Learning item embeddings using biased feedback

Ashudeep Singh
Department of Computer Science
Cornell University
Ithaca, NY-14853
ashudeep@cs.cornell.edu

Thorsten Joachims
Department of Computer Science
Cornell University
Ithaca, NY-14853
tj@cs.cornell.edu

Abstract

Learning item embeddings from browsing logs of recommender systems provides intriguing opportunities for understanding user preferences. However, such log data can be severely biased because recommendations imply a selection bias on the number of clicks an item receives. This selection bias can lead to learned embeddings that are distorted by past recommendations and that do not reflect the true semantic similarity one would like to capture. To overcome this problem, we formulate the task of learning embeddings as a counterfactual learning problem: how would the user have clicked, if the recommendation algorithm had not interfered? To demonstrate effectiveness and promise of this approach, we present synthetic experiments that illustrate how the counterfactual learning approach can recover the true embeddings despite biased data.

1 Introduction

When learning an embedding of items, the learned distances in the embedding space should reflect some notion of semantic similarity. In recommender systems, a promising source of data for learning such semantic embeddings comes from the access logs that users provide, since it is has been widely observed that the items accessed in the same session are semantically more similar than random items. Unfortunately, however, this does not necessarily mean that this data reflects semantic similarity in an unbiased way. For example, consider co-click within a session as a signal of similarity between two items. In order for two items to be clicked within one session, two things must happen. The user must be interested in both items, and the user must have observed these two items. The former conveys the semantic similarity we may want to capture, but the latter introduces a selection bias that distorts the signal as it largely depends on how the items were presented to the users (e.g., how difficult it is to navigate from one item to the other). In this way, two moderately similar items placed next to each other can receive more co-clicks than two highly similar items placed far apart in terms of "browsing distance".

In this paper, we propose a counterfactual learning approach that provides a principled way of separating signal and biases for learning semantically meaningful embeddings. In particular, we propose a method for learning item embeddings from co-access data that users provide in an online recommendation system. Viewing recommendation from a causal inference perspective, we argue that exposing a user to a set of items in a recommendation system is an intervention analogous to exposing a patient to a treatment in a medical study (Schnabel et al., 2016). In both cases, the goal is to estimate the true effect of the intervention despite a possibly biased assignment mechanism.

Formally, our goal is to learn an embedding where the conditional probability of observing a click on an item conditioned on the click on another item decreases with the semantic distance between the items (Globerson et al., 2007). Considering co-click as a signal of similarity between two items, we identify two components that together introduce a selection bias – the websites biases in how items are presented, and biases that result from how humans react to these presentation choices. The

What If Workshop at NIPS 2017, Long Beach, CA, USA.

former we call "system bias" and the latter we call "behavioral bias". An example of the former may be recommending "you may also like y" to a user who has shown interest in item x, and the latter may come from the position where such a recommendation is inserted. To deal with these biases and extract the semantic information, we phrase the embedding problem as the counterfactual question of how the user would have clicked, if items had been presented uniformly at random. To this effect, we propose an embedding objective based on a counterfactual estimator of the unbiased training loss. We evaluate and contrast our approach to naive embedding methods on synthetic click data where we can vary the amount of bias. We find that naive embeddings can be severely distorted by the presence of bias, but that the counterfactual embedding approach can nevertheless recover the true embedding.

2 Setup

Consider a user u_i who visits a website and is shown documents $(d_1, d_2, d_3 \dots d_n)$ — for example, a page showing the most recent 25 papers on `arXiv.org`. We observe clicks on a set of documents in the following sequence $\langle c_{d_1}, c_{d_2}, \dots c_{d_n} \rangle$. Assuming the user went through the list of documents in order from top to bottom, to utilize the co-clicks of these documents, we convert the sequence of clicks into the pairwise co-clicks $\{(d_i, d_j, c_{ij}) : c(d_i) = 1\}$, where $c(d_i) = 1$ means d_i is clicked and $c_{ij} = 1$ if d_j is clicked given d_i was clicked, else $c_{ij} = 0$. For example, in figure 1a, the listing of 4 books on a bestseller list (let's say *b, r, y, g*) list may get clicks on *b*, *y* and *g*. This can be converted into $c_{br} = 0, c_{by} = 1, c_{bg} = 1, c_{yr} = 0, c_{yg} = 1, c_{gr} = 0$ to learn from pairwise relevances.

Our idea is to use this co-click dataset to learn a latent space such that the pairs of documents with $c_{ij} = 1$ are placed closer to each other than the pair of documents that are not co-clicked in a session. Specifically, we want to discover a d-dimensional embedding where the conditional probability of clicking d_j given d_i is clicked in the session is proportional to a monotonically decreasing function of the distance between these documents in the embedding i.e. $p(c_j = 1|c_i = 1, d_i, d_j \in w) \propto f(||X_{d_i} - X_{d_j}||)$ where w is the webpage shown to the user and $X_{d_i}, X_{d_j} \in \mathbb{R}^d$ are the vector representations of documents d_i and d_j in the true underlying embedding respectively.

In our case, we employ a Student t-distribution with a single degree of freedom (similar to Maaten and Hinton (2008)) to define the conditional click probability distribution as a function of distance. This distribution, because of its heavy tailed nature, helps retain the local and the global neighborhood structure in dimensionality reduction and visualization tasks, for example two-dimensional embeddings for activations in neural networks (Maaten and Hinton, 2008). Now we have, $f(d) = \frac{1}{1+d^2}$ such that:

$$p(c_{ij} = 1|c_i = 1, d_i, d_j \in w) = \frac{1}{1 + ||X_i - X_j||^2}$$

For a given embedding, we denote the conditional probability $P(c_{ij} = 1|c_i = 1, d_i, d_j \in w)$ as p_{ij}, and X_{d_i} as X_i in the rest of the work.

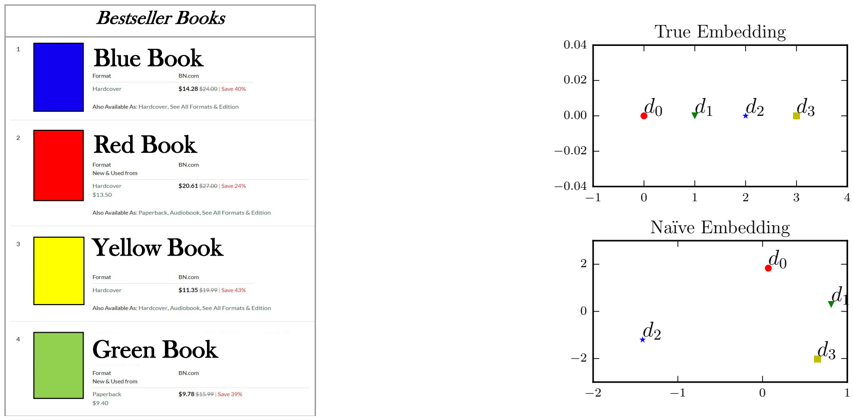
First, let us look at how we may learn given a co-click dataset constructed as above. A naïve approach is to recover an embedding that best explains the conditional probabilities of these co-clicks. Hence we look at the likelihood of the dataset given an embedding:

$$\text{Likelihood}(\mathcal{D}) = \prod_{(d_i, d_j, c_{ij}) \in \mathcal{D}} p_{ij}^{c_{ij}} (1 - p_{ij})^{1 - c_{ij}}$$

To optimize, we can formulate the negative log-likelihood of the dataset as $\mathcal{L}$ as:

$$\hat{\mathcal{L}}_{\text{naïve}}(\mathcal{D}) = \frac{1}{N} \sum_{(d_i, d_j, c_{ij}) \in \mathcal{D}} -c_{ij} \log p_{ij} - (1 - c_{ij}) \log (1 - p_{ij}) \tag{1}$$

where $p_{ij} = 1/(1 + ||X_i - X_j||^2)$. Given a training data $\mathcal{D}$ of co-clicked pairs, we minimize $\hat{\mathcal{L}}_{\text{naïve}}(\mathcal{D})$ to learn the embedding parameters $\hat{X}$[1]. For the rest of the paper, we will refer to the estimator as $\hat{\mathcal{L}}_{\text{naïve}}$ because of the naïvety of the estimator to ignore biases as we will see in the next section.



(a) (b)

Figure 1: In our toy example, (a) Webpage that a visitor sees with the ranking $\langle d_2, d_0, d_3, d_1\rangle$, (b) True embedding that drives the co-clicks and the embedding learned by the naïve estimator

2.1 The effect of selection bias on embeddings

Consider a hand-crafted example where a website shows a homepage to each user consisting of 4 documents $\{d_0, d_1, d_2, d_3\}$. Assume that the true semantic embedding of the documents driving the clicks places the documents along the x-axis as: $X_{d_0} : (0, 0)$, $X_{d_1} : (1, 0)$, $X_{d_2} : (2, 0)$, $X_{d_3} : (3, 0)$, i.e. d_0 is the closest to d_1, then d_2 and then d_3. However, the documents are shown in the order (d_2, d_0, d_3, d_1) as in Figure 1(a).

Let us assume every user certainly clicks on the top-ranked document d_2. Assuming the user behavior follows a position-based click model (Joachims et al., 2005; Chuklin et al., 2015), the user examines the document at position 2 with probability $p(e_2) = 0.5$, at position 3 with probability $p(e_3) = 0.25$, and $p(e_4) = 0.125$. Examination means the user observes the link shown on the page. After clicking the top-ranked d_2, if the user chooses to examine the next document d_0, he or she clicks it with the true embedding probability of the pair X_{d_0} and X_{d_2} i.e. $1/(1 + 2^2) = 1/5$. We see the rest of the clicks in the similar way.

According to the position bias described above, the probability that the pair of documents (d_2, d_0) is examined by a user is 4 times that of the pair (d_2, d_1). In the data we collect, we will observe that d_2 is co-clicked with d_0 60% more often than with d_1 (even though it is more similar to d_1). In other words, if we use the co-click data collected on such a website where the documents are placed in a biased fashion i.e. they are not presented to the users uniformly at random, we learn embeddings that are heavily biased by the presentation order of the documents unless we take into consideration the user click model. For example, in our case, the perceived similarity between documents d_2 and d_0 is higher than between d_2 and d_1, just because very few users examine both d_2 and d_1.

Since we do not observe the examination information, we have a missing data problem at hand. If we nevertheless naively minimize $\hat{\mathcal{L}}_{\text{naïve}}$ on a such a biased dataset, we observe that we recover embeddings that are not equivalent to the known true semantic embedding. Figure 1(b) shows the true embedding and the (naively) learned embedding, which are substantially different. In the next section, we look at a way to fix the estimator in Equation 1 to recover the correct underlying embedding.

3 Unbiased Embedding Estimation

Looking from a causal perspective, we apply a treatment to the user (i.e. the pair of documents (d_i, d_j)) and observe the effect (i.e. click or not (c_{ij})). We will keep this interpretation in mind to de-bias the likelihood objective using an inverse propensity scoring (IPS) estimator.

[1]For all our experiments, we use mini-batch stochastic gradient descent (SGD) for optimization with a batch size of 32 on the training set, using early-stopping over a held-out validation set as the stopping criterion.

Let the examination variable e_{ij} denote whether d_j is examined given d_i was clicked in the same session. If we could observe e_{ij} and knew the observation propensities $P(e_{ij} = 1)$, we could use the following IPS estimator to get an unbiased estimate of the negative log-likelihood despite the bias in examination.

$$\hat{\mathcal{L}}(\mathcal{D}) = \frac{1}{N} \sum_{(d_i,d_j,c_{ij})\in\mathcal{D}} \frac{[e_{ij}=1]}{P(e_{ij}=1)} \Big[- c_{ij} \log \hat{p}_{ij} - (1 - c_{ij}) \log(1-\hat{p}_{ij}) \Big]$$

(Sum over examined pairs only)

Unfortunately, we typically do not observe e_{ij}, unless we can track the user's eye movements. To overcome this lack of access to the examination variables e_{ij}, we derive an alternative unbiased estimator there merely exploits knowledge of the propensities $P(e_{ij} = 1)$ and that clicking an item implies that $e_{ij} = 1$:

$$\begin{aligned}
\hat{\mathcal{L}}(\mathcal{D}) &= -\frac{1}{N} \sum_{(d_i,d_j,c_{ij})\in\mathcal{D}} \frac{[e_{ij}=1][c_{ij}=1]}{P(e_{ij}=1)} \Big(\log \hat{p}_{ij} - \log(1-\hat{p}_{ij}) \Big) - \frac{1}{N} \sum_{(d_i,d_j,c_{ij})\in\mathcal{D}} \frac{[e_{ij}=1]}{P(e_{ij}=1)} \log(1-\hat{p}_{ij}) \\
&= -\frac{1}{N} \sum_{(d_i,d_j,c_{ij})\in\mathcal{D}} \frac{[c_{ij}=1]}{P(e_{ij}=1)} \Big(\log \hat{p}_{ij} - \log(1-\hat{p}_{ij}) \Big) - \frac{1}{N} \sum_{(d_i,d_j,c_{ij})\in\mathcal{D}} \frac{[e_{ij}=1]}{P(e_{ij}=1)} \log(1-\hat{p}_{ij})
\end{aligned}$$

(since $[e_{ij} = 1][c_{ij} = 1] = [c_{ij} = 1]$)

$$\mathbb{E}_{\mathbf{e}}[\hat{\mathcal{L}}(\mathcal{D})] = \frac{-1}{N} \sum_{(d_i,d_j,c_{ij})\in\mathcal{D}} \left(\frac{[c_{ij}=1]}{P(e_{ij}=1)} \Big(\log \hat{p}_{ij} - \log(1-\hat{p}_{ij}) \Big) + \log(1-\hat{p}_{ij}) \right)$$

(Taking expectation over examinations. And since $\mathbb{E}_e\left[\frac{[e_{ij}=1]}{P(e_{ij}=1)}\text{k}\right] = \text{k}$)

$$\begin{aligned}
&= \frac{-1}{N} \sum_{(d_i,d_j,c_{ij})\in\mathcal{D}} \frac{c_{ij}}{P(e_{ij}=1)} \log \hat{p}_{ij} + \left(1 - \frac{c_{ij}}{P(e_{ij}=1)}\right) \log(1-\hat{p}_{ij}) \\
&= \hat{\mathcal{L}}_{IPS}(\mathcal{D})
\end{aligned}$$

We call the estimator $\hat{\mathcal{L}}_{IPS}$ since we used the inverse propensity scoring technique to debias our likelihood estimate. This derivation is similar to the unbiased learning to rank estimator in Joachims et al. (2017), where it was also shown how to derive propensities from a click model. This estimator handles the position (or behavorial) bias, but we still need to handle the 'system' bias. We can use the same IPS technique again to define the following estimator:

$$\hat{\mathcal{L}}_{IPS}(\mathcal{D}) = \frac{1}{N} \sum_{(d_i,d_j,c_{ij})\in\mathcal{D}} \frac{1}{P(s_{ij}=1)} \left[-\frac{c_{ij}}{P(e_{ij}=1)} \log \hat{p}_{ij} - \left(1 - \frac{c_{ij}}{P(e_{ij}=1)}\right) \log(1-\hat{p}_{ij}) \right]$$

where $P(s_{ij} = 1)$ is the probability that the system presents the pair (d_i, d_j) to the user on the same page (and are thus known by design). In expectation over the selections of pairs (d_i, d_j) by the system, this estimator is unbiased (proof omitted).

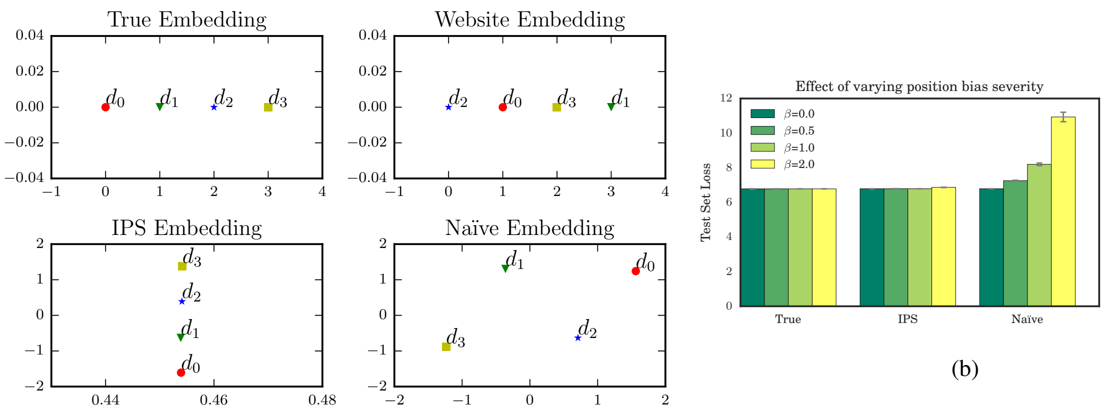
In the rest of the work, we will study the properties of the two estimators– $\hat{\mathcal{L}}_{\text{naïve}}$ and $\hat{\mathcal{L}}_{IPS}$. Specifically, we will look at how the embeddings learnt by minimizing the two estimators compare to the true embeddings.

4 Synthetic Experiments

In this section, we present two sets of synthetic experiments. The first explores the effect of position bias on the learned embeddings under both $\hat{\mathcal{L}}_{\text{naïve}}$ and $\hat{\mathcal{L}}_{\text{IPS}}$. The second set of experiments demonstrates how both system and behavioral biases effect the two estimators, specifically highlighting the robustness of $\hat{\mathcal{L}}_{\text{IPS}}$ to model mismatch in the presence of these biases.

4.1 Experiment 1: Effect of position bias

Let us first experiment with an example similar to the one in Section 2.1. Assume a set of users visiting a website with a set of four documents $d_i, i \in \{1, 2, 3, 4\}$ ranked on it. Lets say there is



(b)

(a)

Figure 2: (a) True embedding of the documents (used to draw co-clicks), Website Embedding X_w (used to draw rankings), Embedding Learned using $\hat{\mathcal{L}}_{\text{IPS}}$, and Embedding learned using $\hat{\mathcal{L}}_{\text{naïve}}$, (b) Test set negative log-likelihood (lower the better) of embeddings with varying severity of the position bias (probability of examining document at rank j, $p(e_j) = \frac{1}{j^\beta}$) for the true embedding (left), learned with IPS estimator (center), learned with the naive estimator (right)

a website (presentation) embedding $X_w = \{d_1 : (2,0), d_2 : (3,0), d_3 : (1,0), d_4 : (0,0)\}$ (e.g., browsing distance in the website) that determines what gets shown together more often and a true document embedding $X_d = \{d_1 : (0,0), d_2 : (1,0), d_3 : (2,0), d_4 : (3,0)\}$ that determines semantic similarity (Figure 2a).

For each user u visiting the website, let the top document d_{u_1} be drawn uniformly at random, the rest of the documents be ranked in the order of their distance to d_{u_1} in the website embedding X_w (ties are broken arbitrarily).

Now consider the following process used by u to decide the clicks in this session being shown $\langle d_{u_1}, d_{u_2}, d_{u_3}, d_{u_4} \rangle$. Let us assume, the user always clicks the first document. The user then moves on to examine the second document with a constant probability $p = 1/2$. If the user examines the next document, the user clicks with a probability $1/(1 + ||X_{d_{u_1}} - X_{d_{u_2}}||^2)$. Then with probability $p = 1/3$ user observes the third document, and if observed, user clicks it with probability $1/(1 + ||X_{d_{u_3}} - X_{d_i}||^2)$, where d_i is the document previously clicked and so on.

Using this setup we collect 10000 tuples of the form (d_i, d_j, c_{ij}). Note that we don't get to observe the random variable of examination for each of the steps, hence we only see whether each of the pairs on the webpage was co-clicked or not (conditioned on the first one of the pair being clicked). Now, we use the dataset $\mathcal{D}$ to minimize the negative log likelihood estimators $\hat{\mathcal{L}}_{\text{naïve}}$ and $\hat{\mathcal{L}}_{\text{IPS}}$ to learn embeddings $\hat{X}$ shown in Figure 2a.

It is apparent that the missing information for the random variable of examination causes the estimator to be biased and hence we do not recover the true underlying pairwise distances. Using the estimator $\hat{\mathcal{L}}_{IPS}$ with the actual propensities of examination logged during the data generation process, we recover an embedding equivalent to the true embedding.

Counterfactual Test Set: We generate a test set to computationally determine the quality of embeddings learned by the different estimators. We draw a random pair of two documents from the set of all pairs, then draw a click or no-click as a Bernoulli trial with a success probability of $1/(1 + d^2)$, where d is the distance between the embedding points of the two documents. Using an embedding's log-likelihood of generating this data, we can estimate the quality of the embedding.

What is the effect of severity of the bias? We also experiment with varying the severity of position bias. By setting the probability of the user examinining the document at position j to $(1/j)^\beta$, we repeat the experiment for values of β in $\{0.0, 0.5, 1.0, 2.0\}$. As you can see in figure 2, $\hat{\mathcal{L}}_{\text{IPS}}$

consistently gives the lower test set negative log likelihood as compared to the biased naïve estimator ($\hat{\mathcal{L}}_{\text{naïve}}$) on the counterfactual test set.

4.2 Experiment 2: Effect of selection bias and position bias

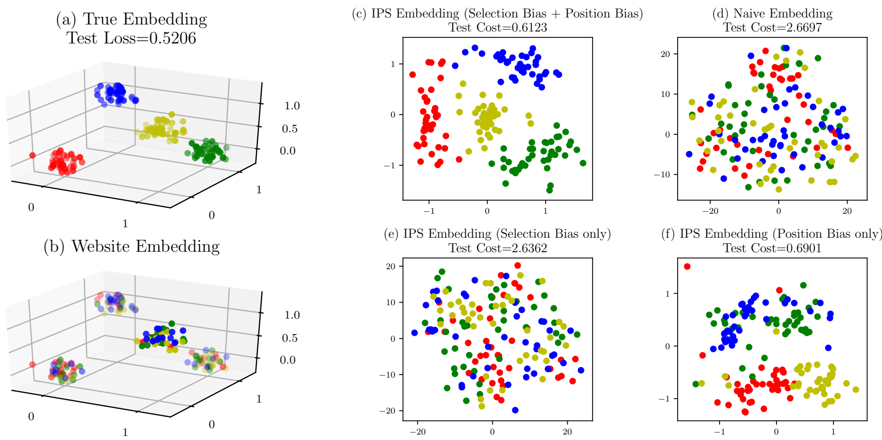


Figure 3: Left: (a) True document embedding, (b) Website embedding (used to draw co-views), Right: (c) Embedding Learned handling both selection and position bias, (d) Embedding learned using naïve estimator $\mathcal{L}_{\text{naïve}}$, (e) Embeddings learned handling only selection bias, (f) Embeddings learned handling only position bias

We also perform another set of experiments with a more realistic synthetic data. Consider a set of 160 documents $\{d_1, d_2, d_3, \dots, d_{4N} | N = 40\}$ with a known true embedding in three-dimensional space. Each quarter of the points belong to a cluster, color coded as red, green, yellow and blue (shown in figure 3 top-left). Each of the clusters is a set of 40 points drawn from a multivariate normal distribution with means at $(0, 0, 0), (1, 1, 0), (0, 1, 1), (1, 0, 1)$ respectively, and a diagonal covariance matrix with all values equal to 0.05. We call this the true embedding X_d. Similar to the true embedding, these 160 points also lie in another three-dimensional embedding, called the *website embedding* X_w. In X_w, about only 20% of the 160 points are in their true position, rest are swapped with another document into a different cluster than the one they belong to in the true embedding (shown in figure 3). The idea of the website embedding is to drive the presentation of documents i.e. induce "system bias" in the data.

Generating clicks We follow a similar idea to the toy experiment in section 2.1 i.e. draw pairs of documents to be presented to the user using the website embedding, introduce position bias, and draw the co-click from the true embeddings. First, for each user, we draw a document d_1 uniformly at random from the set of 160 points. Then, we present the user with 10 documents sampled in proportion to $1/(1 + d)^2$ to their distance from d_1 in the website embedding X_w. Each user clicks the document d_1 and proceeds to examine the rest of the page using position based click model, with $p_i = 1/i$ where i is the position of the document in the ranking. For each position, if the user chooses to examine the document d_j, then the user decides to click it with probability $p(c_j | e_j = 1, c_i = 1) = 1/(1 + ||X_i - X_j||^2)$, X_i and X_j are the true embeddings of d_i and d_j respectively. In this way, we collect a set of pairs of documents that were coclicked or not coclicked.

In this example, we want to demonstrate how the two biases, system bias and behavorial or position bias, affect the embeddings learned from the data. The system bias is due to the biased selection of the pairs shown to the user, while the position bias occurs because the users don't examine all positions equally often. We choose the true embedding to lie in a three-dimensional space and learn a two dimensional embedding to demonstrate that it is meaningful to learn in model mismatch settings too.

Learning Using the data generated by the above process, we minimize the cross entropy loss in equation 1 to learn an embedding using the two estimators $\hat{\mathcal{L}}_{\text{naïve}}$ and $\hat{\mathcal{L}}_{\text{IPS}}$. We also try combinations ignoring one of the two biases at a time. We compare the embeddings obtained by these methods in

Figure 3. One can see that using the IPS estimator $\hat{\mathcal{L}}_{\text{naïve}}$ recovers an embedding similar to the true embedding while others fail to learn the semantic similarities amongst documents.

Counterfactual Test Set To compare the two methods quantitatively, we draw a dataset in the same way as in section 4.1 i.e. randomly drawing a pair of documents and drawing a click from the true embedding. We compare the log-likelihood of generating this test dataset using the embedding learned by the different estimators. The results are compared in Figure 3.

Propensity estimation In case of our synthetic experiments, we are aware of the propensities (*position bias* and *system bias*) because we are in control of the generation process. However, in real world, we would need to use interventions to efficiently calculate the position biases. For example, given the ability to intervene with the rankings, we can use the *swap-k-and-r* method in Joachims et al. (2017) to estimate the relative probability of examining positions k and r. The propensity estimation technique would change with the underlying assumption about the click model (Craswell et al., 2008; Chuklin et al., 2015).

5 Related work

We divide the related work into two groups. First, we mention varios methods in literature for learning embeddings from high-dimensional data or from co-occurrence statistics. Second, we mention methods for learning in the presence of biases for recommendation and information retrieval systems.

Most of the important embedding methods have been developed to reduce the dimensionality of high-dimensional data by embedding it into a lower dimensional space for visualization or other upstream tasks like supervised learning. An embedding algorithm is usually designed to preserve a certain property of the high-dimensional data, for example, pairwise distances in multidimensional scaling (MDS) (Cox and Cox, 2000), neighborhood structure in Locally-Linear embeddings (Roweis and Saul, 2000), and geodesic structure in Isomap (Tenenbaum et al., 2000). Recently, t-SNE embeddings have proven to be successful in visualizing high-dimensional data such as neural network activations, by learning a low dimensional representation through a non-linear transformation while retaining both local and global neighborhood structure (Maaten and Hinton, 2008). It enjoys these benefits due to the heavy tail nature of the t-student distribution. Inspired by these properties, we also use a t-student distribution around an embedding point to define probability of co-occurrence because of its properties.

Another class of embedding methods use co-occurrence data of entities to learn embeddings (Globerson et al., 2007). These methods discover embeddings such that the pairwise distances of items explain the co-occurrence statistics. A special advantage is that these methods can be used to embed entities belonging to different modalities e.g. users and items, authors and papers etc. Some approaches like Logistic Markov Embeddings (LME) embeds songs from music playlists into a euclidean space (Chen et al., 2012) while some embed heterogeneous entities like images and labels in a euclidean space where the true labels of the images are ranked in the neighborhood of the image by relevance (Weston et al., 2011). Recently, word embedding methods such as word2vec (Mikolov et al., 2013) and GLoVe(Pennington et al., 2014) have been successful in learning feature representations for words and their contexts using their co-occurrence information. The learned embeddings are meaningful representations of words that reflect some semantic properties of the language. These embeddings also provide a great baseline feature representation for text classification and clustering (Pennington et al., 2014). Many studies recently have pointed out that these embeddings reflect some cultural biases that are purely a side-effect of using the co-occurrence data blindly (Bolukbasi et al., 2016). Barkan and Koenigstein (2016) uses ideas from word2vec to learn item embeddings from user music history and shopping carts in an online store. Our focus is to reflect how the presentation and human selection introduces a bias in the embeddings regardless of the embedding model used and how we can fix it by reasoning about the bias.

Collaborative filtering exploits the similarity pattern across users to discover similarity between items. However, biases like exposure of the items to users are not traditionally handled. One approach has been to handle by modeling the process using a generative model with exposure as a latent variable and fitting it to the dataset (Liang et al., 2016). Similarly, some works tackle the problem of learning from data with user-item ratings that are missing not at random (MNAR) as a data imputation model based on the joint likelihood of the missing data model and the rating model (Marlin and Zemel, 2009; Hernandez-Lobato et al., 2014). Besides these, in the search result page optimization setting, click models try to model the user behavior in order to explain the bias effects while learning the query-item relevances. For example, a position based model assumes that users have a particular probability of examining each position in a ranked list (Joachims et al., 2005; Craswell et al., 2008), while a cascade model assumes users to only go down examining items in a ranked list sequentially and clicking the relevant documents in the process (Craswell et al., 2008). There are variations of these behavorial assumptions proposed in different click models (Chapelle and Zhang (2009); Borisov et al. (2016), see Chuklin et al. (2015) for details). All these methods are trained to maximize the likelihood of observing the click data while modeling the user behavior according to the click model assumptions. In our work, we try to model item-item similarities, instead of user-item relevances as in the case of click models. We use co-clicks of item-item pairs to determine the similarity of the items, we believe that there is magnitude more amount of data to provide pairwise similarities than user-item relevances since a user or a query might only be seen once in the dataset.

Recent works have pointed out that the evaluation and learning in case of recommendation and evaluation systems is biased if we do not consider the biases due to human behavior(Schnabel et al., 2015; Joachims et al., 2017). Drawing an analogy between the interaction of information retrieval systems and humans to be similar to treatments in causal inference literature, we can use from the inverse propensity scoring (IPS) method (Rosenbaum and Rubin, 1983) in observational studies (Imbens and Rubin, 2015) that is used to debias estimators in case the treatment assignment mechanism is not uniform. Recently, IPS estimators have been employed for unbiased evaluation and learning in information retrieval systems (T. Schnabel, 2016), recommendation systems (Schnabel et al., 2016) and learning to rank framework (Schnabel et al., 2015; Joachims et al., 2017). In the past, weighting approaches have been largely successful in domain adaptation and co-variate shift, where the training and test datasets belong to different distributions (e.g. Bickel et al. (2009); Sugiyama and Kawanabe (2012)). Similarly, we use a weighted empirical risk minimization objective to debias the learning process while benefiting from the rigorous learning guarantees it provides (Schnabel et al., 2016).

6 Discussion

We present a modular way to learn embeddings while disentangling the biases present due to the system and the behavior of human users. Through our synthetic experiments, we show that the IPS estimator proposed is robust to increasing severity of the biases as well as model mismatch (as is the case in real world data). For real world experiments, we propose that the propensities can be estimated by assuming a click model and then doing an intervention experiment (similar to Joachims et al. (2017)). This work presents an application of the broader view of viewing recommender systems as assigning treatments to human users and the data reflecting the outcomes of these treatments. The hope is that this understanding enables us to explore the vast range of tools developed for causal inference in order to make learning for recommender systems better.

References

Oren Barkan and Noam Koenigstein. 2016. Item2vec: neural item embedding for collaborative filtering. In *Machine Learning for Signal Processing (MLSP), 2016 IEEE 26th International Workshop on*. IEEE, 1–6.

Steffen Bickel, Michael Brückner, and Tobias Scheffer. 2009. Discriminative learning under covariate shift. *Journal of Machine Learning Research* 10, Sep (2009), 2137–2155.

Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. 2016. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In *Advances in Neural Information Processing Systems*. 4349–4357.

Alexey Borisov, Ilya Markov, Maarten de Rijke, and Pavel Serdyukov. 2016. A neural click model for web search. In *Proceedings of the 25th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 531–541.

Olivier Chapelle and Ya Zhang. 2009. A dynamic bayesian network click model for web search ranking. In *Proceedings of the 18th international conference on World wide web*. ACM, 1–10.

Shuo Chen, J. Moore, D. Turnbull, and T. Joachims. 2012. Playlist Prediction via Metric Embedding. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*. 714–722.

Aleksandr Chuklin, Ilya Markov, and Maarten de Rijke. 2015. *Click Models for Web Search*. Morgan & Claypool. DOI:http://dx.doi.org/10.2200/S00654ED1V01Y201507ICR043

Trevor F Cox and Michael AA Cox. 2000. *Multidimensional scaling*. CRC press.

Nick Craswell, Onno Zoeter, Michael Taylor, and Bill Ramsey. 2008. An experimental comparison of click position-bias models. In *Proceedings of the 2008 International Conference on Web Search and Data Mining*. ACM, 87–94.

Amir Globerson, Gal Chechik, Fernando Pereira, and Naftali Tishby. 2007. Euclidean embedding of co-occurrence data. *Journal of Machine Learning Research* 8, Oct (2007), 2265–2295.

Jose Miguel Hernandez-Lobato, Neil Houlsby, and Zoubin Ghahramani. 2014. Probabilistic Matrix Factorization with Non-random Missing Data. In *Proceedings of The 31st International Conference on Machine Learning*. 1512–1520.

Guido W Imbens and Donald B Rubin. 2015. *Causal inference in statistics, social, and biomedical sciences*. Cambridge University Press.

T. Joachims, L. Granka, B. Pang, H. Hembrooke, and G. Gay. 2005. Accurately Interpreting Clickthrough Data as Implicit Feedback. In *ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*. 154–161.

T. Joachims, A. Swaminathan, and T. Schnabel. 2017. Unbiased Learning-to-Rank with Biased Feedback. In *International Conference on Web Search and Data Mining (WSDM)*.

Dawen Liang, Laurent Charlin, James McInerney, and David M Blei. 2016. Modeling user exposure in recommendation. In *Proceedings of the 25th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 951–961.

Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of Machine Learning Research* 9, Nov (2008), 2579–2605.

Benjamin M Marlin and Richard S Zemel. 2009. Collaborative prediction and ranking with non-random missing data. In *Proceedings of the third ACM conference on Recommender systems*. ACM, 5–12.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*. 3111–3119.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global Vectors for Word Representation.. In *EMNLP*, Vol. 14. 1532–1543.

Paul R Rosenbaum and Donald B Rubin. 1983. The central role of the propensity score in observational studies for causal effects. *Biometrika* (1983), 41–55.

Sam T Roweis and Lawrence K Saul. 2000. Nonlinear dimensionality reduction by locally linear embedding. *science* 290, 5500 (2000), 2323–2326.

T. Schnabel, A. Swaminathan, and T. Joachims. 2015. Unbiased Ranking Evaluation on a Budget. In *WWW Workshop on Offline and Online Evaluation of Web-based Services*.

T. Schnabel, A. Swaminathan, A. Singh, N. Chandak, and T. Joachims. 2016. Recommendations as Treatments: Debiasing Learning and Evaluation. In *International Conference on Machine Learning (ICML)*.

Masashi Sugiyama and Motoaki Kawanabe. 2012. *Machine learning in non-stationary environments: Introduction to covariate shift adaptation*. MIT press.

P. Frazier T. Joachims T. Schnabel, A. Swaminathan. 2016. Unbiased Comparative Evaluation of Ranking Functions. In *rxiv 1604.07209*.

Joshua B Tenenbaum, Vin De Silva, and John C Langford. 2000. A global geometric framework for nonlinear dimensionality reduction. *science* 290, 5500 (2000), 2319–2323.

Jason Weston, Samy Bengio, and Nicolas Usunier. 2011. Wsabie: Scaling up to large vocabulary image annotation. In *IJCAI*, Vol. 11. 2764–2770.