% confidence intervals are highlighted. ITPS-BERT4Rec was the best in withstanding increasing levels of exposure bias overall.

levels. The ITPS-BERT4Rec model was second-to-best in all configurations, outperforming the naive BERT4Rec and IPS-BERT4Rec. These findings demonstrate the power of the ITPS debiasing framework and validate the theoretical claims of exposure debiasing of the proposed estimator. Finally and interestingly, IPS-BERT4Rec performed worse than the naive BERT4Rec. This is probably due to the fact that it is trained on estimated static propensities, obtained by averaging the temporal propensities, rather than true propensities.

5.1.6 RQ2: How robust is the proposed ITPS estimator to increasing levels of exposure bias? To answer this research question, we train and evaluate the models on semi-synthetic datasets generated with increasing levels of exposure bias. The level of exposure bias is controlled by the power p that governs the propensities $\theta_{S_s,I_i,t}$ in Equation 8. We increase p from 1 to 4 with an increment of 1, where the higher the value of p, the stronger the exposure bias introduced in the data, and show the evolution of the ranking metrics in Figure 2. All the models' performances decrease with increasing levels of exposure bias, however with different slopes. The IPS-BERT4Rec model shows the worst performance in handling increasing exposure bias. Its performance quickly degrades starting from $p = 2$. This shows the inability of the IPS framework to mitigate exposure bias in sequential recommendation. On the other hand, ITPS-BERT4Rec shows the best performance overall in approximating the Oracle. These findings validate the robustness of the proposed ITPS estimator in handling even extreme levels of exposure bias, and in capturing the true relevance of the items in a sequence and temporal context. Finally, as opposed to IPS-BERT4Rec which shows a significantly high and increasing variance, ITPS-BERT4Rec shows a relatively low and steady variance that compares to the variance of BERT4Rec. This further demonstrates the robustness of our proposed approach to increasing levels of exposure bias.

Table 3: Average R@k and NDCG@k over 5 replicates obtained with a standard evaluation process. ↑ means the ranking increased and ↓ means the ranking decreased compared to the unbiased results from section 5.1.5. Best results are in bold and second to best are underlined. A value with * is significantly higher than the next best value (p-value < 0.05).

Model	R@10	NDCG@10	R@5	NDCG@5
BERT4Rec	0.7782 ↓	0.5851 ↓	0.6655 ↓	0.5486
IPS-BERT4Rec	0.7835 ↑	0.5854 ↑	0.6665 ↑	0.5475
ITPS-BERT4Rec	<u>0.7873*</u>	<u>0.5909*</u>	<u>0.6754*</u>	<u>0.5545</u>
Oracle	**0.8000**	**0.5983**	**0.6795**	**0.5593**

5.1.7 RQ3: How important is an unbiased evaluation in assessing exposure debiasing? In this research question, we aim to demonstrate the importance of the unbiased evaluation process, explained in Section 5.1.2, in evaluating the capacity of the models in capturing the true preferences of the users. To do so, we try to re-evaluate the tuned models using a standard Leave One Out (LOO) evaluation process, in which we compare the interacted test items to 100 randomly sampled items. This evaluation process is biased because the test items are not necessarily the most relevant items due to their exposure requirement. This results in an overestimation of the performance of the biased models, and their capacity to capture the true relevance. We summarize the results obtained with the standard LOO evaluation process in Table 3. We notice a discrepancy between the results obtained with the standard and unbiased evaluation processes. In fact, with the standard evaluation process, IPS-BERT4Rec outperformed BERT4Rec in almost all the settings, which reflects an over-estimation of the debiasing capabilities of the IPS framework and its ability to capture the relevance of items given the sequence context. The ITPS-BERT4Rec model was nonetheless still the top performer following the Oracle. These findings validate the necessity of relying on the unbiased evaluation setting, as it allows us to truly evaluate the properties of the different estimators.

5.2 Experiments on Real Data

We perform offline experiments on real recommendation datasets that aim to answer the following research questions:

RQ4: How well does our proposed ITPS estimator perform in terms of ranking accuracy?

RQ5: How well does our proposed ITPS estimator help mitigate popularity bias in the short and long terms?

5.2.1 Data. We rely on three datasets that are commonly used in sequential recommendation research [30], which are: the Movielens 1M (ml-1m)[3] [11], Movielens 20M (ml-20m)[3] [11], and Amazon Beauty (beauty)[4] [22]. For each of the datasets, we consider any rating, regardless of its value, as a positive interaction, then, we filter out users with less than 5 interactions to reduce the data sparsity. The dataset statistics are summarized in Table 4.

5.2.2 Evaluation and Propensity Estimation. Previously (Section 5.1), we were able to train our models using the true (temporal) exposure

---

[3]https://grouplens.org/datasets/movielens
[4]https://nijianmo.github.io/amazon/index.html

Table 4: Real dataset statistics.

Dataset	Task	Sequences	Items	Interactions	Avg. length	Sparsity
ml-1m	Movie rec.	6,040	3,416	999,611	165.49	95.15%
ml-20m	Movie rec.	138,493	18,345	19,984,024	144.29	99.21%
beauty	Product rec.	40,226	54,542	353,962	8.79	99.98%

propensities and evaluate their ability to model the relevance using the temporal relevance levels, which were available through the use of semi-synthetic data. However, in real-world data, neither the (temporal) exposure propensities, nor the temporal relevance levels are available. This causes the following two issues: (1) We cannot evaluate the models' ability to learn the true relevance of the items to the users because we do not know the true temporal relevance levels; and (2) we cannot train the IPS-BERT4Rec and ITPS-BERT4Rec models as they rely on the exposure and temporal exposure propensities. To solve the first issue, we propose an evaluation process that is based on popularity-based negative sampling. In fact, the main issue with the standard LOO evaluation process is that some of the randomly sampled negative items to which we are comparing our test and validation items may be as relevant, or possibly more relevant, than the test and validation items. We propose to sample the negative items for every sequence based on their popularities, meaning the higher the popularity of an item, the higher the probability that it will be sampled as a negative item. The idea is that more popular items have a higher likelihood that they have been exposed to the user and have not been interacted with because of their irrelevance to the user. The latter popularity-based negative sampling does not completely eliminate exposure bias in the evaluation. However, it is intended to mitigate it. Note that using popularity-based sampling to mitigate exposure bias was used in previous work [10] in the training phase. We are extending it to evaluation. To solve the second issue, we build on previous work [6, 27] and estimate the temporal exposure propensity of an item to a user by the temporal popularity of the item such that:

$$\hat{\theta}_{S_s,I_i,t} = \frac{\sum_{j=1}^{|S|} Y_{S_j,I_i,t}}{\sum_{k=1}^{T} \sum_{l=1}^{|I|} \sum_{j=1}^{|S|} Y_{S_j,I_l,k}}. \tag{13}$$

Similarly, we estimate the static exposure propensity of an item in a sequence with the item's popularity, which corresponds to the sum of the estimated temporal exposure propensities expressed as follows: $\hat{\theta}_{S_s,I_i} = \sum_{t=1}^{T} \hat{\theta}_{S_s,I_i,t}$. Thus, we train the IPS-BERT4Rec and ITPS-BERT4Rec models, presented in section 5.1.3, using the estimated exposure propensities and estimated temporal exposure propensities, respectively.

5.2.3 Hyperparameter Tuning. For the beauty and ml-1m datasets, we perform the same hyperparameter tuning process described in Section 5.1.4 on the semi-synthetic dataset. However, for the ml-20m dataset, we increase the ranges of some of the hyperparameters given the relatively higher size and complexity of the dataset. Hence, the number of hidden units is tuned within {64, 128, 256}, the number of transformer blocks within {1, 2, 3}, the number of attention heads within {1, 2, 4, 8}, the batch size within {64, 128, 256}, and the dropout rate within {0, 0.01, 0.1, 0.2}.

5.2.4 RQ4: How well does the proposed ITPS estimator perform in terms of ranking accuracy? To measure the ranking capabilities of the proposed approach, we evaluate the tuned models using the evaluation process presented in Section 5.2.2 which ensures that exposure bias is mitigated. Thus, the ranking accuracy results should provide a good approximation of how well the models capture the true relevance of the items to the users. We summarize the results on the three datasets in Table 5. Our proposed ITPS-BERT4Rec model was the best performer in all the settings, showing significantly superior performance than the BERT4Rec and the IPS-BERT4Rec models in all the metrics and on all the datasets. This validates the ability of the proposed ITPS debiasing framework to learn the true relevance of the items to the users, in addition to its applicability in real recommendation settings. Moreover, interestingly, the ranking performance was not consistent for the second to best model. In fact, IPS-BERT4Rec outperformed BERT4Rec overall on both the ml-1m and beauty datasets but not on the ml-20m dataset.

5.2.5 RQ5: How well does the proposed ITPS estimator help mitigate popularity bias in the short and long terms? To answer this question, we implement a feedback loop which simulates a real recommendation environment. The feedback loop consists of consecutive recommendation iterations where at each iteration, the recommender system is re-trained and generates top 10 recommendations for every user in the dataset. Each user then interacts with one of the recommended items and the interactions are added to the dataset for training future iterations. We simulate the user's choice with a uniform distribution, meaning that the interacted item is chosen at random from the recommendation list. Moreover, the choice of re-training the model at each iteration is related to the nature of our training datasets. In fact, we assume that an iteration corresponds to one day and that users interact with at most one movie or beauty product per day. This setting could be extended to other types of recommendation datasets in the future. Finally, we assume that all the users interact with one item at every iteration. As was discussed in [9], this assumption is meant to speed-up the feedback loop process and should not alter the general characteristics of the emerging phenomena. Thus, no conclusions will be altered. We evaluate the popularity debiasing capabilities by looking at the novelty of the top 10 recommendations. The novelty is assessed using the Expected Free Discovery (EFD) [34], which is a measure of the ability of a system to recommend relevant long-tail items [34] and is calculated as follows

$$EFD@\mathcal{K}(TopK) = -\frac{1}{|S|} \sum_{s=1}^{|S|} \frac{1}{\mathcal{K}} \sum_{i \in TopK(S_s)} log_2 \hat{\theta}_{S_s,i}, \quad (14)$$

where $TopK$ is the top $\mathcal{K}$ recommendation matrix in which every row represents the Top $\mathcal{K}$ recommendations in a sequence.

We summarize the evolution of $EFD@10$ for 10 feedback loop iterations on the three datasets in Figure 3. On both the ml-20m and beauty datasets, our proposed ITPS-BERT4Rec model showed the best results in all iterations. The difference in performance compared to the other two models was significant in all the iterations for the beauty dataset and in most iterations for the ml-20m dataset. However, we notice a change in trend in the ml-1m dataset where
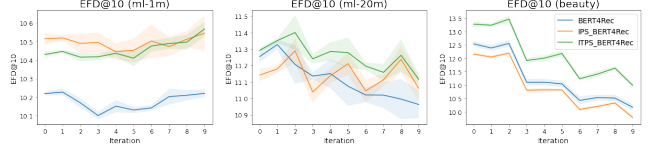

Figure 3: Evolution of EFD@10 with respect to feedback loop iterations. All values are averages over 5 replicates and 90% confidence intervals are highlighted. ITPS-BERT4Rec showed the best short and long-term popularity debiasing capabilities on the ml-20m and beauty datasets.

IPS-BERT4Rec and ITPS-BERT4Rec showed a relatively similar popularity debiasing performance, that still outperformed BERT4Rec. We believe that the difference in trend in the ml-1m dataset is due to the relatively low number of items and low sparsity of the dataset making the popularity bias problem less prominent compared to the other datasets. Moreover and interestingly, the vanilla BERT4Rec outperformed IPS-BERT4Rec on the beauty dataset. The overall superior performance of our proposed ITPS-BERT4Rec model shows the impact of exposure debiasing on popularity debiasing, where modeling the true preferences of the user results in more diverse and novel recommendations yielding a higher item discovery by the user. Moreover, the ml-20m and beauty datasets showed, overall, decreasing trends for *EFD* with respect to the feedback loop iterations for all the models. This means that the issue of popularity bias tends to worsen with time. However, the relatively low slope of ITPS-BERT4Rec demonstrates the importance of mitigating exposure bias to mitigate long-term popularity bias.

6 CONCLUSION

We studied the problem of exposure bias in sequential recommendation within the scope of bidirectional transformers trained to optimize the Cloze task, and proposed an ideal Cloze task loss that captures the true relevance. Then, we argued and proved that IPS estimators do not extend to sequential recommendation. In addition, we proposed a theoretically unbiased estimator for the ideal Cloze task loss, and formulated a framework that allows for an unbiased training and evaluation of sequential recommender systems. Our experiments empirically validated our claims of debiasing of the proposed ITPS-BERT4Rec estimator, and demonstrated its robustness to increasing levels of exposure bias, along with its longitudinal impact on popularity debiasing. Future work should validate and challenge the assumptions on which our theory is based.

REFERENCES

[1] Gediminas Adomavicius, Ramesh Sankaranarayanan, Shahana Sen, and Alexander Tuzhilin. 2005. Incorporating contextual information in recommender systems using a multidimensional approach. *ACM Transactions on Information Systems (TOIS)* 23, 1 (2005), 103–145.

[2] Jiawei Chen, Hande Dong, Xiang Wang, Fuli Feng, Meng Wang, and Xiangnan He. 2020. Bias and Debias in Recommender System: A Survey and Future Directions. *arXiv preprint arXiv:2010.03240* (2020).

Table 5: Average Recall (R) and NDCG (N) results over 5 replicates on the three real interaction datasets. The best results are in bold and second to best results are underlined. A value with * is significantly higher than the next best value (p-value < 0.05).

Dataset	ml-1m				ml-20m				beauty			
Model	N@5	R@5	N@10	R@10	N@5	R@5	N@10	R@10	N@5	R@5	N@10	R@10
BERT4Rec	0.2820	0.4086	0.3262	0.5454	0.4205*	0.5583*	0.4624*	0.6876*	0.1056	0.1516	0.1260	0.2148
IPS-BERT4Rec	0.3416*	0.4751*	0.3801*	0.5940*	0.4004	0.5389	0.4434	0.6715	0.1053	0.1528	0.1268	0.2195
ITPS-BERT4Rec	**0.3451***	**0.4796**	**0.3844***	**0.6007***	**0.4295***	**0.5674***	**0.4709***	**0.6952***	**0.1197***	**0.1745***	**0.1444***	**0.2510***

[3] Jiawei Chen, Can Wang, Sheng Zhou, Qihao Shi, Jingbang Chen, Yan Feng, and Chun Chen. 2020. Fast Adaptively Weighted Matrix Factorization for Recommendation with Implicit Feedback.. In *AAAI*. 3470–3477.

[4] Jiawei Chen, Can Wang, Sheng Zhou, Qihao Shi, Yan Feng, and Chun Chen. 2019. Samwalker: Social recommendation with informative sampling strategy. In *The World Wide Web Conference*. 228–239.

[5] Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (Doha, Qatar). Association for Computational Linguistics, 1724–1734. https://doi.org/10.3115/v1/D14-1179

[6] Khalil Damak, Sami Khenissi, and Olfa Nasraoui. 2021. Debiased Explainable Pairwise Ranking from Implicit Feedback. In *Fifteenth ACM Conference on Recommender Systems*. 321–331.

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[8] Robin Devooght, Nicolas Kourtellis, and Amin Mantrach. 2015. Dynamic matrix factorization with priors on unknown values. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*. 189–198.

[9] Andres Ferraro, Dietmar Jannach, and Xavier Serra. 2020. Exploring Longitudinal Effects of Session-based Recommendations. In *Fourteenth ACM Conference on Recommender Systems*. 474–479.

[10] Zeno Gantner, Lucas Drumond, Christoph Freudenthaler, and Lars Schmidt-Thieme. 2012. Personalized ranking for non-uniformly sampled items. In *Proceedings of KDD Cup 2011*. PMLR, 231–247.

[11] F. Maxwell Harper and Joseph A. Konstan. 2015. The MovieLens Datasets: History and Context. *ACM Trans. Interact. Intell. Syst.* 5, 4, Article 19 (dec 2015), 19 pages. https://doi.org/10.1145/2827872

[12] Xiangnan He, Hanwang Zhang, Min-Yen Kan, and Tat-Seng Chua. 2016. Fast matrix factorization for online recommendation with implicit feedback. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*. 549–558.

[13] Balázs Hidasi and Alexandros Karatzoglou. 2018. Recurrent neural networks with top-k gains for session-based recommendations. In *Proceedings of the 27th ACM international conference on information and knowledge management*. 843–852.

[14] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. 2015. Session-based recommendations with recurrent neural networks. *arXiv preprint arXiv:1511.06939* (2015).

[15] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Comput.* 9, 8 (Nov. 1997), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

[16] Yifan Hu, Yehuda Koren, and Chris Volinsky. 2008. Collaborative filtering for implicit feedback datasets. In *2008 Eighth IEEE International Conference on Data Mining*. Ieee, 263–272.

[17] Wang-Cheng Kang and Julian McAuley. 2018. Self-attentive sequential recommendation. In *2018 IEEE International Conference on Data Mining (ICDM)*. IEEE, 197–206.

[18] Yann LeCun, Patrick Haffner, Léon Bottou, and Yoshua Bengio. 1999. Object recognition with gradient-based learning. In *Shape, contour and grouping in computer vision*. Springer, 319–345.

[19] Yanen Li, Jia Hu, ChengXiang Zhai, and Ye Chen. 2010. Improving one-class collaborative filtering by incorporating rich user information. In *Proceedings of the 19th ACM international conference on Information and knowledge management*. 959–968.

[20] Dawen Liang, Laurent Charlin, James McInerney, and David M Blei. 2016. Modeling user exposure in recommendation. In *Proceedings of the 25th international conference on World Wide Web*. 951–961.

[21] Zachary Chase Lipton. 2015. A Critical Review of Recurrent Neural Networks for Sequence Learning. *CoRR* abs/1506.00019 (2015).

[22] Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton Van Den Hengel. 2015. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*. 43–52.

[23] Rong Pan and Martin Scholz. 2009. Mind the gaps: weighting the unknown in large-scale one-class collaborative filtering. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*. 667–676.

[24] Rong Pan, Yunhong Zhou, Bin Cao, Nathan N Liu, Rajan Lukose, Martin Scholz, and Qiang Yang. 2008. One-class collaborative filtering. In *2008 Eighth IEEE International Conference on Data Mining*. IEEE, 502–511.

[25] Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. 2015. Sequence level training with recurrent neural networks. *arXiv preprint arXiv:1511.06732* (2015).

[26] Ruiyang Ren, Zhaoyang Liu, Yaliang Li, Wayne Xin Zhao, Hui Wang, Bolin Ding, and Ji-Rong Wen. 2020. Sequential recommendation with self-attentive multi-adversarial network. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 89–98.

[27] Yuta Saito. 2019. Unbiased Pairwise Learning from Implicit Feedback. In *NeurIPS 2019 Workshop on Causal Machine Learning*.

[28] Yuta Saito, Suguru Yaginuma, Yuta Nishino, Hayato Sakata, and Kazuhide Nakata. 2020. Unbiased recommender learning from missing-not-at-random implicit feedback. In *Proceedings of the 13th International Conference on Web Search and Data Mining*. 501–509.

[29] Tobias Schnabel, Adith Swaminathan, Ashudeep Singh, Navin Chandak, and Thorsten Joachims. 2016. Recommendations as treatments: Debiasing learning and evaluation. In *international conference on machine learning*. PMLR, 1670–1679.

[30] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. 2019. BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM international conference on information and knowledge management*. 1441–1450.

[31] Wenlong Sun, Sami Khenissi, Olfa Nasraoui, and Patrick Shafto. 2019. Debiasing the human-recommender system feedback loop in collaborative filtering. In *Companion Proceedings of The 2019 World Wide Web Conference*. 645–651.

[32] Jiaxi Tang and Ke Wang. 2018. Personalized top-n sequential recommendation via convolutional sequence embedding. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*. 565–573.

[33] Wilson L Taylor. 1953. "Cloze procedure": A new tool for measuring readability. *Journalism quarterly* 30, 4 (1953), 415–433.

[34] Saúl Vargas and Pablo Castells. 2011. Rank and relevance in novelty and diversity metrics for recommender systems. In *Proceedings of the fifth ACM conference on Recommender systems*. 109–116.

[35] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *arXiv preprint arXiv:1706.03762* (2017).

[36] Shoujin Wang, Longbing Cao, Yan Wang, Quan Z Sheng, Mehmet Orgun, and Defu Lian. 2019. A survey on session-based recommender systems. *arXiv preprint arXiv:1902.04864* (2019).

[37] Xiang Wang, Yaokun Xu, Xiangnan He, Yixin Cao, Meng Wang, and Tat-Seng Chua. 2020. Reinforced Negative Sampling over Knowledge Graph for Recommendation. In *Proceedings of The Web Conference 2020*. 99–109.

[38] Longqi Yang, Yin Cui, Yuan Xuan, Chenyang Wang, Serge Belongie, and Deborah Estrin. 2018. Unbiased offline recommender evaluation for missing-not-at-random implicit feedback. In *Proceedings of the 12th ACM Conference on Recommender Systems*. 279–287.

[39] Hsiang-Fu Yu, Mikhail Bilenko, and Chih-Jen Lin. 2017. Selection of negative samples for one-class matrix factorization. In *Proceedings of the 2017 SIAM International Conference on Data Mining*. SIAM, 363–371.

[40] Jianli Zhao, Shangcheng Yang, Huan Huo, Qiuxia Sun, and Xijiao Geng. 2021. TBTF: an effective time-varying bias tensor factorization algorithm for recommender system. *Applied Intelligence* (2021), 1–12.

[41] Pengyu Zhao, Tianxiao Shui, Yuanxing Zhang, Kecheng Xiao, and Kaigui Bian. 2020. Adversarial Oracular Seq2seq Learning for Sequential Recommendation. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI*. 1905–1911.

A SUPPLEMENTAL MATERIAL

A.1 Proof of Proposition 1

Proof.

$$\mathbb{E}[L_{Cloze}] = \mathbb{E}\Big[\frac{-1}{|S||I|T} \sum_{s=1}^{|S|} \sum_{t=1}^{T} \sum_{i=1}^{|I|} \mathbb{1}_{\{S_{s,t}^m = \langle mask \rangle\}} Y_{S_s,I_i,t}$$
$$\times log\, softmax(f_\Omega(S_{s,t}^m, I_i))\Big]$$
$$= \frac{-1}{|S||I|T} \sum_{s=1}^{|S|} \sum_{t=1}^{T} \sum_{i=1}^{|I|} \mathbb{1}_{\{S_{s,t}^m = \langle mask \rangle\}}$$
$$\times C_{S_s,I_i,t}\, \theta_{S_s,I_i,t}\, \gamma_{S_s,I_i,t} log\, softmax(f_\Omega(S_{s,t}^m, I_i))$$

Given that the temporal propensities $\theta_{S_s,I_i,t}$ cannot always be equal to 1, $\forall (S_s, I_i, t) \in S \times I \times [1..T]$. Thus, $\mathbb{E}[L_{Cloze}] \neq L_{Cloze}^{ideal}$. □

Note that the proof relies on the probabilistic model of the interaction random variable that is proposed later in Definition 4.

A.2 Proof of Proposition 2

Proof.
$$\mathbb{E}[L_{Cloze}^{IPS}] = L_{Cloze}^{ideal}$$
$$\Leftrightarrow \mathbb{E}\Big[\frac{-1}{|S||I|T} \sum_{s=1}^{|S|} \sum_{t=1}^{T} \sum_{i=1}^{|I|} \mathbb{1}_{\{S_{s,t}^m = \langle mask \rangle\}} \frac{Y_{S_s,I_i,t}}{\theta_{S_s,I_i}}$$
$$\times log\, softmax(f_\Omega(S_{s,t}^m, I_i))\Big]$$
$$= \frac{-1}{|S||I|T} \sum_{s=1}^{|S|} \sum_{t=1}^{T} \sum_{i=1}^{|I|} \mathbb{1}_{\{S_{s,t}^m = \langle mask \rangle\}} C_{S_s,I_i,t}$$
$$\times \gamma_{S_s,I_i,t} log\, softmax(f_\Omega(S_{s,t}^m, I_i))$$
$$\Leftrightarrow \mathbb{E}\Big[\frac{-1}{|S||I|T} \sum_{s=1}^{|S|} \sum_{t=1}^{T} \sum_{i=1}^{|I|} \mathbb{1}_{\{S_{s,t}^m = \langle mask \rangle\}} \frac{C_{S_s,I_i,t}\, O_{S_s,I_i,t}\, R_{S_s,I_i,t}}{\theta_{S_s,I_i}}$$
$$\times log\, softmax(f_\Omega(S_{s,t}^m, I_i))\Big]$$
$$= \frac{-1}{|S||I|T} \sum_{s=1}^{|S|} \sum_{t=1}^{T} \sum_{i=1}^{|I|} \mathbb{1}_{\{S_{s,t}^m = \langle mask \rangle\}} C_{S_s,I_i,t}$$
$$\times \gamma_{S_s,I_i,t} log\, softmax(f_\Omega(S_{s,t}^m, I_i))$$

$$\Leftrightarrow \frac{-1}{|S||I|T} \sum_{s=1}^{|S|} \sum_{t=1}^{T} \sum_{i=1}^{|I|} \mathbb{1}_{\{S_{s,t}^m = \langle mask \rangle\}} \frac{C_{S_s,I_i,t}\, \theta_{S_s,I_i,t}\, \gamma_{S_s,I_i,t}}{\theta_{S_s,I_i}}$$
$$\times log\, softmax(f_\Omega(S_{s,t}^m, I_i))$$
$$= \frac{-1}{|S||I|T} \sum_{s=1}^{|S|} \sum_{t=1}^{T} \sum_{i=1}^{|I|} \mathbb{1}_{\{S_{s,t}^m = \langle mask \rangle\}} C_{S_s,I_i,t}$$
$$\times \gamma_{S_s,I_i,t} log\, softmax(f_\Omega(S_{s,t}^m, I_i))$$
$$\Leftrightarrow \theta_{S_s,I_i,t} = \theta_{S_s,I_i}, \forall (S_s, I_i, t) \in S \times I \times [1..T]. \qquad \square$$

Note that the proof also relies on the probabilistic model of the interaction random variable that is proposed later in Definition 4.

A.3 Proof of Proposition 3

Proof.

$$\mathbb{E}[L_{Cloze}^{ITPS}] = \mathbb{E}\Big[\frac{-1}{|S||I|T} \sum_{s=1}^{|S|} \sum_{t=1}^{T} \sum_{i=1}^{|I|} \mathbb{1}_{\{S_{s,t}^m = \langle mask \rangle\}}$$
$$\times \frac{C_{S_s,I_i,t}\, O_{S_s,I_i,t}\, R_{S_s,I_i,t}}{\theta_{S_s,I_i,t}} log\, softmax(f_\Omega(S_{s,t}^m, I_i))\Big]$$
$$= \frac{-1}{|S||I|T} \sum_{s=1}^{|S|} \sum_{t=1}^{T} \sum_{i=1}^{|I|} \mathbb{1}_{\{S_{s,t}^m = \langle mask \rangle\}}$$
$$\times \frac{C_{S_s,I_i,t}\, \theta_{S_s,I_i,t}\, \gamma_{S_s,I_i,t}}{\theta_{S_s,I_i,t}} log\, softmax(f_\Omega(S_{s,t}^m, I_i))$$
$$= \frac{-1}{|S||I|T} \sum_{s=1}^{|S|} \sum_{t=1}^{T} \sum_{i=1}^{|I|} \mathbb{1}_{\{S_{s,t}^m = \langle mask \rangle\}} C_{S_s,I_i,t}\, \gamma_{S_s,I_i,t}$$
$$\times log\, softmax(f_\Omega(S_{s,t}^m, I_i)) = L_{Cloze}^{ideal} \qquad \square$$

Note that the proof assumes independence between exposure and relevance. Also, it assumes that the outcome of the choice model for an item is deterministic, which is reasonable if we assume a rational user who tends to choose the most relevant item among the exposed items.