Identifiability Matters: Revealing the Hidden Recoverable Condition in Unbiased Learning to Rank

Mouxiang Chen [1] **Chenghao Liu** [2] **Zemin Liu** [3] **Zhuo Li** [4] **Jianling Sun** [1]

Abstract

Unbiased Learning to Rank (ULTR) aims to train unbiased ranking models from biased click logs, by explicitly modeling a generation process for user behavior and fitting click data based on examination hypothesis. Previous research found empirically that the true latent relevance is mostly recoverable through click fitting. However, we demonstrate that this is not always achievable, resulting in a significant reduction in ranking performance. This research investigates the conditions under which relevance can be recovered from click data in the first principle. We initially characterize a ranking model as *identifiable* if it can recover the true relevance up to a scaling transformation, a criterion sufficient for the pairwise ranking objective. Subsequently, we investigate an equivalent condition for identifiability, articulated as a graph connectivity test problem: the recovery of relevance is feasible if and only if the *identifiability graph* (IG), derived from the underlying structure of the dataset, is connected. The presence of a disconnected IG may lead to degenerate cases and suboptimal ranking performance. To tackle this challenge, we introduce two methods, namely *node intervention* and *node merging*, designed to modify the dataset and restore the connectivity of the IG. Empirical results derived from a simulated dataset and two real-world LTR benchmark datasets not only validate our proposed theory but also demonstrate the effectiveness of our methods in alleviating data bias when the relevance model is unidentifiable.

[1]Zhejiang University [2]Salesforce Research Asia [3]National University of Singapore [4]State Street Technology (Zhejiang) Ltd. Correspondence to: Chenghao Liu <chenghao.liu@salesforce.com>, Zhuo Li <lizhuo@zju.edu.cn>.

Proceedings of the 41^{st} International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

1. Introduction

The utilization of click data for Learning to Rank (LTR) methodologies has become prevalent in contemporary information retrieval systems. The accumulated feedback effectively demonstrates the value of individual documents to users (Agarwal et al., 2019a) and is comparatively effortless to amass on an extensive scale. Nevertheless, inherent biases stemming from user behaviors are presented within these datasets (Joachims et al., 2007). One example is position bias (Joachims et al., 2005), wherein users exhibit a propensity to examine documents situated higher in the rankings, causing clicks to be biased with the position. Removing these biases is critical as they impede learning the correct ranking model from click data. To address this issue, Unbiased Learning to Rank (ULTR) is developed to mitigate these biases (Joachims et al., 2017). The central idea is to explicitly model a generation process for user clicks using the **examination hypothesis**. This hypothesis posits that each document has a probability of being observed (depending on certain bias factors, *e.g.*, position (Joachims et al., 2017) or context (Fang et al., 2019)), and subsequently clicked based on its relevance to the query (depending on ranking features encoding the query and document). It can be formulated as:

$$P(\text{click}) = \underbrace{P(O \mid \text{bias factors})}_{\text{observation model}} \cdot \underbrace{P(R \mid \text{ranking features})}_{\text{ranking model}}.$$

In practice, a joint optimization is utilized to optimize both the observation and ranking models based on the examination hypothesis (Wang et al., 2018; Ai et al., 2018a; Zhao et al., 2019; Guo et al., 2019). This approach efficiently accommodates the fitting of click data.

However, our focus extends beyond predicting the click probability; we are more interested in recovering the true relevance through the ranking model, which is the primary objective of ULTR. Previous empirical studies have suggested that this objective can be achieved in most cases after the click probability is fitted (Ai et al., 2018a; Wang et al., 2018; Ai et al., 2021). Regrettably, no existing literature provides a theoretical guarantee to substantiate this claim, and it is possible that the relevance becomes *unrecoverable* in certain cases. This was demonstrated in Oosterhuis (2022), which constructed a simplified dataset, revealing that a per-

fectly trained click model on this dataset can still produce inaccurate and inconsistent relevance estimates. Despite the limited scale of their example, we argue that in real scenarios with large-scale data, the unrecoverable phenomenon may persist, particularly in the presence of excessive bias factors. We illustrate it in the following example.

Example 1. *Consider a large-scale dataset where each displayed query-document pair comes with a* ***distinct*** *bias factor*[1]*. Unfortunately, in such cases, it becomes challenging to discern whether the influence on relevance arises from the ranking feature or the bias factor itself (Chen et al., 2022a; Zhu et al., 2020). One might train a naive ranker that predicts a constant one, coupled with an observation model that maps the unique bias factor to the true click probability. While the product of the two models yields an accurate click estimation, the estimated relevance is erroneous.*

Given the recent research emphasis on enhancing datasets by incorporating additional bias factors to refine observation estimation (Vardasbi et al., 2020; Chen et al., 2021; 2022a; 2023), we contend that there exists an urgent need to address a fundamental question: *under what circumstances can the relevance be recovered?* However, tackling this problem is notably challenging as it heavily relies on the exact data collection procedure (Oosterhuis, 2022). To the best of our knowledge, existing research has not yet presented the condition of relevance recovery at a fundamental level.

In this work, we present a general identifiability framework to address the core of relevance recovery. Importantly, this framework starts from the most basic examination hypothesis and does not impose additional requirements on the model implementation, optimization process, or specific types of bias. Focusing on the ranking objective, we define a ranking model as ***identifiable*** in § 3 if it can recover the true relevance probability *up to a scaling transformation*. The absence of identifiability can lead to degenerate scenarios and suboptimal ranking performance. We establish that the identifiability of a ranking model is linked to the underlying structure of the dataset, which can be translated into a practical graph connectivity test problem based on an *identifiability graph* (IG) related to the dataset. Specifically, if and only if the IG is connected, we can ensure the identifiability of the ranking model. Disconnections in the IG may result in inaccurate rankings. Theoretical and empirical analyses unveil that the identifiability probability is influenced by the dataset's size, bias factors, and features. Smaller datasets, more features, or an increased number of bias factors elevate the likelihood of IG disconnection. Furthermore, our analysis of the TianGong-ST dataset (Chen et al., 2019) indicates that the unidentifiability issue is present in the real world. These insights provide valuable guidance for the future design of ULTR datasets and algorithms.

Building upon the aforementioned theory, we delve into strategies for addressing datasets characterized by a disconnected IG. We introduce two methods aimed at enhancing IG connectivity to ensure the identifiability of ranking models: (1) *node intervention* (§ 4.1), which swaps documents between two bias factors to augment the dataset. While the intervention is common in ULTR (Joachims et al., 2017), our approach explores the minimal required number of interventions and significantly reduces the online cost; and (2) *node merging* (§ 4.2), which merges two bias factors and assumes identical observation probabilities. We conducted extensive experiments using a fully simulated dataset and two real-world LTR datasets to validate our theorems and demonstrate the efficacy of our methods when dealing with disconnected IGs.

To the best of our knowledge, we are the first to study the identifiability of ranking models for ULTR in the first principle. Our main contributions are summarized as follows:

1. We propose the concept of identifiability, ensuring the capacity to recover the true relevance from click data, and frame it as a graph connectivity test problem from the dataset perspective.
2. We propose model-agnostic methods to handle the unidentifiable cases by restoring graph connectivity.
3. We provide both theoretical guarantees and empirical studies to verify our proposed framework.

2. Preliminaries

Given a query $q \in \mathcal{Q}$, the goal of learning to rank is to learn a ranking model to sort a set of documents. Let $\boldsymbol{x} \in \mathcal{X}$ denote the query-document ranking features, and the ranking model aims to estimate a relevance score with $\boldsymbol{x}$. In practical scenarios where acquiring ground truth relevance at scale is challenging, users' click logs are frequently employed as labels for training the model (Joachims et al., 2005). While user clicks usually exhibit bias compared to the true relevance, researchers propose examination hypothesis to decompose the biased clicks into relevance probability and observation probability, formulated as:

$$c(\boldsymbol{x}, \boldsymbol{t}) = r(\boldsymbol{x}) \cdot o(\boldsymbol{t}), \tag{1}$$

where $c(\boldsymbol{x}, \boldsymbol{t})$, $r(\boldsymbol{x})$, and $o(\boldsymbol{t})$ denote the probabilities that the document is clicked, relevant, and observed, respectively. In this paper, we focus on the scenario in which the true observation probability o and relevance probability r are

[1] Achieving this is possible by incorporating sufficiently fine-grained bias factors identified in previous studies, such as positions, other document clicks (Vardasbi et al., 2020; Chen et al., 2021), contextual information (Fang et al., 2019), representation styles (Liu et al., 2015; Zheng et al., 2019), or various other contextual features (Ieong et al., 2012; Sun et al., 2020; Sarvi et al., 2023).

both *unknown*. $\boldsymbol{t} \in \mathcal{T}$ denotes bias factors that introduce clicks to be biased, such as position (Joachims et al., 2017), other document clicks (Vardasbi et al., 2020; Chen et al., 2021), contextual information (Fang et al., 2019) or the representation style (Liu et al., 2015). Let $\mathcal{D} = \{(\boldsymbol{x}_i, \boldsymbol{t}_i)\}_{i=1}^{|\mathcal{D}|}$ denote pairs of ranking features and bias factors. To simplify the analysis, we suppose all bias factors $\boldsymbol{t} \in \mathcal{T}$ and features $\boldsymbol{x} \in \mathcal{X}$ appear in $\mathcal{D}$. By explicitly modeling the bias via observation, we can attain an unbiased estimate of the ranking objective.

To jointly obtain the relevance score and observation score, we optimize a ranking model $r'(\cdot)$ and an observation model $o'(\cdot)$ to fit clicks in the form of the examination hypothesis. For example, two-tower (Guo et al., 2019) used the following objective:

$$\mathcal{L} = \sum_{i=1}^{|\mathcal{D}|} l(r'(\boldsymbol{x}_i) \cdot o'(\boldsymbol{t}_i), c_i), \tag{2}$$

where c_i denotes the click, and $l(\cdot, \cdot)$ denotes a loss function of interest, such as mean square error or cross-entropy error.

3. Identifiability

Most current work presumes that optimizing the two models in the form of Eq.(1) can yield an accurate ranking model (Wang et al., 2018; Chen et al., 2022a; 2023). Regrettably, there are no guarantees regarding this assumption. While it is established that the product of the outputs from the two models is accurate, there are instances, as demonstrated in Example 1, where this approach results in poor ranking performance. Our objective is to investigate a fundamental condition under which the underlying latent relevance function can be recovered (*i.e.,* **identifiable**), formulated as:

$$r(\boldsymbol{x}) \cdot o(\boldsymbol{t}) = r'(\boldsymbol{x}) \cdot o'(\boldsymbol{t}) \quad \Longrightarrow \quad r = r'.$$

In this paper, we intentionally prevent imposing specific constraints on the models of r' and o', making our approach broadly adaptable to existing work based on the examination hypothesis. It is worth noting that directly recovering an exact relevance model is impractical, as scaling $r(\cdot)$ by a factor of n and $o(\cdot)$ by $1/n$ would leave their product unchanged. In practice, we are often interested in making relevance identifiable up to a scaling transformation, which is sufficient for pairwise ranking objectives. Consequently, we introduce the following definition of identifiability:

Definition 1 (Identifiability)**.** *We say that the relevance model is identifiable, if:*

$$r(\boldsymbol{x}) \cdot o(\boldsymbol{t}) = r'(\boldsymbol{x}) \cdot o'(\boldsymbol{t}), \quad \forall (\boldsymbol{x}, \boldsymbol{t}) \in \mathcal{D}$$
$$\Longrightarrow \quad \exists C > 0, \ s.t. \ r(\boldsymbol{x}) = C r'(\boldsymbol{x}), \forall \boldsymbol{x} \in \mathcal{X}.$$

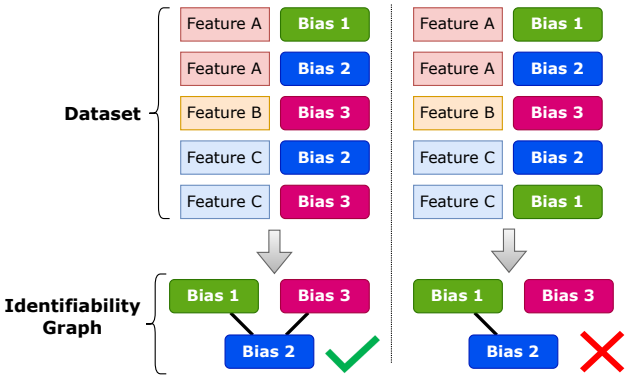


Figure 1. Examples for identifiable case and unidentifiable case.

Identifiability serves as a sufficient condition for ensuring accurate ranking. The absence of such guarantees can lead to degenerate scenarios, as illustrated in Example 1. Our following main result (with proof in Appendix A.1) establishes that identifiability is intrinsically linked to the dataset's underlying structure, which can be readily mined.

Theorem 1 (**Main result:** Equivalent condition of identifiability)**.** *The relevance model is identifiable, if and only if an undirected graph $G = (V, E)$ is connected, where V is a node set and E is an edge set, defined as:*

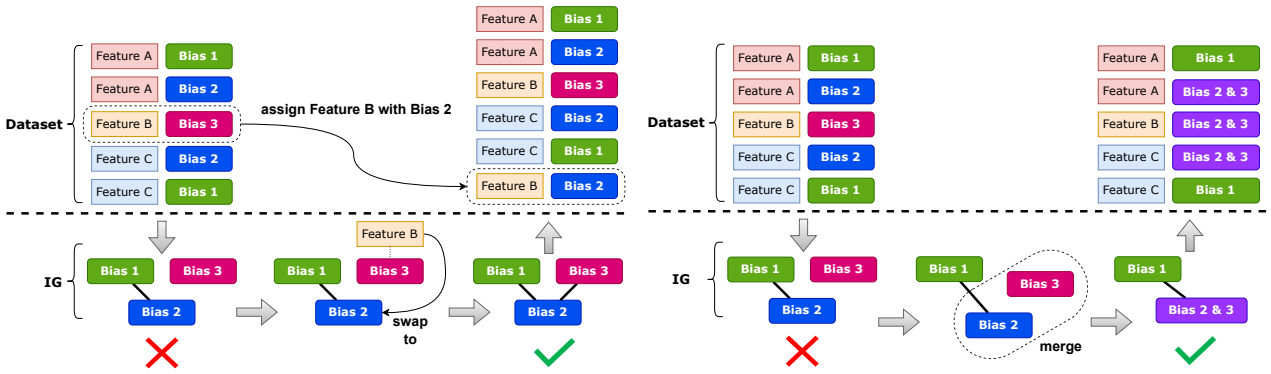
$$V = \{v_1, v_2, \cdots, v_{|\mathcal{T}|}\},$$
$$E = \{(v_s, v_t) \mid \exists \boldsymbol{x} \in \mathcal{X}, \ s.t. \ (\boldsymbol{x}, \boldsymbol{s}) \in \mathcal{D} \wedge (\boldsymbol{x}, \boldsymbol{t}) \in \mathcal{D}\},$$

We refer to this graph as ***identifiability graph*** *(IG).*

Remark 1. *The relevance identifiability is equivalent to a graph connectivity test problem. The IG is constructed as follows: we first create nodes for each bias factor $t \in \mathcal{T}$. If there exists a feature appearing with two bias factors together, add an edge between the two nodes. Theorem 1 establishes connections to recent ULTR research (Agarwal et al., 2019c; Oosterhuis, 2022; Zhang et al., 2023), which are elaborated in § 6.*

Figure 1 illustrates examples for applying Theorem 1 to verify the identifiability of the datasets. In the left figure, bias factors 1 and 2 are connected through feature 1, and bias factors 2 and 3 are connected through feature 3. As a result, the graph is connected and the relevance is identifiable. Conversely, the right figure depicts a scenario where bias factor 3 remains isolated, leading to unidentifiability of relevance. Based on Theorem 1, we illustrate the identifiability check algorithm in Appendix B.1.

Our next objective is to ascertain the probability of a ranking model being identifiable, particularly focusing on how it scales with the sizes $|\mathcal{D}|$, $|\mathcal{X}|$, and $|\mathcal{T}|$, to offer an intuitive understanding. However, accurately calculating this probability is intricate due to the uncertain generation process of



(a) node intervention (b) node merging

Figure 2. Illustrations for the proposed two methods to deal with unidentifiable datasets. In node intervention, we swap Feature B related to Bias 3 to Bias 2, which connects the two disconnected nodes. In node merging, we merge Bias 2 and Bias 3 into a new node 2 & 3, which indicates that Bias 2 and Bias 3 will have the same estimated observation. Both methods are applied to datasets before ULTR training.

$\mathcal{D}$. To address this, we consider a simplified distribution for $\mathcal{D}$ in the following example (proof in Appendix A.2).

Example 2 (Estimation of identifiability probability)**.** *Considering the following simplified example: each feature $\boldsymbol{x} \in \mathcal{X}$ and bias factor $\boldsymbol{t} \in \mathcal{T}$ are selected independently and uniformly to construct a dataset $\mathcal{D}$. Then the probability of identifiability can be estimated by:*

$$P(\textit{identifiability}) \sim 1 - |\mathcal{T}| \exp\left(-|\mathcal{D}| + f\right),$$
$$\textit{where } f = |\mathcal{X}||\mathcal{T}| \log\left[2 - \exp\left(-\frac{|\mathcal{D}|}{|\mathcal{X}||\mathcal{T}|}\right)\right].$$

Remark 2. *Note that $\lim_{|\mathcal{T}|\to+\infty} f = |\mathcal{D}|$. This implies a conclusion that when $|\mathcal{T}|$ is sufficiently large, the probability of identifiability decays, which requires a sufficiently large dataset $|\mathcal{D}|$ to offset this effect. While it is derived from a simplified case, this conclusion is consistent with our empirical evaluations in a more realistic setting (§ 5.2). Given the recent research focused on enhancing datasets with extra bias factors while keeping dataset size constant (Chen et al., 2021; 2023; Sarvi et al., 2023), these insights provide potential valuable guidance for future work.*

4. Dealing with unidentifiable dataset

In this section, we discuss how to deal with datasets whose relevance is unidentifiable. We propose two methods applied on datasets, namely *node intervention* and *node merging*, to establish connectivity within the IG. The former method necessitates the inclusion of supplementary data which enables the accurate recovery of relevance, while the latter method only needs existing data but may introduce approximation errors. In practice, the choice depends on the specific requirements of the problem.

4.1. Node intervention

Given that unidentifiability results from the incomplete dataset, one method is to augment datasets by swapping some pairs of documents between different bias factors (mainly positions). Swapping is a widely adopted technique that can yield accurate observation probabilities (Joachims et al., 2017). However, existing methods are often heuristic in nature, lacking a systematic theory to guide the swapping process, and sometimes performing unnecessary swapping operations. Instead, we leverage IGs to identify a minimal set of critical documents for effective swapping.

Our basic idea is that (1) find the connected components in the IG; (2) for any two components, find a node in each of them; (3) for these two nodes (representing two bias factors, *e.g.*, $\boldsymbol{t}_1$ and $\boldsymbol{t}_2$), find a feature (denoted by $\boldsymbol{x}$) related to one of them (denoted by $\boldsymbol{t}_1$), and swap it to the other (denoted by $\boldsymbol{t}_2$). This creates a new data point $(\boldsymbol{x}, \boldsymbol{t}_2)$. Since $(\boldsymbol{x}, \boldsymbol{t}_1)$ exists, this action bridges $\boldsymbol{t}_1$ and $\boldsymbol{t}_2$ in the IG and thus connects two components. Repeat this process until the IG is connected. We refer to this method as **node intervention**, which is illustrated in Figure 2(a).

It's worth mentioning that there are many choices to select the features and bias factors. To be specific, if we assume there are infinity data and the click probability $r(x)o(t)$ can be observed accurately, any intervention target that makes the IG connected can lead to identifiability, with no theoretical distinction in quality. However, note that we can only observe N samples of $r(x)o(t)$ in practice, which causes some choices to be less effective. For example, the observation probability of some bias factors is relatively low (*e.g.*, the last position in the list), necessitating more clicks to obtain a valid click rate estimation. These bias factors are not suitable for swapping.

Therefore, to choose the optimal intervention target, instead of simply assuming that we observe an accurate click probability $r(\boldsymbol{x}) \cdot o(\boldsymbol{t})$, here we assume that we can only observe a random variable for click rate which is an average of N clicks: $1/N \sum_{i=1}^{N} c_i$. $c_i \in \{0, 1\}$ is a binary random variable sampling from a probability $r(\boldsymbol{x}) \cdot o(\boldsymbol{t})$, indicating a click occurrence. Definition 1 can be seen as a special case when $N \to +\infty$. Based on it, we establish the following proposition, with its proof delegated to Appendix A.3.

Proposition 1. *For a feature $\boldsymbol{x}$ and two bias factors $\boldsymbol{t}_1, \boldsymbol{t}_2$, suppose $r'(\boldsymbol{x}) \cdot o'(\boldsymbol{t}) = 1/N \sum_{i=1}^{N} c_i(\boldsymbol{x}, \boldsymbol{t})$, where $\boldsymbol{t} \in \{\boldsymbol{t}_1, \boldsymbol{t}_2\}$ and $c_i(\boldsymbol{x}, \boldsymbol{t})$ are random variables i.i.d. sampled from Bernoulli$(r(\boldsymbol{x}) \cdot o(\boldsymbol{t}))$ for $1 \le i \le N$. Assuming $r(\boldsymbol{x})$, $o(\boldsymbol{t}_1)$ and $o(\boldsymbol{t}_2)$ are non-zero, then:*

$$\mathbb{E}\left[\Omega \mid r'(\boldsymbol{x})\right] = 0,$$
$$\mathbb{V}\left[\Omega \mid r'(\boldsymbol{x})\right] = \frac{1}{NR}\left[\frac{1}{r(\boldsymbol{x}) \cdot o(\boldsymbol{t}_1)} + \frac{1}{r(\boldsymbol{x}) \cdot o(\boldsymbol{t}_2)} - 2\right],$$

where $\Omega = \frac{o'(\boldsymbol{t}_1)}{o(\boldsymbol{t}_1)} - \frac{o'(\boldsymbol{t}_2)}{o(\boldsymbol{t}_2)}$ and $R = r'(\boldsymbol{x})^2 / r(\boldsymbol{x})^2$.

Remark 3. *The event Ω is closely related to the identifiability: As N or $r(\boldsymbol{x})o(\boldsymbol{t})$ increases, the variance $\mathbb{V}$ of Ω decreases, leading to $\Omega \to 0$ and $o'(\boldsymbol{t})/o(\boldsymbol{t})$ approaching a constant, which indicates an identifiable ranking model according to Definition 1. In practice, optimal feature and bias factors can be chosen to minimize the variance and reduce the required N.*

Based on Proposition 1, for two components $G_A = (V_A, E_A)$ and $G_B = (V_B, E_B)$, we use the following process to connect G_A and G_B, by minimizing $\mathbb{V}$ to decreasing the necessary clicks and facilitate identifiability:

$$\mathcal{C}(\boldsymbol{x}, \boldsymbol{t}_1, \boldsymbol{t}_2) = \frac{1}{r(\boldsymbol{x}) \cdot o(\boldsymbol{t}_1)} + \frac{1}{r(\boldsymbol{x}) \cdot o(\boldsymbol{t}_2)} - 2, \tag{3}$$

$$\boldsymbol{t}_A^{(A)}, \boldsymbol{t}_B^{(A)}, \boldsymbol{x}_A = \underset{\boldsymbol{t}_A \in V_A, \boldsymbol{t}_B \in V_B, \boldsymbol{x} \in X_{t_A}}{\arg\min} \mathcal{C}(\boldsymbol{x}, \boldsymbol{t}_A, \boldsymbol{t}_B), \tag{4}$$

$$\boldsymbol{t}_A^{(B)}, \boldsymbol{t}_B^{(B)}, \boldsymbol{x}_B = \underset{\boldsymbol{t}_A \in V_A, \boldsymbol{t}_B \in V_B, \boldsymbol{x} \in X_{t_B}}{\arg\min} \mathcal{C}(\boldsymbol{x}, \boldsymbol{t}_A, \boldsymbol{t}_B), \tag{5}$$

$$\boldsymbol{x}^*, \boldsymbol{t}^* = \begin{cases} \boldsymbol{x}_A, \boldsymbol{t}_B^{(A)} & \text{if } \mathcal{C}(\boldsymbol{x}_A, \boldsymbol{t}_A^{(A)}, \boldsymbol{t}_B^{(A)}) \le \mathcal{C}(\boldsymbol{x}_B, \boldsymbol{t}_A^{(B)}, \boldsymbol{t}_B^{(B)}), \\ \boldsymbol{x}_B, \boldsymbol{t}_A^{(B)} & \text{otherwise,} \end{cases} \tag{6}$$

where $X_{t_A} = \{\boldsymbol{x}_i \mid (\boldsymbol{x}_i, \boldsymbol{t}_A) \in \mathcal{D}\}$ and $X_{t_B} = \{\boldsymbol{x}_i \mid (\boldsymbol{x}_i, \boldsymbol{t}_B) \in \mathcal{D}\}$. Here Eq.(3) defines a cost[2] of swapping $\boldsymbol{x}$ from $\boldsymbol{t}_1$ to $\boldsymbol{t}_2$ (or from $\boldsymbol{t}_2$ to $\boldsymbol{t}_1$) based on $\mathbb{V}$ derived by Proposition 1. We ignore R since it is a constant when the relevance model is identifiable. Eq.(4) defines the process that we find a feature $\boldsymbol{x}_A$ related to a bias factor $\boldsymbol{t}_A^{(A)}$ (belongs to G_A) and swap it to another bias factor $\boldsymbol{t}_B^{(A)}$ (belongs to G_B). Reversely, Eq.(5) defines the process to swap the feature from G_B to G_A. The final decision depends on the process with less cost (Eq.(6)). We refer to this cost as the *intervention cost* between G_A and G_B. Finally, we add $(\boldsymbol{x}^*, \boldsymbol{t}^*)$ to the dataset $\mathcal{D}$ and collect enough user clicks about it, which connects G_A and G_B together.

To extend the above process of connecting two components to the entire IG, consider a graph G comprising K connected components: $G = G_1 \cup \cdots \cup G_K$. We first construct another complete graph with K nodes, each representing a component in G. Intervention costs between components G_i and G_j serve as edge weights in the constructed complete graph. A Minimum Spanning Tree (MST) algorithm is applied to this complete graph to find edges with minimal total weights for connection. Subsequent interventions (Eq.(3) - Eq.(6)) are performed on these edges to connect the entire IG with the minimal total intervention cost. The comprehensive algorithm is detailed in Appendix B.2.

Compared to traditional intervention strategies that often necessitate random swapping across all queries (Joachims et al., 2017; Radlinski & Joachims, 2006; Carterette & Chandar, 2018; Yue et al., 2010; Wang et al., 2018), node intervention requires only $K - 1$ swaps for an IG with K connected components. Notably, K is generally much smaller than the total query count, particularly when positions are the sole bias factors, which is the usual focus of traditional intervention strategies. Consequently, node intervention markedly reduces online interventions and improves the user experience.

4.2. Node merging

Despite node intervention being effective in achieving identifiability, it still requires additional online experiments, which can be time-consuming and may pose a risk of impeding user experience by displaying irrelevant documents at the top of the ranking list. What's worse, some types of bias factors may not be appropriate for swapping (*e.g.*, contextual information or other documents' clicks). Therefore, we propose another simple and general methodology for addressing the unidentifiability issue, which involves merging nodes from different connected components and forcing them to have the same observation prediction. We refer to this strategy as **node merging**, which is illustrated in Figure 2(b).

Similar to node intervention, there are numerous options for selecting node pairs to merge. Note that merging two dissimilar nodes with distinct observation probabilities will inevitably introduce approximation errors, as stated in the following proposition (We defer the proof to Appendix A.4):

Proposition 2 (Error bound of merging two components)**.** *Suppose an IG $G = (V, E)$ consists of two connected com-*

[2]The value of r and o in Eq.(3) can be based on rational estimations. For instance, r can be chosen from a ranking model trained on biased clicks, and a manually crafted model can serve as o. Additionally, the node merging method, which we will soon introduce, may be utilized to generate initial estimates for r and o.

ponents $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$. If we merge two nodes $v_1 \in G_1$ and $v_2 \in G_2$ by forcing $o'(\boldsymbol{t}') = o'(\boldsymbol{t}'')$ where v_1 and v_2 represent bias factors $\boldsymbol{t}'$ and $\boldsymbol{t}''$, then:

$$r(\boldsymbol{x}) \cdot o(\boldsymbol{t}) = r'(\boldsymbol{x}) \cdot o'(\boldsymbol{t})$$
$$\Longrightarrow \left| \frac{r'(\boldsymbol{x}_1)}{r(\boldsymbol{x}_1)} - \frac{r'(\boldsymbol{x}_2)}{r(\boldsymbol{x}_2)} \right| \leq \left| \frac{o(\boldsymbol{t}') - o(\boldsymbol{t}'')}{o'(\boldsymbol{t}')} \right|, \forall \boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{X},$$

where we suppose r, r', o and o' are not zero.

Remark 4. *When $o(\boldsymbol{t}') = o(\boldsymbol{t}'')$, the relevance model is identifiable. If the gap between $o(\boldsymbol{t}')$ and $o(\boldsymbol{t}'')$ is large, ${r'(\boldsymbol{x})}/{r(\boldsymbol{x})}$ will change greatly, hurting the performance.*

Therefore, we propose to merge similar nodes exhibiting minimal differences in their observation probabilities. Suppose each bias factor $\boldsymbol{t}$ can be represented using a feature vector $\boldsymbol{X_t}$ (namely bias features). We further assume that vectors with greater similarity correspond to closer observation probabilities. As a simple example, we can consider the number of positions as a 1-dimensional bias feature, since it is reasonable that documents in proximate positions will exhibit similar observation probabilities.

Based on it, we use the following process to connect two components $G_A = (V_A, E_A)$ and $G_B = (V_B, E_B)$:

$$\mathcal{C}(\boldsymbol{t}_1, \boldsymbol{t}_2) = ||\boldsymbol{X}_{\boldsymbol{t}_1} - \boldsymbol{X}_{\boldsymbol{t}_2}||, \tag{7}$$
$$\boldsymbol{t}_A^*, \boldsymbol{t}_B^* = \arg\min_{t_A \in V_A, t_B \in V_B} \mathcal{C}(\boldsymbol{t}_A, \boldsymbol{t}_B). \tag{8}$$

Here Eq.(7) defines the merging cost to merge $\boldsymbol{t}_1$ and $\boldsymbol{t}_2$. Eq.(8) finds two bias factors $\boldsymbol{t}_A^*$ and $\boldsymbol{t}_B^*$ from two components that have the minimal merging cost. We refer to this cost as the merging cost between G_A and G_B. Analogous to node intervention, the MST algorithm is employed to connect the IG, with the sole distinction being the definition of edge weights in the complete graph as merging costs. The comprehensive algorithm is detailed in Appendix B.3.

Furthermore, we present two key properties of node merging in Appendix A.5: (1) *Consistency*: the merging constraints imposed on o' are compatible with the preconditions for identifiability (*i.e.*, $r(\boldsymbol{x}) \cdot o(\boldsymbol{t}) = r'(\boldsymbol{x}) \cdot o'(\boldsymbol{t})$), ensuring that click probabilities can still be accurately fitted after applying node merging; and (2) *Error bound*: extended from Proposition 2, the error for node merging is bounded by the diameter of the constructed MST.

Compared to node intervention, node merging performs on the offline dataset, making it a simple and time-efficient approach. However, merging bias factors brings additional approximation error which has the potential to adversely impact the ranking performance.

5. Experiments

In this section, we describe our experimental setup and show the empirical results, in both the fully synthetic setting and large-scale study. [3]

5.1. Fully synthetic study

Dataset To verify the correctness of Theorem 1, and the effectiveness of proposed methods, we first conducted experiments on a fully synthetic dataset, which allowed for precise control of the connectivity of IGs. We generated four datasets with different numbers K of connected components within each IG ($K = 1, 2, 3, 4$), as illustrated in Appendix C.1. The bias factors only consist of positions (*i.e.*, position-based model or PBM). We defer the click simulation setup to Appendix C.2.

Baselines For comparative analysis, we evaluated our methods on several baselines: *No debias*, which trains the ranking model using click data without an observation model, and three widely-used ULTR optimization algorithms based on examination hypothesis, *DLA* (Ai et al., 2018a), *Regression-EM* (Wang et al., 2018) and *Two-Tower* (Guo et al., 2019). Notably, many recent models (Vardasbi et al., 2020; Chen et al., 2021; Sarvi et al., 2023; Chen et al., 2020; 2022a; 2023) are variants of these three ULTR algorithms, primarily varying in their bias factor handling. We will discuss the influence of bias factors on identifiability in the next section (§ 5.2). Training details for the baselines can be found in Appendix C.3. Our *node intervention* and *node merging*, being model-agnostic and training-independent, are applied to datasets before training.

Evaluation metrics To evaluate the performance of the methods, we computed the mean correlation coefficient (**MCC**) between the true relevance probability $r(\cdot)$ and the predicted relevance probability $r'(\cdot)$, defined as

$$\frac{\sum_{i=1}^{|\mathcal{D}|} \left(r(\boldsymbol{x}_i) - \overline{r(\boldsymbol{x})}\right) \left(r'(\boldsymbol{x}_i) - \overline{r'(\boldsymbol{x})}\right)}{\sqrt{\sum_{i=1}^{|\mathcal{D}|} \left(r(\boldsymbol{x}_i) - \overline{r(\boldsymbol{x})}\right)^2} \sqrt{\sum_{i=1}^{|\mathcal{D}|} \left(r'(\boldsymbol{x}_i) - \overline{r'(\boldsymbol{x})}\right)^2}}.$$

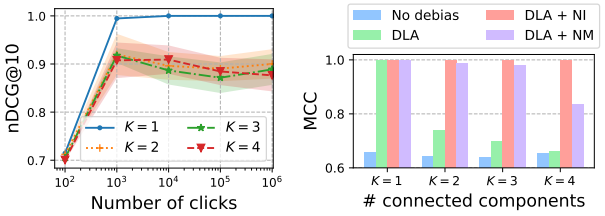
A high MCC means that we successfully identified the true model and recovered the true relevance up to a scaling transformation. We also computed **nDCG**, which are standard ranking metrics prevalently used in LTR, and **Click MSE**, which is the mean squared error between the true and predicted click probability for evaluating the fitting goodness.

Analysis: How does the connectivity of IGs impact the ranking performance? Figure 3(a) shows the effects of varying numbers of connected components K within IGs on ranking performance (using DLA), with different numbers of clicks. Here, $K = 1$ indicates a connected IG. We

[3] Code is available at `https://github.com/Keytoyze/ULTR-identifiability`

Table 1. Performance of different methods on the $K = 2$ simulation dataset under PBM bias. We ran each experiment 10 times and reported the average results as well as the standard deviations.

Method	MCC ↑	nDCG@1 ↑	nDCG@3 ↑	nDCG@5 ↑	nDCG@10 ↑	Click MSE
No debias	$0.521_{\pm.000}$	$0.711_{\pm.000}$	$0.625_{\pm.000}$	$0.665_{\pm.000}$	$0.820_{\pm.000}$	2×10^{-5}
DLA	$0.707_{\pm.105}$	$0.836_{\pm.061}$	$0.742_{\pm.091}$	$0.789_{\pm.070}$	$0.886_{\pm.040}$	$< 10^{-8}$
+ Node intervention	$\mathbf{1.000_{\pm.000}}$	$\mathbf{1.000_{\pm.000}}$	$\mathbf{1.000_{\pm.000}}$	$\mathbf{1.000_{\pm.000}}$	$\mathbf{1.000_{\pm.000}}$	$< 10^{-8}$
+ Node merging	$0.975_{\pm.000}$	$\mathbf{1.000_{\pm.000}}$	$\mathbf{1.000_{\pm.000}}$	$\mathbf{1.000_{\pm.000}}$	$\mathbf{1.000_{\pm.000}}$	$< 10^{-8}$
Regression-EM	$0.580_{\pm.117}$	$0.786_{\pm.035}$	$0.677_{\pm.063}$	$0.752_{\pm.044}$	$0.857_{\pm.027}$	$< 10^{-8}$
+ Node intervention	$\mathbf{0.980_{\pm.023}}$	$0.999_{\pm.001}$	$0.995_{\pm.010}$	$0.989_{\pm.023}$	$0.997_{\pm.006}$	$< 10^{-7}$
+ Node merging	$0.975_{\pm.000}$	$\mathbf{1.000_{\pm.000}}$	$\mathbf{1.000_{\pm.000}}$	$\mathbf{1.000_{\pm.000}}$	$\mathbf{1.000_{\pm.000}}$	$< 10^{-8}$
Two-Tower	$0.830_{\pm.050}$	$0.883_{\pm.034}$	$0.832_{\pm.054}$	$0.857_{\pm.045}$	$0.925_{\pm.022}$	$< 10^{-8}$
+ Node intervention	$\mathbf{1.000_{\pm.000}}$	$\mathbf{1.000_{\pm.000}}$	$\mathbf{1.000_{\pm.000}}$	$\mathbf{1.000_{\pm.000}}$	$\mathbf{1.000_{\pm.000}}$	$< 10^{-8}$
+ Node merging	$0.975_{\pm.000}$	$\mathbf{1.000_{\pm.000}}$	$\mathbf{1.000_{\pm.000}}$	$\mathbf{1.000_{\pm.000}}$	$\mathbf{1.000_{\pm.000}}$	$< 10^{-8}$



(a) DLA (b) Different methods

Figure 3. (a) Performance of DLA on different numbers of connected components K across different click counts. Shadowed areas depict variance. (b) Performance of different methods across different K. NI = Node Intervention. NM = Node Merging.

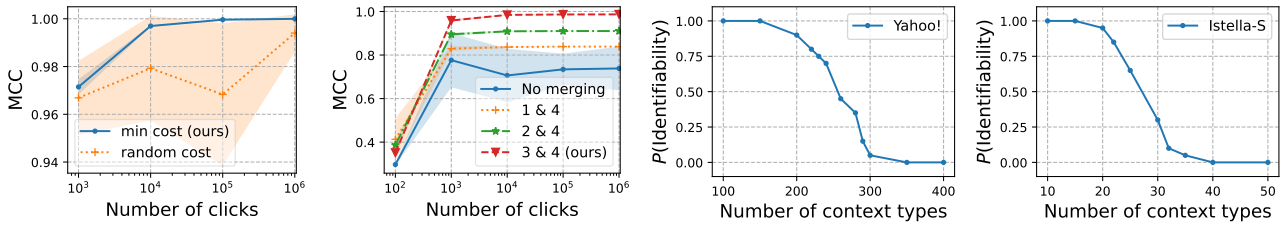
can observe that the ranking model is capable of achieving perfect ranking accuracy only if the IG is connected. Otherwise, the performance exhibits significant instability and poor quality, regardless of the number of clicks collected. Besides, larger Ks (*e.g.*, when $K > 1$) do not give rise to significant differences. These observations serve to validate the correctness of Theorem 1.

Analysis: Can our proposed two methods handle unidentifiable datasets? We tested the different methods in the $K = 2$ scenario and summarized the results in Table 1. Our methods consistently achieved nearly perfect ranking accuracy, markedly surpassing the baselines. Notably, various ULTR algorithms tend to yield disparate suboptimal performances in unidentifiable cases, despite perfectly fitting click probability as indicated by Click MSE. However, with identifiability established through our approaches, they all converge to a common set of effective parameters that accurately recover relevance. Furthermore, node intervention recovers a more accurate relevance than node merging, evidenced by the MCC. This shows the trade-offs between the two proposed methods: compared to node merging, node intervention does not introduce approximation errors, but requires an additional step to augment the dataset.

Analysis: How does the methods' performance degrade with decreased IG connectivity? We evaluated DLA and our approaches under varying the value of K. Figure 3(b) illustrates that both DLA and node merging show performance declines as K increases, while node merging exhibits a slower rate. Remarkably, node intervention sustains a perfect relevance recovery ability. We also noted the node merging performance for $K = 4$ is suboptimal compared to $K = 3$ and $K = 2$, attributed to merging two relatively divergent nodes in $K = 4$. A comparative analysis of observation estimation in Appendix D.1 provides an in-depth discussion of this suboptimal performance.

Ablation study: Impact of different selection strategies for node intervention. In our ablation study of the node intervention method within the $K = 2$ scenario (using DLA), we explored a variation termed *random cost*. Here, the cost function (Eq.(3)) was modeled to follow a uniform distribution over $[0, 1]$ and unrelated to $\boldsymbol{x}$ and $\boldsymbol{t}$, which leads to random selection of intervention pairs (Eq.(4)-Eq.(6)). The original method is denoted as *min cost*. As observed in Figure 4(a), the *random cost* approach exhibits notably higher variance than *min cost* and requires sufficient clicks to obtain a stable performance, which confirms the validity of Proposition 1. Additionally, despite its variance, *random cost* attains a commendable performance level relative to the unidentifiable baseline, reinforcing the significance of maintaining a connected IG.

Ablation study: Impact of different merging strategies for node merging. Similarly, we conducted an ablation study for the node merging method, based on DLA within the $K = 2$ scenario. We use three different merging strategies, where a & b represents merging nodes corresponding to the position a and b: 1 & 4, 2 & 4, and 3 & 4. Note that all of the strategies ensure a connected IG (See Figure 5 in Appendix C.1 for details), and 3 & 4 is the proposed node merging strategy. Figure 4(b) demonstrates that performance improves as the merging nodes are closer, thereby



(a) Node intervention (b) Node merging (c) Yahoo! (d) Istella-S

Figure 4. (a) Impact of cost selection strategies in node intervention. (b) Impact of merging strategies in node merging. (c)(d) Impact of the number of context types on the identifiability probabilities on the two datasets.

validating Proposition 2.

5.2. Large-scale study

Dataset We also performed another empirical study on the large-scale semi-synthetic setup that is prevalent in unbiased learning to rank (Joachims et al., 2017; Ai et al., 2021; Chen et al., 2022a) on two widely used benchmark datasets: Yahoo! LETOR (Chapelle & Chang, 2011) and Istella-S (Lucchese et al., 2016). We provide further details for them in Appendix C.1. In both datasets, only the top 10 documents were considered to be displayed. In addition to positions, we also incorporated context types as another bias factor that is prevalent in recent research (*i.e.*, contextual position-based model or CPBM) (Fang et al., 2019). A random context type was assigned to each query-document pair. Furthermore, we conducted identifiability testing on the TianGong-ST (Chen et al., 2019), a large-scale real-world dataset with an abundance of genuine bias factors.

Analysis: Does the unidentifiability issue exist in the real world? We first applied the identifiability check algorithm on TianGong-ST and found that when accounting for **positions** and all provided **vertical types** as bias factors (a total of 20,704), the IG of this dataset is *disconnected*: there are 2,900 connected components within it. This observation suggests that the unidentifiability phenomenon could occur in the real world, even on a large-scale dataset. We further excluded certain bias factors from that dataset and assessed its identifiability, which is elaborated in Appendix D.2.

Analysis: How do the number of bias factors $|\mathcal{T}|$, dataset scale $|\mathcal{D}|$ and feature count $|\mathcal{X}|$ affect the identifiability? We tuned the number of context types within Yahoo! and Istella-S and computed the frequency with which the IG was connected to determine the identifiability probability. From Figure 4(c) and 4(d), it can be observed that if positions are the only bias factors, both datasets are identifiable for the ranking model. However, upon the consideration of context types, the identifiability probability drastically decreases as the number of context types increases. Merely 20 (on Istella-S) or 200 (on Yahoo!) are sufficient to render the IG disconnected. When the number is further increased, it becomes exceedingly challenging to obtain an identifiable ranking model, which aligns with the conclusion of Example 2. We also experimented with investigating the impact of dataset scale and feature count on identifiability, elaborated in Appendix D.3.

Analysis: How does node merging perform under contextual bias? We simulated 5,000 context types on Yahoo! and Istella-S to evaluate the efficacy of node merging. To eliminate the influence of initialization, all predicted observation probabilities are initially set to 1.0. In such cases, node intervention is not practical since context types cannot be swapped. Results in both the training and test sets in Table 2 show that node merging correctly manages unidentifiable cases and restores relevance, significantly surpassing the baselines in MCC and nDCG.

Analysis: How does model initialization impact performance? In unidentifiable scenarios, models can converge to various parameters that predict click probability correctly, but only a limited range truly represents correct relevance. Therefore, model initialization notably affects performance in unidentifiable scenarios. We detail it in Appendix D.4.

6. Related work

Unbiased learning to rank (ULTR) Unbiased learning to rank (ULTR) tries to directly learn unbiased ranking models from biased clicks. The core of ULTR lies in the estimation of observation probabilities, which is typically achieved through intervention (Wang et al., 2016; Joachims et al., 2017). These methods are related to the node intervention we proposed in § 4.1, but they are prone to useless swapping operations and negatively impact the user's experience. To avoid intervention, Agarwal et al. (2019c) and Fang et al. (2019) proposed intervention harvest methods that exploit click logs with multiple ranking models. These methods are related to Theorem 1, but they did not delve into the identifiability conditions. Our proposed theory bridges the

Table 2. Performance (with the standard deviations) comparison on two datasets under CPBM bias.

Dataset	Method	Training				Test	
		MCC ↑	nDCG@5 ↑	nDCG@10 ↑	Click MSE	nDCG@5 ↑	nDCG@10 ↑
Yahoo	No debias	$0.765_{\pm.000}$	$0.841_{\pm.000}$	$0.915_{\pm.000}$	4×10^{-4}	$0.693_{\pm.002}$	$0.741_{\pm.001}$
	DLA	$0.750_{\pm.000}$	$0.844_{\pm.000}$	$0.914_{\pm.000}$	2×10^{-5}	$0.693_{\pm.001}$	$0.741_{\pm.001}$
	DLA + Node merging	$\mathbf{0.771_{\pm.000}}$	$\mathbf{0.853_{\pm.000}}$	$\mathbf{0.920_{\pm.000}}$	4×10^{-5}	$\mathbf{0.697_{\pm.001}}$	$\mathbf{0.745_{\pm.001}}$
Istella-S	No debias	$0.764_{\pm.000}$	$0.885_{\pm.000}$	$0.941_{\pm.000}$	4×10^{-5}	$0.634_{\pm.001}$	$0.682_{\pm.001}$
	DLA	$0.764_{\pm.000}$	$0.886_{\pm.000}$	$0.941_{\pm.000}$	1×10^{-6}	$0.633_{\pm.001}$	$0.682_{\pm.001}$
	DLA + Node merging	$\mathbf{0.772_{\pm.000}}$	$\mathbf{0.892_{\pm.000}}$	$\mathbf{0.944_{\pm.000}}$	2×10^{-5}	$\mathbf{0.636_{\pm.001}}$	$\mathbf{0.684_{\pm.001}}$

gap between these two groups of work by determining when intervention is necessary or not. Recently, some researchers proposed to jointly estimate relevance and bias, including IPS-based methods (Wang et al., 2018; Ai et al., 2018a; Hu et al., 2019; Jin et al., 2020) and two-tower based models (Zhao et al., 2019; Guo et al., 2019; Haldar et al., 2020; Yan et al., 2022). These models are based on the examination hypothesis and are optimized to maximize user click likelihood, therefore our proposed framework can be applied to these models as well.

On the other side, researchers developed models to extend the scope of bias factors that affect observation probabilities, which contain position (Wang et al., 2018; Ai et al., 2018a; Hu et al., 2019; Ai et al., 2021), contextual information (Fang et al., 2019; Tian et al., 2020), clicks in the same query list (Vardasbi et al., 2020; Chen et al., 2021), presentation style (Zheng et al., 2019; Liu et al., 2015; Chen et al., 2023), search intent (Sun et al., 2020), result domain (Ieong et al., 2012), ranking features (Chen et al., 2022a) and outliers (Sarvi et al., 2023). While incorporating additional bias factors is beneficial for improving the estimation of accurate observation probabilities (Chen et al., 2023), as we mention in Example 2 and § 5.2, an excessive number of bias factors may pose a risk of unidentifiability.

Relevance recovery in ULTR The identifiability condition (Theorem 1) establishes connections and generalizations to recent ULTR research. Agarwal et al. (2019c) constructed intervention sets to uncover documents that are put at two different positions to estimate observation probabilities, which, however, did not further explore the recoverable conditions. Oosterhuis (2022) also showed that a perfect click model can provide incorrect relevance estimates and the estimation consistency depends on the data collection procedure. We take a further step by delving into the root cause and digesting the concrete condition based on the data. Zhang et al. (2023) found that some features (*e.g.*, with high relevance) are more likely to be assigned with specific bias factors (*e.g.*, top positions). This phenomenon results in a decline in performance, called confounding bias. This bias is related to the identifiability issue since when a severe confounding bias is present, the IG is more likely to be disconnected due to insufficient data coverage.

Identifiability Identifiability is a fundamental concept in various machine learning fields, such as independent component analysis (Hyvarinen & Morioka, 2017; Hyvarinen et al., 2019), latent variable models (Allman et al., 2009; Guillaume et al., 2019; Khemakhem et al., 2020), missing not at random data (Ma & Zhang, 2021; Miao et al., 2016) and causal discovery (Addanki et al., 2021; Peters et al., 2011; Spirtes & Zhang, 2016). It defines a model's capacity to recover some unique latent structure, variables, or causal relationships from the data. In this work, we embrace the commonly used notion of identifiability and apply its definition to the ULTR domain.

7. Conclusions

In this paper, we take the first step to exploring if and when relevance can be recovered from click data from a foundational perspective. We first define the identifiability of a ranking model, which refers to the ability to recover relevance probabilities up to a scaling transformation. Our research reveals that (1) the ranking model is not always identifiable, which depends on the underlying structure of the dataset (*i.e.*, an identifiability graph should be connected); (2) identifiability depends on the data size, the number of bias factors and features, and unidentifiability issue is possible on large-scale real-world datasets; (3) two methods, node intervention and node merging, can be utilized to address the unidentifiability issues. Our proposed framework is theoretically and empirically verified.

Limitations and future work (1) While examination hypothesis (Eq.(1)) is the most widely used hypothesis, other models like trust bias (Agarwal et al., 2019b) and vector-based examination hypothesis (Chen et al., 2022b; Yan et al., 2022) also merit attention. Adapting our graph-based approach to them is a promising area for future research. (2) While the assumption of perfect data fitting is typical for identifiability theory in machine learning (Hyvarinen et al., 2019; Khemakhem et al., 2020), investigating the propagation of approximation errors within the identifiability graph is an interesting future direction in ULTR.

Impact statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none of which we feel must be specifically highlighted here.

Acknowledgments

Research work mentioned in this paper is supported by State Street Zhejiang University Technology Center. We would also like to thank Lefei Shen for the assistance in the experiments and thank Yusu Hong, Yu Mou, Shanda Li, and Wencheng Cai for the valuable discussion of the theory.

References

Addanki, R., McGregor, A., and Musco, C. Intervention efficient algorithms for approximate learning of causal graphs. In *Algorithmic Learning Theory*, pp. 151–184. PMLR, 2021.

Agarwal, A., Takatsu, K., Zaitsev, I., and Joachims, T. A general framework for counterfactual learning-to-rank. In *SIGIR 2019*, pp. 5–14, 2019a.

Agarwal, A., Wang, X., Li, C., Bendersky, M., and Najork, M. Addressing trust bias for unbiased learning-to-rank. In *TheWebConf 2019*, pp. 4–14, 2019b.

Agarwal, A., Zaitsev, I., Wang, X., Li, C., Najork, M., and Joachims, T. Estimating position bias without intrusive interventions. In *WSDM 2019*, pp. 474–482, 2019c.

Ai, Q., Bi, K., Luo, C., Guo, J., and Croft, W. B. Unbiased learning to rank with unbiased propensity estimation. In *SIGIR 2018*, pp. 385–394, 2018a.

Ai, Q., Mao, J., Liu, Y., and Croft, W. B. Unbiased learning to rank: Theory and practice. In *CIKM 2018*, pp. 2305–2306, New York, NY, USA, 2018b. ACM. ISBN 978-1-4503-6014-2. doi: 10.1145/3269206.3274274.

Ai, Q., Yang, T., Wang, H., and Mao, J. Unbiased learning to rank: Online or offline? *TOIS*, 39(2):1–29, 2021.

Allman, E. S., Matias, C., and Rhodes, J. A. Identifiability of parameters in latent structure models with many observed variables. *The Annals of Statistics*, 37(6A):3099 – 3132, 2009. doi: 10.1214/09-AOS689. URL https://doi.org/10.1214/09-AOS689.

Carterette, B. and Chandar, P. Offline comparative evaluation with incremental, minimally-invasive online feedback. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pp. 705–714, 2018.

Chapelle, O. and Chang, Y. Yahoo! learning to rank challenge overview. In *Proceedings of the learning to rank challenge*, pp. 1–24. PMLR, 2011.

Chapelle, O., Metlzer, D., Zhang, Y., and Grinspan, P. Expected reciprocal rank for graded relevance. In *CIKM 2009*, pp. 621–630, 2009.

Chen, J., Mao, J., Liu, Y., Zhang, M., and Ma, S. Tiangongst: A new dataset with large-scale refined real-world web search sessions. In *CIKM 2019*. ACM, 2019.

Chen, J., Mao, J., Liu, Y., Zhang, M., and Ma, S. A context-aware click model for web search. In *WSDM 2020*, pp. 88–96, 2020.

Chen, M., Liu, C., Sun, J., and Hoi, S. C. Adapting interactional observation embedding for counterfactual learning to rank. In *SIGIR 2021*, pp. 285–294, 2021.

Chen, M., Liu, C., Liu, Z., and Sun, J. Lbd: Decouple relevance and observation for individual-level unbiased learning to rank. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A. (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 33400–33413. Curran Associates, Inc., 2022a.

Chen, M., Liu, C., Liu, Z., and Sun, J. Scalar is not enough: Vectorization-based unbiased learning to rank. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 136–145, 2022b.

Chen, X., Li, X., Wei, K., Hu, B., Jiang, L., Huang, Z., and Kang, Z. Multi-feature integration for perception-dependent examination-bias estimation. *arXiv preprint arXiv:2302.13756*, 2023.

Fang, Z., Agarwal, A., and Joachims, T. Intervention harvesting for context-dependent examination-bias estimation. In *SIGIR 2019*, pp. 825–834, 2019.

Gilbert, E. N. Random graphs. *The Annals of Mathematical Statistics*, 30(4):1141–1144, 1959.

Guillaume, J. H., Jakeman, J. D., Marsili-Libelli, S., Asher, M., Brunner, P., Croke, B., Hill, M. C., Jakeman, A. J., Keesman, K. J., Razavi, S., et al. Introductory overview of identifiability analysis: A guide to evaluating whether you have the right type of data for your modeling purpose. *Environmental Modelling & Software*, 119:418–432, 2019.

Guo, H., Yu, J., Liu, Q., Tang, R., and Zhang, Y. Pal: a position-bias aware learning framework for ctr prediction in live recommender systems. In *Proceedings of the 13th ACM Conference on Recommender Systems*, pp. 452–456, 2019.

Haldar, M., Ramanathan, P., Sax, T., Abdool, M., Zhang, L., Mansawala, A., Yang, S., Turnbull, B., and Liao, J. Improving deep learning for airbnb search. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2822–2830, 2020.

Hu, Z., Wang, Y., Peng, Q., and Li, H. Unbiased lambdamart: An unbiased pairwise learning-to-rank algorithm. In *TheWebConf 2019*, pp. 2830–2836, 2019.

Hyvarinen, A. and Morioka, H. Nonlinear ica of temporally dependent stationary sources. In *Artificial Intelligence and Statistics*, pp. 460–469. PMLR, 2017.

Hyvarinen, A., Sasaki, H., and Turner, R. Nonlinear ica using auxiliary variables and generalized contrastive learning. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 859–868. PMLR, 2019.

Ieong, S., Mishra, N., Sadikov, E., and Zhang, L. Domain bias in web search. In *WSDM 2012*, pp. 413–422, 2012.

Jin, J., Fang, Y., Zhang, W., Ren, K., Zhou, G., Xu, J., Yu, Y., Wang, J., Zhu, X., and Gai, K. A deep recurrent survival model for unbiased ranking. In *SIGIR 2020*, pp. 29–38, 2020.

Joachims, T. Training linear svms in linear time. In *KDD 2006*, pp. 217–226, 2006.

Joachims, T., Granka, L., Pan, B., Hembrooke, H., and Gay, G. Accurately interpreting clickthrough data as implicit feedback. In *SIGIR 2005*, pp. 154–161, 2005.

Joachims, T., Granka, L., Pan, B., Hembrooke, H., Radlinski, F., and Gay, G. Evaluating the accuracy of implicit feedback from clicks and query reformulations in web search. *TOIS*, 25(2):7–es, 2007.

Joachims, T., Swaminathan, A., and Schnabel, T. Unbiased learning-to-rank with biased feedback. In *WSDM 2017*, pp. 781–789, 2017.

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., and Liu, T.-Y. Lightgbm: A highly efficient gradient boosting decision tree. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Khemakhem, I., Kingma, D., Monti, R., and Hyvarinen, A. Variational autoencoders and nonlinear ica: A unifying framework. In *International Conference on Artificial Intelligence and Statistics*, pp. 2207–2217. PMLR, 2020.

Liu, Z., Liu, Y., Zhou, K., Zhang, M., and Ma, S. Influence of vertical result in web search examination. In *SIGIR 2015*, pp. 193–202, 2015.

Lucchese, C., Nardini, F. M., Orlando, S., Perego, R., Silvestri, F., and Trani, S. Post-learning optimization of tree ensembles for efficient ranking. In *SIGIR 2016*, pp. 949–952, 2016.

Ma, C. and Zhang, C. Identifiable generative models for missing not at random data imputation. *Advances in Neural Information Processing Systems*, 34:27645–27658, 2021.

Miao, W., Ding, P., and Geng, Z. Identifiability of normal and normal mixture models with nonignorable missing data. *Journal of the American Statistical Association*, 111 (516):1673–1683, 2016.

Oosterhuis, H. Reaching the end of unbiasedness: Uncovering implicit limitations of click-based learning to rank. In *Proceedings of the 2022 ACM SIGIR International Conference on Theory of Information Retrieval*, pp. 264–274, 2022.

Peters, J., Mooij, J. M., Janzing, D., and Schölkopf, B. Identifiability of causal graphs using functional models. In *Proceedings of the Twenty-Seventh Conference on Uncertainty in Artificial Intelligence*, pp. 589–598, 2011.

Radlinski, F. and Joachims, T. Minimally invasive randomization for collecting unbiased preferences from clickthrough logs. In *Proceedings of the national conference on artificial intelligence*, volume 21, pp. 1406. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2006.

Sarvi, F., Vardasbi, A., Aliannejadi, M., Schelter, S., and de Rijke, M. On the impact of outlier bias on user clicks. *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2023.

Spirtes, P. and Zhang, K. Causal discovery and inference: concepts and recent methodological advances. In *Applied informatics*, volume 3, pp. 1–28. SpringerOpen, 2016.

Sun, Y., Kolacinski, R., and Loparo, K. Eliminating search intent bias in learning to rank. In *ICSC*, pp. 108–115. IEEE, 2020.

Tian, M., Guo, C., Ostuni, V., and Zhu, Z. Counterfactual learning to rank using heterogeneous treatment effect estimation. *arXiv:2007.09798*, 2020.

Vardasbi, A., de Rijke, M., and Markov, I. Cascade model-based propensity estimation for counterfactual learning to rank. *SIGIR 2020*, Jul 2020. doi: 10.1145/3397271.3401299.

Wang, X., Bendersky, M., Metzler, D., and Najork, M. Learning to rank with selection bias in personal search. In *SIGIR 2016*, pp. 115–124, 2016.

Wang, X., Golbandi, N., Bendersky, M., Metzler, D., and Najork, M. Position bias estimation for unbiased learning to rank in personal search. In *WSDM 2018*, pp. 610–618, 2018.

Yan, L., Qin, Z., Zhuang, H., Wang, X., Bendersky, M., and Najork, M. Revisiting two-tower models for unbiased learning to rank. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 2410–2414, 2022.

Yue, Y., Patel, R., and Roehrig, H. Beyond position bias: Examining result attractiveness as a source of presentation bias in clickthrough data. In *WWW 2010*, pp. 1011–1018, 2010.

Zhang, Y., Yan, L., Qin, Z., Zhuang, H., Shen, J., Wang, X., Bendersky, M., and Najork, M. Towards disentangling relevance and bias in unbiased learning to rank. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 5618–5627, 2023.

Zhao, Z., Hong, L., Wei, L., Chen, J., Nath, A., Andrews, S., Kumthekar, A., Sathiamoorthy, M., Yi, X., and Chi, E. Recommending what video to watch next: a multitask ranking system. In *Proceedings of the 13th ACM Conference on Recommender Systems*, pp. 43–51, 2019.

Zheng, Y., Mao, J., Liu, Y., Luo, C., Zhang, M., and Ma, S. Constructing click model for mobile search with viewport time. *TOIS*, 37(4):1–34, 2019.

Zhu, Z., He, Y., Zhang, Y., and Caverlee, J. Unbiased implicit recommendation and propensity estimation via combinational joint learning. In *Proceedings of the 14th ACM Conference on Recommender Systems*, RecSys '20, pp. 551–556, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450375832. doi: 10.1145/3383313.3412210. URL https://doi.org/10.1145/3383313.3412210.

Appendix

A. Theoretical results

A.1. Proof for Theorem 1

Step 1. We first prove the "if" part: assume that

$$r(\boldsymbol{x}) \cdot o(\boldsymbol{t}) = r'(\boldsymbol{x}) \cdot o'(\boldsymbol{t}) \quad \forall (\boldsymbol{x}, \boldsymbol{t}) \in \mathcal{D}, \tag{A.1}$$

and G is connected, our goal is to prove that ${}^{r(\boldsymbol{x})}/_{r'(\boldsymbol{x})} =$ constant. Note that we only consider the nontrivial case $r(\boldsymbol{x}) \neq 0$ and $r'(\boldsymbol{x}) \neq 0$. Otherwise, C can be any positive number.

For any two bias factors $\boldsymbol{s} \in \mathcal{T}$ and $\boldsymbol{t} \in \mathcal{T}$, since G is connected, there exists a path $v_{\boldsymbol{a}_1} \to v_{\boldsymbol{a}_2} \to \cdots \to v_{\boldsymbol{a}_n}$ in G where $v_{\boldsymbol{a}_1}, \cdots, v_{\boldsymbol{a}_n}$ are the nodes in the identifiability graph representing different bias factors, and $\boldsymbol{a}_1 = \boldsymbol{s}, \boldsymbol{a}_n = \boldsymbol{t}$. Consider a middle edge $v_{\boldsymbol{a}_m} \to v_{\boldsymbol{a}_{m+1}} (1 \leq m \leq n-1)$, according to the definition of the edge,

$$\exists \boldsymbol{x} \in \mathcal{X}, \text{ s.t. } (\boldsymbol{x}, \boldsymbol{a}_m) \in \mathcal{D} \wedge (\boldsymbol{x}, \boldsymbol{a}_{m+1}) \in \mathcal{D}. \tag{A.2}$$

According to Eq.(A.1) and Eq.(A.2), we have $r(\boldsymbol{x}) \cdot o(\boldsymbol{a}_m) = r'(\boldsymbol{x}) \cdot o'(\boldsymbol{a}_m)$ and $r(\boldsymbol{x}) \cdot o(\boldsymbol{a}_{m+1}) = r'(\boldsymbol{x}) \cdot o'(\boldsymbol{a}_{m+1})$, and therefore

$$\frac{o'(\boldsymbol{a}_m)}{o(\boldsymbol{a}_m)} = \frac{r(\boldsymbol{x})}{r'(\boldsymbol{x})} = \frac{o'(\boldsymbol{a}_{m+1})}{o(\boldsymbol{a}_{m+1})}. \tag{A.3}$$

Let $f(\boldsymbol{x}) = {}^{r(\boldsymbol{x})}/_{r'(\boldsymbol{x})}$ and $g(\boldsymbol{t}) = {}^{o'(\boldsymbol{t})}/_{o(\boldsymbol{t})}$. Applying Eq.(A.3) to the path $v_{\boldsymbol{a}_1} \to v_{\boldsymbol{a}_2} \to \cdots \to v_{a_n}$, we obtain $g(\boldsymbol{s}) = g(\boldsymbol{t})$. Given that $\boldsymbol{s}$ and $\boldsymbol{t}$ are selected arbitrarily, we have $g(\boldsymbol{t}) =$ constant for all bias factors $\boldsymbol{t}$. Since $f(\boldsymbol{x}) = g(\boldsymbol{t})$ holds for all $(\boldsymbol{x}, \boldsymbol{t}) \in \mathcal{D}$ according to Eq.(A.1), $f(\boldsymbol{x})$ is also constant.

Step 2. We then prove the "only if" part: assume that the relevance is identifiable, and prove that G is connected. We prove this by contradiction: Given a disconnected IG G, our goal is to prove that the ranking model is unidentifiable, by showing that we can construct two click models such that the click probabilities are equal, yet the inside relevance models differ.

Since $G = (V, E)$ is disconnected, we suppose G can be divided into two disjoint graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$. Note that we do not require G_1 or G_2 to be connected, therefore this division is always feasible even when G has more than two components. Based on G_1 and G_2, we can divide the dataset $\mathcal{D}$ into two disjoint sets D_1 and D_2: $\mathcal{D}_1 = \{(\boldsymbol{x}, \boldsymbol{t}) \mid v_{\boldsymbol{t}} \in V_1\}$ and $\mathcal{D}_2 = \{(\boldsymbol{x}, \boldsymbol{t}) \mid v_{\boldsymbol{t}} \in V_2\}$. Let $\mathcal{X}_1 = \{\boldsymbol{x} \mid (\boldsymbol{x}, \boldsymbol{t}) \in \mathcal{D}_1\}$ denote features in $\mathcal{D}_1$, and $\mathcal{X}_2 = \{\boldsymbol{x} \mid (\boldsymbol{x}, \boldsymbol{t}) \in \mathcal{D}_2\}$ denote features in $\mathcal{D}_2$. Note that $\mathcal{X}_1$ and $\mathcal{X}_2$ are disjoint, *i.e.*, $\mathcal{X}_1 \cap \mathcal{X}_2 = \varnothing$, otherwise according to the definition of the edge set, there exists an edge between V_1 and V_2 which connects G_1 and G_2.

Next, given any relevance function r and observation function o, we define r' and o' as follows.

$$r'(\boldsymbol{x}) = \begin{cases} \alpha r(\boldsymbol{x}) & \text{if } \boldsymbol{x} \in \mathcal{X}_1, \\ \beta r(\boldsymbol{x}) & \text{if } \boldsymbol{x} \in \mathcal{X}_2, \end{cases}$$

$$o'(\boldsymbol{t}) = \begin{cases} {}^{o(\boldsymbol{t})}/_{\alpha} & \text{if } v_{\boldsymbol{t}} \in V_1, \\ {}^{o(\boldsymbol{t})}/_{\beta} & \text{if } v_{\boldsymbol{t}} \in V_2, \end{cases}$$

where $\alpha \neq \beta$ are two positive numbers. Note that if $(\boldsymbol{x}, \boldsymbol{t}) \in \mathcal{D}_1$, then $\boldsymbol{x} \in \mathcal{X}_1$ and $v_{\boldsymbol{t}} \in V_1$, therefore $r'(\boldsymbol{x})o'(\boldsymbol{t}) = \alpha r(\boldsymbol{x}) \cdot {}^{o(\boldsymbol{t})}/_{\alpha} = r(\boldsymbol{x})o(\boldsymbol{t})$. If $(\boldsymbol{x}, \boldsymbol{t}) \in \mathcal{D}_2$, then $\boldsymbol{x} \in \mathcal{X}_2$ and $v_{\boldsymbol{t}} \in V_2$, therefore $r'(\boldsymbol{x})o'(\boldsymbol{t}) = \beta r(\boldsymbol{x}) \cdot {}^{o(\boldsymbol{t})}/_{\beta} = r(\boldsymbol{x})o(\boldsymbol{t})$. Based on it, Eq.(A.1) holds for all $(\boldsymbol{x}, \boldsymbol{t}) \in \mathcal{D}$. However, it is obvious that C isn't constant in $r(\boldsymbol{x}) = Cr'(\boldsymbol{x})$, since $C = \alpha$ when $\boldsymbol{x} \in \mathcal{X}_1$ and $C = \beta$ otherwise. It indicates that the relevance model isn't identifiable.

A.2. Proof for Example 2

We begin by estimating the disconnected probability between two nodes in the identifiability graph, as the following lemma.

Lemma A.1. *In an identifiability graph, the probability of two nodes $v_{\boldsymbol{s}}$ and $v_{\boldsymbol{t}}$ are disconnected can be estimated as:*

$$P(\textit{disconnected} \mid \mathcal{D}, \boldsymbol{s}, \boldsymbol{t}) \sim \exp\left(-\frac{|\mathcal{D}|}{|\mathcal{T}|}\right)\left[2 - \exp\left(-\frac{|\mathcal{D}|}{|\mathcal{X}||\mathcal{T}|}\right)\right]^{|\mathcal{X}|},$$

when $|\mathcal{X}||\mathcal{T}| \to \infty$.

Proof. Let $P(\boldsymbol{x}, \boldsymbol{s})$ and $P(\boldsymbol{x}, \boldsymbol{t})$ denote the probabilities of selecting $(\boldsymbol{x}, \boldsymbol{s})$ and $(\boldsymbol{x}, \boldsymbol{t})$ respectively. We have:

$$\begin{aligned}
P(\text{disconnected} \mid \mathcal{D}, \boldsymbol{s}, \boldsymbol{t}) &= P\left(\bigcap_{\boldsymbol{x} \in \mathcal{X}} (\boldsymbol{x}, \boldsymbol{s}) \notin \mathcal{D} \vee (\boldsymbol{x}, \boldsymbol{t}) \notin \mathcal{D}\right) \\
&= \prod_{\boldsymbol{x} \in \mathcal{X}} P\left((\boldsymbol{x}, \boldsymbol{s}) \notin \mathcal{D} \vee (\boldsymbol{x}, \boldsymbol{t}) \notin \mathcal{D}\right) \\
&= \prod_{\boldsymbol{x} \in \mathcal{X}} 1 - P\left((\boldsymbol{x}, \boldsymbol{s}) \in \mathcal{D}\right) \cdot P\left((\boldsymbol{x}, \boldsymbol{t}) \in \mathcal{D}\right) \\
&= \prod_{\boldsymbol{x} \in \mathcal{X}} 1 - \left(1 - P\left((\boldsymbol{x}, \boldsymbol{s}) \notin \mathcal{D}\right)\right) \cdot \left(1 - P\left((\boldsymbol{x}, \boldsymbol{t}) \notin \mathcal{D}\right)\right).
\end{aligned}$$

Note that $P\left((\boldsymbol{x}, \boldsymbol{t}) \notin \mathcal{D}\right)$ is the probability that $(\boldsymbol{x}, \boldsymbol{t})$ is not sampled for $|\mathcal{D}|$ times, we have $P\left((\boldsymbol{x}, \boldsymbol{s}) \notin \mathcal{D}\right) = \left[1 - P(\boldsymbol{x}, \boldsymbol{s})\right]^{|\mathcal{D}|}$ and $P\left((\boldsymbol{x}, \boldsymbol{t}) \notin \mathcal{D}\right) = \left[1 - P(\boldsymbol{x}, \boldsymbol{t})\right]^{|\mathcal{D}|}$, therefore,

$$P(\text{disconnected} \mid \mathcal{D}, \boldsymbol{s}, \boldsymbol{t}) = \prod_{\boldsymbol{x} \in \mathcal{X}} 1 - \left(1 - \left[1 - P(\boldsymbol{x}, \boldsymbol{s})\right]^{|\mathcal{D}|}\right)\left(1 - \left[1 - P(\boldsymbol{x}, \boldsymbol{t})\right]^{|\mathcal{D}|}\right).$$

Using the condition that features and bias factors are sampled independently and uniformly, we have $P(\boldsymbol{x}, \boldsymbol{s}) = P(\boldsymbol{x}, \boldsymbol{t}) = {}^{1}/_{|\mathcal{X}||\mathcal{T}|}$. Therefore,

$$\begin{aligned}
P(\text{disconnected} \mid \mathcal{D}, \boldsymbol{s}, \boldsymbol{t}) &= \left\{1 - \left[1 - \left(1 - \frac{1}{|\mathcal{X}||\mathcal{T}|}\right)^{|\mathcal{D}|}\right]^2\right\}^{|\mathcal{X}|} \\
&= \left\{1 - \left[1 - \left(1 - \frac{1}{|\mathcal{X}||\mathcal{T}|}\right)^{-|\mathcal{X}||\mathcal{T}| \cdot \frac{-|\mathcal{D}|}{|\mathcal{X}||\mathcal{T}|}}\right]^2\right\}^{|\mathcal{X}|} \\
&\sim \left\{1 - \left[1 - \exp\left(-\frac{|\mathcal{D}|}{|\mathcal{X}||\mathcal{T}|}\right)\right]^2\right\}^{|\mathcal{X}|} \\
&= \left[2\exp\left(-\frac{|\mathcal{D}|}{|\mathcal{X}||\mathcal{T}|}\right) - \exp\left(-\frac{2|\mathcal{D}|}{|\mathcal{X}||\mathcal{T}|}\right)\right]^{|\mathcal{X}|} \\
&= \exp\left(-\frac{|\mathcal{D}|}{|\mathcal{T}|}\right)\left[2 - \exp\left(-\frac{|\mathcal{D}|}{|\mathcal{X}||\mathcal{T}|}\right)\right]^{|\mathcal{X}|},
\end{aligned}$$

where the third line uses $(1 + {}^{1}/_{n})^n \to e$ when $n \to \infty$.

□

We next provide a lemma to estimate the probability that a random graph is connected.

Lemma A.2. *(Gilbert, 1959) Suppose a graph G is constructed from a set of N nodes in which each one of the ${}^{N(N-1)}/_{2}$ possible links is present with probability p independently. The probability that G is connected can be estimated as:*

$$P(\textit{connected} \mid G) \sim 1 - N(1-p)^{N-1}.$$

Applying Theorem 2, Lemma A.1 and Lemma A.2, we obtain:

$$\begin{aligned}
P(\text{identifiability} \mid \mathcal{D}) &\sim 1 - |\mathcal{T}| \left\{ \exp\left(-\frac{|\mathcal{D}|}{|\mathcal{T}|}\right) \left[2 - \exp\left(-\frac{|\mathcal{D}|}{|\mathcal{X}||\mathcal{T}|}\right)\right]^{|\mathcal{X}|} \right\}^{|\mathcal{T}|-1} \\
&= 1 - |\mathcal{T}| \exp\left[-|\mathcal{D}|\left(1 - \frac{1}{|\mathcal{T}|}\right)\right] \left[2 - \exp\left(-\frac{|\mathcal{D}|}{|\mathcal{X}||\mathcal{T}|}\right)\right]^{|\mathcal{X}|(|\mathcal{T}|-1)} \\
&\sim 1 - |\mathcal{T}| \exp\left(-|\mathcal{D}|\right) \left[2 - \exp\left(-\frac{|\mathcal{D}|}{|\mathcal{X}||\mathcal{T}|}\right)\right]^{|\mathcal{X}||\mathcal{T}|} \\
&= 1 - |\mathcal{T}| \exp\left(-|\mathcal{D}| + |\mathcal{X}||\mathcal{T}| \log\left[2 - \exp\left(-\frac{|\mathcal{D}|}{|\mathcal{X}||\mathcal{T}|}\right)\right]\right)
\end{aligned}$$

where the third line uses $1/|\mathcal{T}| \to 0$ and $|\mathcal{T}| - 1 \to |\mathcal{T}|$ when $|\mathcal{T}|$ is large enough.

A.3. Proof for Proposition 1

Note that $Nr'(\boldsymbol{x})o'(\boldsymbol{t})$ follows a binomial distribution, *i.e.*,

$$Nr'(\boldsymbol{x})o'(\boldsymbol{t}) \sim B(N, r(\boldsymbol{x})o(\boldsymbol{t})),$$

which implies:

$$\mathbb{E}[Nr'(\boldsymbol{x})o'(\boldsymbol{t})] = Nr(\boldsymbol{x})o(\boldsymbol{t}), \quad \mathbb{V}[Nr'(\boldsymbol{x})o'(\boldsymbol{t})] = Nr(\boldsymbol{x})o(\boldsymbol{t})[1 - r(\boldsymbol{x})o(\boldsymbol{t})].$$

Denote $g(\boldsymbol{t}) = {o'(\boldsymbol{t})}/{o(\boldsymbol{t})}$, then we have:

$$\begin{aligned}
\mathbb{E}[g(\boldsymbol{t}) \mid r'(\boldsymbol{x})] &= \frac{Nr(\boldsymbol{x})o(\boldsymbol{t})}{Nr'(\boldsymbol{x})o(\boldsymbol{t})} = \frac{r(\boldsymbol{x})}{r'(\boldsymbol{x})}, \\
\mathbb{V}[g(\boldsymbol{t}) \mid r'(\boldsymbol{x})] &= \frac{Nr(\boldsymbol{x})o(\boldsymbol{t})[1 - r(\boldsymbol{x})o(\boldsymbol{t})]}{[Nr'(\boldsymbol{x})o(\boldsymbol{t})]^2} = \frac{r(\boldsymbol{x})(1 - r(\boldsymbol{x})o(\boldsymbol{t}))}{Nr'(\boldsymbol{x})^2 o(\boldsymbol{t})}.
\end{aligned}$$

Since $c(\boldsymbol{x}, \boldsymbol{t})$ are sampled i.i.d., $g(\boldsymbol{t})$ is independent of $g(\boldsymbol{t}')$ conditioned on $r'(\boldsymbol{x})$. Therefore,

$$\begin{aligned}
\mathbb{E}[g(\boldsymbol{t}_1) - g(\boldsymbol{t}_2) \mid r'(\boldsymbol{x})] &= \frac{r(\boldsymbol{x})}{r'(\boldsymbol{x})} - \frac{r(\boldsymbol{x})}{r'(\boldsymbol{x})} = 0, \\
\mathbb{V}[g(\boldsymbol{t}_1) - g(\boldsymbol{t}_2) \mid r'(\boldsymbol{x})] &= \mathbb{V}[g(\boldsymbol{t}_1) \mid r'(\boldsymbol{x})] + \mathbb{V}[g(\boldsymbol{t}_2) \mid r'(\boldsymbol{x})] \\
&= \frac{r(\boldsymbol{x})(1 - r(\boldsymbol{x})o(\boldsymbol{t}_1))}{Nr'(\boldsymbol{x})^2 o(\boldsymbol{t}_1)} + \frac{r(\boldsymbol{x})(1 - r(\boldsymbol{x})o(\boldsymbol{t}_2))}{Nr'(\boldsymbol{x})^2 o(\boldsymbol{t}_2)} \\
&= \frac{r(\boldsymbol{x})^2}{Nr'(\boldsymbol{x})^2} \left[\frac{1 - r(\boldsymbol{x})o(\boldsymbol{t}_1)}{r(\boldsymbol{x})o(\boldsymbol{t}_1)} + \frac{1 - r(\boldsymbol{x})o(\boldsymbol{t}_2)}{r(\boldsymbol{x})o(\boldsymbol{t}_2)}\right] \\
&= \frac{1}{NR} \left[\frac{1}{r(\boldsymbol{x})o(\boldsymbol{t}_1)} + \frac{1}{r(\boldsymbol{x})o(\boldsymbol{t}_2)} - 2\right].
\end{aligned}$$

A.4. Proof for Proposition 2

We first separate the dataset $\mathcal{D}$ into two parts: $\mathcal{D}_1$ (corresponding to G_1) and $\mathcal{D}_2$ (corresponding to G_2), formally,

$$\begin{aligned}
\mathcal{D}_1 &= \{(\boldsymbol{x}, \boldsymbol{t}) \in D \mid \boldsymbol{t} \in V_1\}, \\
\mathcal{D}_2 &= \{(\boldsymbol{x}, \boldsymbol{t}) \in D \mid \boldsymbol{t} \in V_2\}.
\end{aligned}$$

According to Theorem 1, the relevance model $r(\boldsymbol{x})$ ($\boldsymbol{x} \in \{\boldsymbol{x}_i \mid (\boldsymbol{x}_i, \boldsymbol{t}_i) \in \mathcal{D}_1\}$) is identifiable on the dataset $\mathcal{D}_1$, and the relevance model $r(\boldsymbol{x})$ ($\boldsymbol{x} \in \{\boldsymbol{x}_i \mid (\boldsymbol{x}_i, \boldsymbol{t}_i) \in \mathcal{D}_2\}$) is identifiable on the dataset $\mathcal{D}_2$. That is,

$$\begin{aligned}
\frac{r'(\boldsymbol{x}_a)}{r(\boldsymbol{x}_a)} &= \frac{r'(\boldsymbol{x}_b)}{r(\boldsymbol{x}_b)}, \quad \forall \boldsymbol{x}_a, \boldsymbol{x}_b \in \{\boldsymbol{x}_i \mid (\boldsymbol{x}_i, \boldsymbol{t}_i) \in \mathcal{D}_1\}, \\
\frac{r'(\boldsymbol{x}_c)}{r(\boldsymbol{x}_c)} &= \frac{r'(\boldsymbol{x}_d)}{r(\boldsymbol{x}_d)}, \quad \forall \boldsymbol{x}_c, \boldsymbol{x}_d \in \{\boldsymbol{x}_i \mid (\boldsymbol{x}_i, \boldsymbol{t}_i) \in \mathcal{D}_2\}.
\end{aligned} \tag{A.4}$$

Since we have assumed that $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ appear in $\mathcal{D}$, we can find $\boldsymbol{t}_1$ and $\boldsymbol{t}_2$ such that $(\boldsymbol{x}_1, \boldsymbol{t}_1) \in \mathcal{D}$ and $(\boldsymbol{x}_2, \boldsymbol{t}_2) \in \mathcal{D}$.

(1) If $\boldsymbol{t}_1 \in V_1 \wedge \boldsymbol{t}_2 \in V_1$, or $\boldsymbol{t}_1 \in V_2 \wedge \boldsymbol{t}_2 \in V_2$, then according to Eq.(A.4),

$$\left| \frac{r'(\boldsymbol{x}_1)}{r(\boldsymbol{x}_1)} - \frac{r'(\boldsymbol{x}_2)}{r(\boldsymbol{x}_2)} \right| = 0. \tag{A.5}$$

(2) Otherwise, without loss of generality we suppose $\boldsymbol{t}_1 \in V_1 \wedge \boldsymbol{t}_2 \in V_2$. For t' and t'', we can find $\boldsymbol{x}'$ and $\boldsymbol{x}''$ such that $(\boldsymbol{x}', \boldsymbol{t}') \in \mathcal{D}_1$ and $(\boldsymbol{x}'', \boldsymbol{t}'') \in \mathcal{D}_2$. According to Eq.(A.4),

$$\frac{r'(\boldsymbol{x}_1)}{r(\boldsymbol{x}_1)} = \frac{r'(\boldsymbol{x}')}{r(\boldsymbol{x}')}, \quad \frac{r'(\boldsymbol{x}_2)}{r(\boldsymbol{x}_2)} = \frac{r'(\boldsymbol{x}'')}{r(\boldsymbol{x}'')}.$$

Since

$$\frac{r'(\boldsymbol{x}')}{r(\boldsymbol{x}')} = \frac{o(\boldsymbol{t}')}{o'(\boldsymbol{t}')}, \quad \frac{r'(\boldsymbol{x}'')}{r(\boldsymbol{x}'')} = \frac{o(\boldsymbol{t}'')}{o'(\boldsymbol{t}'')},$$

we have

$$\left| \frac{r'(\boldsymbol{x}_1)}{r(\boldsymbol{x}_1)} - \frac{r'(\boldsymbol{x}_2)}{r(\boldsymbol{x}_2)} \right| = \left| \frac{o(\boldsymbol{t}')}{o'(\boldsymbol{t}')} - \frac{o(\boldsymbol{t}'')}{o'(\boldsymbol{t}'')} \right| = \left| \frac{o(\boldsymbol{t}') - o(\boldsymbol{t}'')}{o'(\boldsymbol{t}')} \right|, \tag{A.6}$$

where we use the fact that $o'(\boldsymbol{t}') = o'(\boldsymbol{t}'')$.

Combining Eq.(A.5) and Eq.(A.6) we obtain the desired result.

A.5. Further theoretical analysis on node merging

In this section, we provide further theoretical analysis on node merging, including the consistency guarantee and the error bound. We first formally introduce the Minimum Spanning Tree (MST) construction process of node merging, laying the groundwork for the subsequent analysis. Suppose an IG consists of K connected components $\{G_i = (V_i, E_i)\}_{i=1}^{K}$. A node merging algorithm merges $K-1$ pairs of nodes $\mathcal{M} = \{(\boldsymbol{t}_{a_i}, \boldsymbol{t}_{b_i})\}_{i=1}^{K-1}$, forcing $o'(\boldsymbol{t}_{a_i}) = o'(\boldsymbol{t}_{b_i})$. Based on it, we construct the following weighted connected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ built on K components:

$$\begin{aligned}
&\mathcal{V} = \{G_1, G_2, \cdots, G_K\}, \\
&\mathcal{E} = \{(G_i, G_j, w_{i,j}) \mid \exists (\boldsymbol{t}_i, \boldsymbol{t}_j) \in \mathcal{M}, \text{s.t.}, \boldsymbol{t}_i \in V_i, \boldsymbol{t}_j \in V_j, w_{i,j} := \left| {}^{o(\boldsymbol{t}_i) - o(\boldsymbol{t}_j)} / {}_{o'(\boldsymbol{t}_i)} \right| \},
\end{aligned}$$

where (G_i, G_j, w) denotes an edge connecting (G_i, G_j) with a weight w. Notably, each merging pair in $\mathcal{M}$ corresponds to an edge in $\mathcal{E}$ exactly.

A.5.1. Consistency of node merging

Node merging enforces a constraint on function o' to have identical values at certain bias factors. A concern arises that this additional constraint might conflict with the conditions for identifiability (*i.e.*, $c \cdot r = c' \cdot r'$). Fortunately, such conflicts are absent when the graph $\mathcal{G}$ is acyclic, ensuring the node merging approach is **consistent**, as detailed in the following proposition.

Proposition 3 (Consistency of node merging)**.** *Given an acyclic graph $\mathcal{G}$ with merging pairs $\mathcal{M} = (\boldsymbol{t}_{a_i}, \boldsymbol{t}_{b_i})_{i=1}^{K-1}$, the true relevance $r(\cdot)$, and the true observation $o(\cdot)$, we can find functions r' and o' such that they satisfy:*

Condition 1: Unbiased click probability estimation: $r(\boldsymbol{x}) \cdot o(\boldsymbol{t}) = r'(\boldsymbol{x}) \cdot o'(\boldsymbol{t}), \quad \forall (\boldsymbol{x}, \boldsymbol{t}) \in \mathcal{D}$;

Condition 2: Compliance with node merging constraints: $o'(\boldsymbol{t}_{a_i}) = o'(\boldsymbol{t}_{b_i}), \quad \forall (\boldsymbol{t}_{a_i}, \boldsymbol{t}_{b_i}) \in \mathcal{M}$.

Proof. The proof employs induction.

Base step: At $K = 1$, the IG is connected, free from node merging constraints. Setting $r'(\cdot) = Cr(\cdot)$ and $o'(\cdot) = {}^{o(\cdot)}/_{C}$ for any positive constant C satisfies unbiased click probability estimation.

Inductive step: Assuming conditions 1 and 2 hold for $K = n$ ($n \geq 1$), we examine $K = n + 1$. Given $\mathcal{G}$ as an acyclic graph, we can find a $G_i = (V_i, E_i) \in \mathcal{V}$ with a single edge connecting it. Let the corresponding merging pair be $(\boldsymbol{t}_a, \boldsymbol{t}_b)$, where $\boldsymbol{t}_a \in V_i$, and $\boldsymbol{t}_b$ belongs to another node in $G \setminus G_i$. The two conditions are met in $G \setminus G_i$ (with n nodes) by induction assumption. We need only find r' and o' for G_i, as follows:

$$o'(\boldsymbol{t}) = o(\boldsymbol{t})\frac{o'(\boldsymbol{t}_b)}{o(\boldsymbol{t}_a)}, \quad r'(\boldsymbol{x}) = r(\boldsymbol{x})\frac{o(\boldsymbol{t}_a)}{o'(\boldsymbol{t}_b)}, \quad \forall(\boldsymbol{x}, \boldsymbol{t}) \in \mathcal{D} \wedge \boldsymbol{t} \in V_i, \tag{A.7}$$

with $o'(\boldsymbol{t}_b)$'s value fixed in $G \setminus G_i$, meeting both conditions by induction. It is easy to verify that Eq.(A.7) meets both conditions as well.

□

Note that in cases where $\mathcal{G}$ is not a tree and contains cycles, the inductive step fails at the step that finding a node with a single edge connecting it, resulting in inconsistency.

A.5.2. Error bound for node merging

Based on the above notation, we derive the following error bound for node merging.

Proposition 4 (Error bound of node merging)**.**

$$r(\boldsymbol{x}) \cdot o(\boldsymbol{t}) = r'(\boldsymbol{x}) \cdot o'(\boldsymbol{t}) \quad \Longrightarrow \quad \left|\frac{r'(\boldsymbol{x}_1)}{r(\boldsymbol{x}_1)} - \frac{r'(\boldsymbol{x}_2)}{r(\boldsymbol{x}_2)}\right| \leq D(\mathcal{G}), \quad \forall \boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{X},$$

where $D(\mathcal{G})$ is the diameter of $\mathcal{G}$, i.e., the maximal distance between nodes in $\mathcal{G}$.

Proof. In our proof, we employ Proposition 2 and the triangle inequality on the paths within $\mathcal{G}$. Initially, define $\boldsymbol{x}(G_i) = \{\boldsymbol{x} \mid \exists \boldsymbol{t} \in V_i, \text{s.t. } (\boldsymbol{x}, \boldsymbol{t}) \in \mathcal{D}\}$ to represent all features associated with the bias factors in component G_i. It is crucial to note that a feature cannot belong to multiple components, as this would imply connectivity between these components through the said feature.

For any two components G_s and G_t, we find a path in $\mathcal{G}$ connecting them, *i.e.*, $G_{a_1} \to G_{a_2} \to \cdots \to G_{a_n}$ with $a_1 = s$ and $a_n = t$. We have:

$$\begin{aligned}
&\forall \boldsymbol{x}_1 \in \boldsymbol{x}(G_{a_1}), \boldsymbol{x}_2 \in \boldsymbol{x}(G_{a_2}), \cdots, \boldsymbol{x}_n \in \boldsymbol{x}(G_{a_n}), \\
&\qquad \left|\frac{r'(\boldsymbol{x}_1)}{r(\boldsymbol{x}_1)} - \frac{r'(\boldsymbol{x}_n)}{r(\boldsymbol{x}_n)}\right| = \left|\left(\frac{r'(\boldsymbol{x}_1)}{r(\boldsymbol{x}_1)} - \frac{r'(\boldsymbol{x}_2)}{r(\boldsymbol{x}_2)}\right) + \left(\frac{r'(\boldsymbol{x}_2)}{r(\boldsymbol{x}_2)} - \frac{r'(\boldsymbol{x}_3)}{r(\boldsymbol{x}_3)}\right) + \cdots + \left(\frac{r'(\boldsymbol{x}_{n-1})}{r(\boldsymbol{x}_{n-1})} - \frac{r'(\boldsymbol{x}_n)}{r(\boldsymbol{x}_n)}\right)\right| \\
&\qquad\qquad \leq \left|\frac{r'(\boldsymbol{x}_1)}{r(\boldsymbol{x}_1)} - \frac{r'(\boldsymbol{x}_2)}{r(\boldsymbol{x}_2)}\right| + \left|\frac{r'(\boldsymbol{x}_2)}{r(\boldsymbol{x}_2)} - \frac{r'(\boldsymbol{x}_3)}{r(\boldsymbol{x}_3)}\right| + \cdots + \left|\frac{r'(\boldsymbol{x}_{n-1})}{r(\boldsymbol{x}_{n-1})} - \frac{r'(\boldsymbol{x}_n)}{r(\boldsymbol{x}_n)}\right| \\
&\qquad\qquad \leq w_{a_1,a_2} + w_{a_2,a_3} + \cdots + w_{a_{n-1},a_n}.
\end{aligned}$$

Here, the first inequality is derived using the triangle inequality, while the second follows from the definition of $\mathcal{G}$ and Proposition 2. Consequently, the error between $\boldsymbol{x}_1$ and $\boldsymbol{x}_n$ $\left|{}^{r'(\boldsymbol{x}_1)}/_{r(\boldsymbol{x}_1)} - {}^{r'(\boldsymbol{x}_n)}/_{r(\boldsymbol{x}_n)}\right|$ is limited by the cumulative weights of the path connecting G_s and G_t. Given the arbitrary nature of this selection, the error between any two features is constrained by the diameter of $\mathcal{G}$. □

B. Algorithms

B.1. Identifiability check

Based on Theorem 1, we illustrate the identifiability check in Algorithm 1. In lines 1-8, we construct a mapping, from feature to the bias factor sets that ever appear together with it. In lines 9-15, we connect the bias factor set for each feature to a complete graph, since they are all related to the same feature and thus connect.

Algorithm 1: Identifiability check

Input: Dataset $\mathcal{D} = \{(\boldsymbol{x}_i, \boldsymbol{t}_i)\}_{i=1}^{|\mathcal{D}|}$
Output: Whether a relevance model trained on $\mathcal{D}$ is identifiable

$S \leftarrow$ Dictionary() ; ▷ Initialize S with an empty dictionary
▷ Construct a mapping: feature $\rightarrow$ bias factors list
for $i = 1$ **to** $|\mathcal{D}|$ **do**
  if $\boldsymbol{x}_i \notin S$ **then**
    $S[\boldsymbol{x}_i] \leftarrow \{\boldsymbol{t}_i\}$;
  else
    $S[\boldsymbol{x}_i] \leftarrow S[\boldsymbol{x}_i] \cup \{\boldsymbol{t}_i\}$;
  end
end
$V \leftarrow \{v_1, v_2, \cdots, v_{|\mathcal{T}|}\}$;
$E \leftarrow \varnothing$;
▷ Construct identifiability graph (IG) based on S
for $\boldsymbol{x} \in S$ **do**
  for $\boldsymbol{t}_1, \boldsymbol{t}_2 \in S[\boldsymbol{x}] \times S[\boldsymbol{x}]$ **do**
    $E \leftarrow E \cup \{(v_{\boldsymbol{t}_1}, v_{\boldsymbol{t}_2})\}$;
  end
end
if $G = (V, E)$ *is connected* **then**
  return *true*
else
  return *false*
end

B.2. Full algorithm for node intervention

We illustrate the full algorithm for node intervention (§ 4.1) in Algorithm 2. Here we use Prim's algorithm to find the MST. In lines 1-2, we construct the IG and find K connected components. In line 3, we initialize the intervention set. In line 4, we initialize the found node set to the first connected components for running Prim's algorithm. In line 7, we traverse the components in the unfound set $U - U'$ (denoted by G_i) and in the found set U' (denoted by G_j), and compute the intervention cost and the best intervention pair between G_i and G_j in line 8. If the cost is the best, we record the cost, intervention pair, and the target component in lines 10-12. Finally, we add the best intervention pair we found in line 15 and update the found set in line 16 for Prim's algorithm.

B.3. Full algorithm for node merging

We illustrate the full algorithm for node merging (§ 4.2) in Algorithm 3.

Similar to node intervention, here we also use Prim's algorithm to find the MST. In lines 1-2, we construct the IG and find K connected components. In line 3, we initialize the merging set. In line 4, we initialize the found node set to the first connected components for running Prim's algorithm. In line 7, we traverse the components in the unfound set $U - U'$ (denoted by G_i) and in the found set U' (denoted by G_j) and compute the merging cost and the best intervention pair between G_i and G_j in line 8. If the cost is the best, we record the cost, merging the pair and the target component in lines 10-12. Finally, we add the best merging pair we found in line 15 and update the found set in line 16 for Prim's algorithm.

Algorithm 2: Node intervention

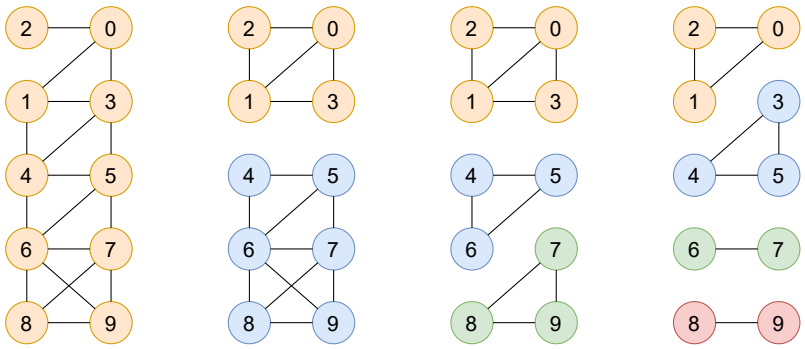
Input: Dataset $\mathcal{D} = \{(\boldsymbol{x}_i, \boldsymbol{t}_i)\}_{i=1}^{|\mathcal{D}|}$
Output: Intervention set E'

Construct the IG $G = (V, E)$ on $\mathcal{D}$ using Algorithm 1;
$U \leftarrow \{G_1 = (V_1, E_1), \cdots, G_K = (V_K, E_K)\}$ denoting K connected components for G;
$E' \leftarrow \{\}$; ▷ Initialize intervention set
$U' \leftarrow \{G_1\}$; ▷ Initialize found nodes for Prim's algorithm
▷ Construct an MST using Prim's algorithm
while $|U'| \neq K$ **do**
  $c_{\min} = +\infty$;
  for $G_i \in U - U', G_j \in U'$ **do**
    Compute $\boldsymbol{x}^{(i,j)}, \boldsymbol{t}^{(i,j)}$ and the intervention cost c based on Eq.(3) - Eq.(6);
    if $c < c_{\min}$ **then**
      $c_{\min} \leftarrow c$;
      $\boldsymbol{x}^*, \boldsymbol{t}^* \leftarrow \boldsymbol{x}^{(i,j)}, \boldsymbol{t}^{(i,j)}$;
      $G^* \leftarrow G_j$;
    end
  end
  $E' \leftarrow E' \cup \{(\boldsymbol{x}^*, \boldsymbol{t}^*)\}$; ▷ Add the best intervention pair
  $U' \leftarrow U' \cup \{G^*\}$; ▷ Update found nodes using Prim's algorithm
end
return E'

Algorithm 3: Node merging

Input: Dataset $\mathcal{D} = \{(\boldsymbol{x}_i, \boldsymbol{t}_i)\}_{i=1}^{|\mathcal{D}|}$
Output: Merging set E'

Construct the IG $G = (V, E)$ on $\mathcal{D}$ using Algorithm 1;
$U \leftarrow \{G_1 = (V_1, E_1), \cdots, G_K = (V_K, E_K)\}$ denoting K connected components for G;
$E' \leftarrow \{\}$; ▷ Initialize merging set
$U' \leftarrow \{G_1\}$; ▷ Initialize found nodes for Prim's algorithm
▷ Construct an MST using Prim's algorithm
while $|U'| \neq K$ **do**
  $c_{\min} = +\infty$;
  for $G_i \in U - U', G_j \in U'$ **do**
    Compute $\boldsymbol{t}_i^*, \boldsymbol{t}_j^*$ and the merging cost c on Eq.(7) - Eq.(8);
    if $c < c_{\min}$ **then**
      $c_{\min} \leftarrow c$;
      $\boldsymbol{t}_A^*, \boldsymbol{t}_B^* \leftarrow \boldsymbol{t}_i^*, \boldsymbol{t}_j^*$;
      $G^* \leftarrow G_j$;
    end
  end
  $E' \leftarrow E' \cup \{(\boldsymbol{t}_A^*, \boldsymbol{t}_B^*)\}$; ▷ Add the best merging pair
  $U' \leftarrow U' \cup \{G^*\}$; ▷ Update found nodes using Prim's algorithm
end
return E'



(a) $K = 1$ (b) $K = 2$ (c) $K = 3$ (d) $K = 4$

Figure 5. IGs of the simulated datasets. Numbers in the nodes denote the position index (starting from 0 to 9).

Table 3. Dataset statistics

	Yahoo!	Istella-S
# Queries	28,719	32,968
# Documents	700,153	3,406,167
# Features	700	220
# Relevance levels	5	5

Table 4. Identifiability graph statistics

	Yahoo!	Istella-S
# Nodes	48,894	48,919
# Edges	59,811,898	5,972
# Connected components (CC)	28,958	47,976
# Nodes in the Top 1 CC	15,684	101
# Nodes in the Top 2 CC	98	7
# Nodes in the Top 3 CC	96	5

C. Experiment details

C.1. Datasets

In this section, we show details about the datasets we used in this work, including the simulated datasets and semi-synthetic datasets. The datasets can be downloaded from `https://webscope.sandbox.yahoo.com/` (Yahoo!), `http://quickrank.isti.cnr.it/istella-dataset/` (Istella-S) and `http://www.thuir.cn/tiangong-st/` (TianGong-ST).

Further details of the simulated datasets For a fair comparison, all simulated datasets comprised 10,000 one-hot encoded documents and 1,150 queries with randomly sampled 5-level relevance, and each query contains 10 documents. Figure 5 demonstrates the IGs we used for simulating datasets, where the number of connected components is $K = 1, 2, 3, 4$ respectively.

Further details of the semi-synthetic datasets We followed the given data split of training, validation, and testing. To generate initial ranking lists for click simulation, we followed the standard process (Joachims et al., 2017; Ai et al., 2018a; Chen et al., 2021; 2022a) to train a Ranking SVM model (Joachims, 2006) with 1% of the training data with relevance labels, and sort the documents. We used the ULTRA framework (Ai et al., 2018b; 2021) to pre-process datasets. Table 3 shows the characteristics of the two datasets we used.

Since the IGs of the two datasets we used are too large to visualize, we show several graph characteristics about them in Table 4 where the number of context types is 5,000.

C.2. Click simulation

Position-based model We sampled clicks according to the examination hypothesis (Eq.(1)) for fully simulated datasets. Following the steps proposed by Chapelle et al. (2009), we set the relevance probability to be:

$$r(\boldsymbol{x}) = \epsilon + (1 - \epsilon)\frac{2^{y_{\boldsymbol{x}}} - 1}{2^{y_{\max}} - 1}, \tag{C.1}$$

where $y_{\boldsymbol{x}}$ is the relevance level of $\boldsymbol{x}$, and $y_{\max} = 5$ in our case. ϵ is the click noise level and we set $\epsilon = 0.1$ as the default setting. For the observation part, following Ai et al. (2021) we adopted the position-based examination probability $o(p)$ for each position p by eye-tracking studies (Joachims et al., 2005).

Contextual position-based model For simulating contextual bias on the semi-synthetic dataset, following Fang et al. (2019), we assigned each context id t with a vector $\boldsymbol{X}_t \in \mathbb{R}^{10}$ where each element is drawn from $\mathcal{N}(0, 0.35)$. We followed the same formula as a position-based model for click simulation, while the observation probability takes the following formula:

$$o(t, p) = o(p)^{\max\{\boldsymbol{w}^\top \boldsymbol{X}_t + 1, 0\}},$$

where $o(p)$ is the position-based examination probability used in the fully synthetic experiment. $\boldsymbol{w}$ is fixed to a 10-dimensional vector uniformly drawn from $[-1, 1]$.

C.3. Training details

Implementation details of baselines ***DLA*** (Ai et al., 2018b; Vardasbi et al., 2020; Chen et al., 2021; 2023) uses the following formula to learn the relevance model r' and observation model o' dually:

$$r'_k(\boldsymbol{x}) \leftarrow \arg\min_{r'(\boldsymbol{x})} \sum_{i=1}^{|\mathcal{D}|} \mathbb{1}_{\boldsymbol{x}_i = \boldsymbol{x}} \mathcal{L}(c_i, o'_{k-1}(\boldsymbol{t}_i), r'(\boldsymbol{x})),$$
$$o'_k(\boldsymbol{t}) \leftarrow \arg\min_{o'(\boldsymbol{t})} \sum_{i=1}^{|\mathcal{D}|} \mathbb{1}_{\boldsymbol{t}_i = \boldsymbol{t}} \mathcal{L}(c_i, o'(\boldsymbol{t}), r'_{k-1}(\boldsymbol{x}_i)),$$

where $\mathcal{D} = \{(\boldsymbol{x}_i, \boldsymbol{t}_i, c_i)\}_{i=1}^{|\mathcal{D}|}$ is the dataset, $r'_k(\boldsymbol{x})$ and $o'_k(\boldsymbol{t})$ are the k-th step model output ($1 \le k \le T$). The initial values for o'_0 and r'_0 were randomly initialized from a uniform distribution within the range of $[0, 1]$. After each step, we applied a clipping operation to constrain the outputs within the interval $[0, 1]$. $\mathcal{L}$ is the training objective function, and we implement it with MSE loss:

$$\mathcal{L}(c_i, o'_i, r'_i) = (c_i - o'_i r'_i)^2.$$

Regression-EM (Wang et al., 2018; Sarvi et al., 2023) uses an iterative process similar to DLA, while the relevance model r' and observation model o' are learned as follows:

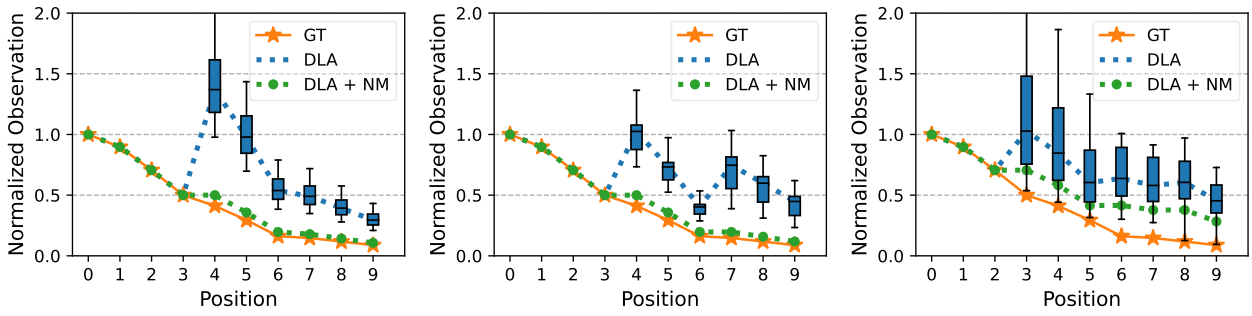
$$r'_k(\boldsymbol{x}) \leftarrow \frac{\sum_{i=1}^{|\mathcal{D}|} \mathbb{1}_{\boldsymbol{x}_i = \boldsymbol{x}} \left\{ c_i + (1 - c_i) \frac{[1 - o'_{k-1}(\boldsymbol{t}_i)] r'_{k-1}(\boldsymbol{x}_i)}{1 - o'_{k-1}(\boldsymbol{t}_i) r'_{k-1}(\boldsymbol{x}_i)} \right\}}{\sum_{i=1}^{|\mathcal{D}|} \mathbb{1}_{\boldsymbol{x}_i = \boldsymbol{x}}},$$
$$o'_k(\boldsymbol{t}) \leftarrow \frac{\sum_{i=1}^{|\mathcal{D}|} \mathbb{1}_{\boldsymbol{t}_i = \boldsymbol{t}} \left\{ c_i + (1 - c_i) \frac{[1 - r'_{k-1}(\boldsymbol{x}_i)] o'_{k-1}(\boldsymbol{t}_i)}{1 - o'_{k-1}(\boldsymbol{t}_i) r'_{k-1}(\boldsymbol{x}_i)} \right\}}{\sum_{i=1}^{|\mathcal{D}|} \mathbb{1}_{\boldsymbol{t}_i = \boldsymbol{t}}}.$$

Two-Tower (Guo et al., 2019; Chen et al., 2020; 2022a) treats r' and o' as two towers and facilitate the multiplication of the outputs of them close to clicks. They use the binary cross entropy loss to train the models by gradient descent, formulated as:

$$\mathcal{L} = \sum_{i=1}^{|\mathcal{D}|} -c_i \log[r'(\boldsymbol{x}_i) o'(\boldsymbol{t}_i)] - (1 - c_i) \log[1 - r'(\boldsymbol{x}_i) o'(\boldsymbol{t}_i)],$$

where r' and o' are constrained to $[0, 1]$ by applying a sigmoid function.

Hyper-parameters We ran each experiment 10 times and reported the average values as well as the standard deviations. On the fully synthetic datasets, we implemented the ranking and observation models as embedding models and controlled $T = 20,000$ to ensure the convergence. On the semi-synthetic datasets, we also implemented the ranking and observation



(a) $K = 2$ (b) $K = 3$ (c) $K = 4$

Figure 6. Observation curves on each fully simulated dataset. GT = Ground Truth. NM = Node Merging.

models as embedding models by assigning a unique identifier based on ranking features to each document, which improves the model's ability to fit clicks during training. The number of epochs was $T = 100$. After training, to generalize to the test set, we trained a LightGBM (Ke et al., 2017) as a ranking model with the learned relevance embeddings of each feature. The total number of trees was 500, the learning rate was 0.1, number of leaves for one tree was 255.

Implementation details of node merging For node merging, we used the position number as the bias feature on the fully synthetic dataset. On the semi-synthetic dataset, we formed the bias feature $\boldsymbol{X}_{p,t}$ for each bias (consisting of position p and context type t) as follows: we multiplied p by 10 and added it to the end of the 10-dimensional context vector $\boldsymbol{X}_t$, to form an 11-dimensional bias feature. This method increases the weight of the position, forcing node merging to give priority to merging different context types rather than positions.

Implementation details of node intervention For node intervention on the fully synthetic dataset, we trained the ranking model and observation model using node merging and used their values to implement the cost function (Eq.(3)).

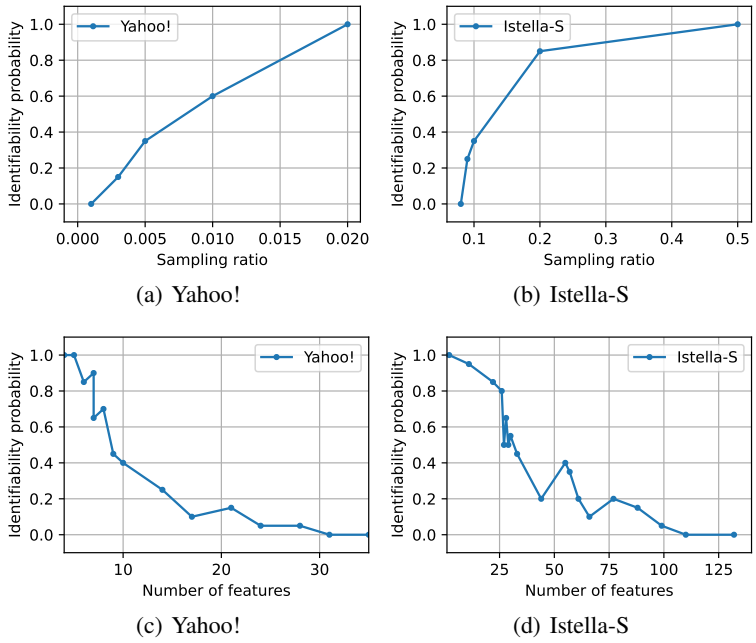
D. Further experiment results

D.1. Observation probability curves

Observation probability estimation, also known as propensity estimation (Agarwal et al., 2019c; Fang et al., 2019), is a critical task in ULTR. Figure 6 illustrates ground truth (GT) for observation probability for $K = 2$, $K = 3$, and $K = 4$ datasets, alongside estimations by DLA and node merging (NM). We normalize the model's predictions by dividing the predicted probabilities at each position by the predicted probability of the first position. One can observe substantial volatility in DLA's predictions under unidentifiability, as reflected in the boxplots. Node merging, by mandating shared observation probabilities for disconnected positions on the IG, achieves model identifiability and closer alignment with the GT observation curve. Notably, node merging's accuracy for the $K = 4$ dataset is inferior to that of $K = 3$ and $K = 2$, due to the merging of positions with larger observational disparities in $K = 4$. This is attributed to the merging of positions with greater differences in observations in $K = 4$ (positions 2 versus 3) since they are disconnected (visualized in Figure 5), as opposed to smaller differences in $K = 3$ and $K = 2$. Consequently, this results in a relatively poorer performance of node merging in recovering relevance for $K = 4$ (as seen in Figure 3(b)). This phenomenon further corroborates Proposition 2 and illustrates the dependence of node merging's performance on dataset characteristics.

D.2. Further investigation on the identifiability of TianGong-ST

On the TianGong-ST dataset, vertical types are represented in the format "v_1#v_2", *e.g.*, "-1#-1" or "30000701#131". We investigated the consequences of excluding specific bias factors, for example, disregarding either v_1 or v_2. A summary of the results is presented in Table 5. Our findings reveal that when one of v_1 and v_2 is kept, the IG loses its connectivity, demonstrating the prevalence of unidentifiability issues in real-world scenarios. However, when only the position is retained, the IG regains connectivity, rendering it identifiable. We observed the reason is that some queries are repeated in the dataset, with variations in the order of related documents. This observation suggests that the search engine may have done some position intervention during deployment which enhances the IG's connectivity. Overall, it reveals again that introducing



(a) Yahoo! (b) Istella-S

(c) Yahoo! (d) Istella-S

Figure 7. The influence of (a)(b) the sampling ratio and (c)(d) number of features of datasets on the identifiability probabilities, on Yahoo! and Istella-S, respectively.

excessive bias factors leads to a higher probability of unidentifiability.

Table 5. Identifiability of TianGong-ST in different bias factor settings.

Bias factor	# Connected components	Identifiable?
(v_1, v_2, position)	2,900	×
(v_1, position)	1,106	×
(v_2, position)	87	×
position only	1	✓

D.3. Impact of the sampling ratio and feature count on the identifiability probabilities

We sampled the datasets randomly according to a sampling ratio 20 times and calculated the frequency that the IG calculated on the sampled datasets is connected, when *positions are the only bias factors*. From Figure 7(a) and 7(b), one can find that although the IGs on both the entire datasets are connected, the subsets of the datasets are not. Specifically, the connectivity of IGs can be guaranteed only when the size of the subset comprises more than approximately 2% (in the case of Yahoo!) and 50% (in the case of Istella-S) of the respective dataset.

Subsequently, we explored the impact of feature count on identifiability. Due to dataset constraints, increasing the number of features was not feasible. Therefore, we adopted a strategy of progressively removing features from an unidentifiable dataset until the IG became connected. As both datasets are identifiable in the absence of context, we generated 500 contexts for Yahoo! and 50 for Istella-S, as depicted in Figures 4(c) and 4(d). The results are displayed in Figure 7(c) and 7(d). One can observe that the probability of identifiability diminishes with an increase in feature count. Notably, in the presence of a sufficient number of bias factors, a relatively small number of features (5-10) is sufficient to induce unidentifiability.

D.4. Impact of initialization strategies

In unidentifiable scenarios, models can converge to various parameters that predict click probability correctly, but only a limited range truly represents correct relevance. Therefore, the choice of model initialization plays a crucial role in determining convergence quality. We examined two strategies for initializing the observation model: setting all bias factors'

Table 6. Performance comparison on two datasets under CPBM bias with different initialization strategies on the observation model. We ran each experiment 10 times and reported the average results as well as the standard deviations.

Dataset	Method	Training				Test	
		MCC ↑	nDCG@5 ↑	nDCG@10 ↑	Click MSE	nDCG@5 ↑	nDCG@10 ↑
Yahoo	No debias	$0.765_{\pm.000}$	$0.841_{\pm.000}$	$0.915_{\pm.000}$	3.7×10^{-4}	$0.693_{\pm.002}$	$0.741_{\pm.001}$
	DLA$^{\mathcal{I}}$	$0.750_{\pm.000}$	$0.844_{\pm.000}$	$0.914_{\pm.000}$	1.7×10^{-5}	$0.693_{\pm.001}$	$0.741_{\pm.001}$
	DLA$^{\mathcal{R}}$	$0.619_{\pm.003}$	$0.819_{\pm.001}$	$0.892_{\pm.001}$	1.7×10^{-5}	$0.688_{\pm.001}$	$0.737_{\pm.001}$
	DLA$^{\mathcal{I}}$ + Node merging	$\mathbf{0.771_{\pm.000}}$	$0.853_{\pm.000}$	$0.920_{\pm.000}$	4.4×10^{-5}	$0.697_{\pm.001}$	$0.745_{\pm.001}$
	DLA$^{\mathcal{R}}$ + Node merging	$0.744_{\pm.002}$	$\mathbf{0.863_{\pm.001}}$	$\mathbf{0.921_{\pm.001}}$	4.3×10^{-5}	$\mathbf{0.699_{\pm.001}}$	$\mathbf{0.746_{\pm.001}}$
Istella-S	No debias	$0.764_{\pm.000}$	$0.885_{\pm.000}$	$0.941_{\pm.000}$	4.4×10^{-5}	$0.634_{\pm.001}$	$0.682_{\pm.001}$
	DLA$^{\mathcal{I}}$	$0.764_{\pm.000}$	$0.886_{\pm.000}$	$0.941_{\pm.000}$	1.1×10^{-6}	$0.633_{\pm.001}$	$0.682_{\pm.001}$
	DLA$^{\mathcal{R}}$	$0.748_{\pm.001}$	$0.874_{\pm.001}$	$0.931_{\pm.000}$	1.1×10^{-6}	$\mathbf{0.638_{\pm.002}}$	$\mathbf{0.686_{\pm.002}}$
	DLA$^{\mathcal{I}}$ + Node merging	$0.772_{\pm.000}$	$0.892_{\pm.000}$	$0.944_{\pm.000}$	2.2×10^{-5}	$0.636_{\pm.001}$	$0.684_{\pm.001}$
	DLA$^{\mathcal{R}}$ + Node merging	$\mathbf{0.782_{\pm.001}}$	$\mathbf{0.900_{\pm.001}}$	$\mathbf{0.947_{\pm.000}}$	2.2×10^{-5}	$\mathbf{0.638_{\pm.001}}$	$\mathbf{0.686_{\pm.001}}$

observation probabilities to 1.0 (denoted by $\mathcal{I}$) and employing random initialization within [0.0, 1.0] (denoted by $\mathcal{R}$). Table 6 presents the performance of DLA and node merging using different initialization strategies, which show that DLA's efficacy on unidentifiable datasets is highly sensitive to initial observation probabilities, with randomization often leading to subpar convergence. In contrast, for identifiable datasets processed with node merging, DLA exhibits more consistent results, demonstrating resilience to varying initialization approaches and supporting our theory of identifiability.