BiUCB: A Contextual Bandit Algorithm for Cold-Start and Diversified Recommendation

Lu Wang, Chengyu Wang, Keqiang Wang, Xiaofeng He*
Shanghai Key Laboratory of Trustworthy Computing,
School of Computer Science and Software Engineering, East China Normal University,
Shanghai, China
Email: joywanglulu@163.com, {chywang2013, sei.wkq2008}@gmail.com, xfhe@sei.ecnu.edu.cn

***Abstract*—In web-based scenarios, new users and new items frequently join the recommendation system over time without prior events. In addition, users always hold dynamic and diversified preferences. Therefore, cold-start and diversity are two serious challenges of the recommendation system. Recent works show that these problems can be effectively solved by contextual multi-armed bandit (CMAB) algorithms which consider the cold-start and diversified recommendation process as a bandit game. But existing methods only treat either items or users as arms, causing a lower accuracy on the other side. In this paper, we propose a novel bandit algorithm called binary upper confidence bound (BiUCB), which employs a binary UCB to consider both users and items to be arms of each other. BiUCB can deal with the item-user-cold-start problem where there is no information about users and items. Furthermore, BiUCB and k-ϵ-greedy can be combined as a switching algorithm which lead to significant improvement of the temporal diversity of entire recommendation. Extensive experiments on real world datasets demonstrate the precision of BiUCB and the diversity of switching algorithm.**

I. Introduction

Cold-start problem is pervasive in web-based scenarios, where there is no preference information of users or the items are new introduced [1]. Traditional recommendation systems suggest items based on the similarity or intersection of the history attributes. However, in real world, new users and new items frequently join the system over time, and the profiles of the pre-existing users update dynamically [2]. A content-based algorithm [3] solved user-cold-start with *demographic information*. Due to the difficulty of collecting these information, [4] [5] [6] use the information of neighborhoods. Other hybrid approaches as [7] can also solve cold start in a simply way.

Cold-start problem can also be naturally modeled as a CMAB problem. Multi-armed bandit problem (MAB) which derives from the gamble game is a simplified setting of reinforcement learning [8]. In order to earn the maximal sum of rewards, the gambler has two choices, one is trying to play some new arms of the multi-armed bandit which may have higher reward (exploration), while the other is sticking on playing the arm (exploitation) given the high reward so far.

In particular, upper confidence bound (UCB) is an effective method to solve CMAB, which suggests an arm with maximal confidence upper bound. LinUCB [9] extends UCB by considering contextual information about the arm. To recommend diversified items, [10] suggests a batch of items (called super arm) to each user with an entropy regularization.

Diversity is another important topic of recommendation, due to the dynamic and diverse preferences of users. In addition, recommending diversified items can help to catch the interests of users in cold start environment. Traditional diversified recommendation algorithms consider the difference among items. [11] [12] [13] deal with the top-k diversified recommendation based on balancing the diversity and relevance among items. [14] infers the users' novel preferences by historical features and demographic information. To prevent a redundancy and over-specified recommendation, [10] C2UCB uses *Entropy Regularization* [12] to improve the diversity of the super arm.

However, the standard CMAB algorithms overlook the dynamic states of items. That is, they assume the feature of the items invariant. But, when the number of interactions between users and items increases, we get more evidence to learn the feature vectors of users and items more precisely. In this paper, we assume the feature vectors of the item are variant.

We propose a new CMAB algorithm BiUCB to track the dynamic states of users and items, which assumes that both users and items to be arms for each other. The idea is that when the agent chooses an item for a user, it also means the agent selects the user to that item. For example, while the algorithm chooses the *Waiting to Exhale* for the user *Mary*, at the same time, it suggests *Mary* for *Waiting to Exhale* .

The main contributions on this work can be summarized as follows:

- We propose a new contextual multi-armed bandit algorithm called BiUCB, which can solve the cold-start and diversified problems in recommendation systems.
- BiUCB can effectively solve item-user-cold-start problem of recommendation.
- A switch algorithm combing BiUCB and k-ϵ-greedy enables to improve the temporal diversity of entire recommendation system.

In this paper, we briefly review related research in Section 2. Section 3 presents a formulation problem definition of BiUCB. We compare our approach with other methods in Section 4. Finally, we make a conclusion in Section 5.

*Corresponding author.

978-1-5386-3120-1/17 $31.00 © 2017 IEEE
DOI 10.1109/ICBK.2017.49

II. Related Work

Most traditional recommendation systems learn the preferences of user from the historical ratings [15] , profiles [16] [17], overlaps between users and items [18]. However, new users and new items frequently join the recommendation, we need to catch the dynamic states of both items and users. These cold-start problems can be naturally modeled as CMAB.

Take an example of using linear CMAB for cold start movie recommendation [9]. As a new user $\mathbf{z}_t$ come to the recommendation system, the agent observes the contexts $\mathbf{x}_{t,n}$ of the arms (movies) $\mathcal{S}_t$. The agent chooses an arm $x_t \in \mathcal{S}_t$ for the user based on a policy. We assume the true expectation of $r_{t,n}$ is linearly proportional to its context $\mathbf{x}_{t,n}$ with some unknown parameters $\mathbf{z}_t$. After receiving the feedback of user, the agent improves its policy.

Our goal is to learn a policy π to match decision between users and items with the observation of context.

In this section, we focus the researches which is related to our approach.

A. Contextual Multi-armed Bandit

In early researches, Lai and Robbins [19] first proposed the standard stochastic multi-armed bandit problem. Some traditional MAB algorithms extent to CMAB. Because contextual multi-armed bandit (CMAB) is more effective to model the real world.

ϵ-greedy is the earliest MAB algorithm. At each trial, it randomly selects an arm (exploration) to suggest with probability ϵ , and select the highest rewards arm (exploitation) with $1-\epsilon$.

Auer [20] extends CMAB into linear stochastic bandit LinREL, which assumes the reward and the context are linearly related. Li [9] and Chu [21] models the personality recommendation as a linear CMAB problem. Except for different regularization techniques, LinUCB is similar to LinREL.

UCB is an effective linear stochastic bandit algorithm, which is applied in personality recommendation extensively. CoFineUCB [22], LogUCB [23], LinUCB [9] are all UCB-style algorithms. LinUCB is used to solve personalized news article recommendation. Instead of suggesting a single arm, C2UCB[10] suggests a super arm each trial. This is a CMAB approach, which works on the movie recommendation. By adding an entropy regularization, it performs well on diversified recommendation. [24] proposed an ensemble method to aggregate diverse policies of bandit algorithms. [25] developed a time-varying CMAB approach to dynamically change the mapping between the reward and action.

To the best of our knowledge, none of the traditional bandits algorithms consider tracking both dynamic states of users and items. In fact, users continually interact with items, so that we get more evidence to learn the states of items and users more precisely. Consequently, we assume the states of users and items are both varying. As a result, we consider both items and users to be arms of each other.

B. Diversified Recommendation

Traditional diversified recommendation algorithms consider the difference among items. [11] [12] [13] deal with the top-k diversified recommendation. [12] proposed an entropy regularization for the PMF. Except for the difference among items, [13] also considers the coverage of user's interest. However, these work ignore the temporal diversity among the recommendation, which causes the same items suggested to users at multiple trials. To resolve this problem, we combine k-ϵ-greedy and entropy-based BiUCB as a switching algorithm, which significantly improve the diversity of recommendation system.

III. Algorithm

In this section, we introduce the preliminary knowledge of CMAB algorithm, and formulate BiUCB to solve cold-start recommendation. We also extend BiUCB to deal with diversified recommendation based on entropy regularization. Furthermore, we simply combines k-ϵ-greedy and BiUCB as a hybrid algorithm to effectively improve the temporal diversity.

A. Problem Formulation

In this section, we use an example to discuss the learning process of CMAB.

This is an example of recommendation process of CMAB. Suppose there are 5 movies *Toy Story*, *Waiting to Exhale*, *Grumpier Old Men*, *Sudden Death* and *Heat* in our recommendation system. At the t th trial, Each movie is represented as an attribute vector $\mathbf{x}_{t,n}$ ($n = 0, ..., 4$), and each user is represented as a preference vector $\mathbf{z}_t$. We can create the following design matrix $\mathbf{X}_t$:

$$\mathbf{X}_t = \begin{pmatrix} 0.3 & 0.5 & 0.8 & 0.2 & 0.1 \\ 0.7 & 0.1 & 0.1 & 0.7 & 0.8 \end{pmatrix}$$

each column denotes a movie. When a new user *Mary* ($\mathbf{z}_t = [0.7, 0.2]$) comes to the system, the predicted reward $\hat{\mathbf{r}} = \mathbf{x}_{t,n}^{\intercal}\mathbf{z}_t^*$ can be calculated as:

$$\hat{\mathbf{r}}^{\intercal} = \begin{pmatrix} 0.35 & 0.37 & 0.58 & 0.28 & 0.23 \end{pmatrix}$$

However, in oder to balance the exploration and the exploitation, we suggest the film with highest upper confidence bound which can be calculated by Eq. (2) instead of the one with highest predicted reward for *Mary*. Suppose the UCB of all movies equals $\mathbf{p}_t$:

$$\mathbf{p}_t^{\intercal} = \begin{pmatrix} 0.4 & 0.6 & 0.5 & 0.2 & 0.2 \end{pmatrix}$$

Hence, the movie *Waiting to Exhale* is suggested for *Mary*. If *Mary* chooses it, we receive a positive reward. Otherwise, we receive 0. At the same time, $\mathbf{z}_t$, the preference vector of *Mary*, should be adjusted by the reward to track the dynamic preference of *Mary*.

In fact, both users and items are affected by the reward, because with the number of rewards arise, more evidence generates, so that the movie's feature can be learn more precisely. Therefore, BiUCB adjusts $\mathbf{z}_t$ and $\mathbf{x}_{t,1}$ by the reward simultaneously.

B. BiUCB with Linear Model

In this section, we solve the cold-start problem by BiUCB. BiUCB contains two CMAB models: B1UCB and B2UCB. We call them both BiUCB. We consider users and items are both arms to each other. While the B1UCB chooses the items for each user positively, the B2UCB chooses the user for each item passively.

Formally, let T be the total number of trials and N be the total number of items. Let $\mathcal{S}_t \subseteq 2^{[N]}$ be the candidate item set. $\mathbf{x}_{t,1}, ..., \mathbf{x}_{t,n} \subseteq \mathbb{R}^d$ and $\mathbf{z}_t \subseteq \mathbb{R}^d$ corresponding to n items and users at trial t separately. Following the previous work, we call the set $\mathbf{S}_t \subseteq \mathcal{S}_t$ super arm, which are suggested to the user at trial t.

Using the notation in section 1, we assume that at each trial t ($t = 1, 2...T$), the task of BiUCB is to suggest a super arm to each user based on the selection policy. Meanwhile BiUCB suggests an arm (user) passively to each item. After choosing the items for the user, BiUCB receives a reward $\mathbf{r}_{t,n}$ of the super arm. Formally, the reward can be described as:

$$\mathbf{E}[\mathbf{r}_{t,n} \mid \mathbf{x}_{t,n}, \mathbf{z}_t] = \mathbf{x}_{t,n}^{*\intercal} \mathbf{z}_t^* + \epsilon_{t,n}$$

where $\epsilon_{t,n} \sim \mathcal{N}(0, \sigma^2)$ and the $\mathbf{x}_{t,n}^*$ denotes the unknown item features, and $\mathbf{z}_t^*$ is the unknown preference of user . The prior distribution of $\mathbf{r}_{t,n}, \mathbf{z}_t, \mathbf{x}_{t,n}$ defined as:

$$P(\mathbf{r}_{t,n} \mid \mathbf{z}_t, \mathbf{x}_n, \sigma^2) \sim \mathcal{N}(\mathbf{x}_{t,n}^\intercal \mathbf{z}_t, \sigma^2)$$

$$P(\mathbf{z}_t \mid \sigma_z) \sim \mathcal{N}(0, \sigma_z^2), P(\mathbf{x}_n \mid \sigma_x) \sim \mathcal{N}(0, \sigma_x^2)$$

Let $\mathbf{X}_{t,n} \sim \mathbb{R}^{n \times d}$ and $\mathbf{Z}_t \sim \mathbb{R}^{m \times d}$ be a design matrix, whose row denotes m training records. $\mathbf{r}_t \sim \mathbb{R}^n$ corresponds to the rewards of the user. Therefore the regret (object function) of the algorithm can drive from the log of posterior distribution, which is described as:

$$J(\mathbf{Z}, \mathbf{X}, \mathbf{R}) = \frac{1}{2}\|\mathbf{R} - \mathbf{Z}\mathbf{X}^\intercal\|_F^2 + \frac{\lambda_Z}{2}\|\mathbf{Z}\|_F^2 + \frac{\lambda_X}{2}\|\mathbf{X}\|_F^2$$

As for user $\mathbf{z}_t$:

$$J(\mathbf{z}_t, \mathbf{X}, \mathbf{R}) = \frac{1}{2}\|\mathbf{r}_t - \mathbf{X}\mathbf{z}_t\|^2 + \frac{\lambda_z}{2}\|\mathbf{z}_t\|^2$$

As for each item $\mathbf{x}_n$:

$$J(\mathbf{Z}, \mathbf{x}_n, \mathbf{R}) = \frac{1}{2}\|\mathbf{r}_{t,n} - \mathbf{Z}\mathbf{x}_n\|^2 + \frac{\lambda_x}{2}\|\mathbf{x}_n\|^2$$

While suggesting items to the user, we consider the items as arms, and choose an arm with maximized upper confidence bound. We can naturally get the closed form of the estimate of $\hat{\mathbf{z}}_t$:

$$\hat{\mathbf{z}}_t = (\mathbf{X}_t^\intercal \mathbf{X}_t + \lambda_Z \mathbf{I}_d)^{-1} \mathbf{X}^\intercal \mathbf{r}_t$$

Based on the UCB-style strategy, BiUCB suggests the maximal upper confidence bound among these arms (Line 6), and there is a high probability bound (Eq.1) on the selected items, which is described as:

$$\mathbf{x}_t \overset{def}{=} \arg\max_{x \in \mathcal{S}_t} (\mathbf{x}_{t,n}^\intercal \hat{\mathbf{z}}_t + \alpha \sqrt{\mathbf{x}_{t,n}^\intercal \mathbf{V}_t^{-1} \mathbf{x}_{t,n}})$$

where $\mathbf{V}_t \overset{def}{=} \mathbf{X}_t^\intercal \mathbf{X}_t + \mathbf{I}_d$. The agent adjust its policy by optimizing the preference of the user $\mathbf{z}_t$ (Line 4) . As we mentioned before, each arm passively chooses the user. In this way, the reward can be shared among items and users. Consequently, we track both the dynamic states of users and items at same time. Note that we assume the profile of items are invariant, but their feature vectors keep varying.

While suggesting the user for items, the use is regard as the arm, and we need not choose an arm, for there is a single user for a specific item each trial. Let $\mathbf{Z}_t \sim \mathbb{R}^{m \times d}$ be a design matrix. $\mathbf{r_{t,n}} \sim \mathbb{R}^m$ corresponding to the feedback of the user. We can also naturally get the closed form of the estimate of $\hat{\mathbf{x}}_n$:

$$\hat{\mathbf{x}}_{t,n} = (\mathbf{Z}_t^\intercal \mathbf{Z}_t + \lambda_X \mathbf{I}_d)^{-1} \mathbf{Z}^\intercal \mathbf{r}_t$$

Algorithm 1 describes the details of entire BiUCB algorithm, where $\mathbf{b}_{t,n}^x = \mathbf{Z}_t^\intercal \mathbf{r}_{t,n}$, $\mathbf{b}_t^z = \mathbf{X}_t^\intercal \mathbf{r}_t$, and $\mathbf{V}_t \overset{def}{=} \mathbf{X}_t^\intercal \mathbf{X}_t + \mathbf{I}_d$, $\mathbf{U}_{t,n} \overset{def}{=} \mathbf{Z}_t^\intercal \mathbf{Z}_t + \mathbf{I}_d$.

Algorithm 1 BiUCB

Input: $\lambda_X, \lambda_Z, \alpha_1, \ldots, \alpha_n$

1: Initialize $\mathbf{V}_0 \leftarrow \lambda \mathbf{I}_{d\times d}, b_0^{(z)} \leftarrow \mathbf{0}_d, \mathbf{U}_{0,0} \leftarrow \lambda \mathbf{I}_{d\times d}, b_{0,0}^{(x)} \leftarrow \mathbf{0}_d$
2: **for** $t = 1, 2, 3 \ldots$,T **do**
3: $\quad \hat{\mathbf{z}}_{t,m} \leftarrow \mathbf{V}_{t-1}^{-1} \mathbf{b}_{t-1}^z$
4: $\quad$ **for** $n = 1, 2, 3 \ldots$,N **do**
5: $\quad\quad \hat{\mathbf{x}}_{t,n} \leftarrow \mathbf{U}_{t-1,n}^{-1} \mathbf{b}_{t-1,n}^x$
6: $\quad\quad p_{t,n} \leftarrow \hat{\mathbf{z}}_t^\intercal \hat{\mathbf{x}}_{\mathbf{t},\mathbf{n}} + \alpha \sqrt{\hat{\mathbf{x}}_{t,n}^\intercal \mathbf{V}_t^{-1} \hat{\mathbf{x}}_{t,n}}$
7: $\quad$ **end for**
8: $\quad$ Choose item $\mathbf{x}_t = \arg\max_{i \in \mathcal{S}_t} p_{t,n}$ for user $\mathbf{z}_t$
9: $\quad$ Choose user $\mathbf{z}_t$ for item $\mathbf{x}_t$
10: $\quad \mathbf{V}_t \leftarrow \mathbf{V}_{t-1} + \mathbf{x}_t \mathbf{x}_t^\intercal$
11: $\quad \mathbf{b}_t^z \leftarrow \mathbf{b}_{t-1}^z + r_t \mathbf{x}_t$
12: $\quad \mathbf{U}_{t,n} \leftarrow \mathbf{U}_{t-1,n} + \hat{\mathbf{z}}_t \hat{\mathbf{z}}_t^\intercal$
13: $\quad \mathbf{b}_{t,n}^x \leftarrow \mathbf{b}_{t-1,n}^x + r_t \hat{\mathbf{z}}_t$
14: **end for**

C. Diversified Contextual Multi-Armed Bandits

In this section, we apply BiUCB to recommend diverse items. Following the previous work, we add an oracle function $\mathcal{O}_{\mathbf{S}_t}(\hat{\mathbf{r}}, \mathbf{X})$ into the reward function to measure both diversity and relevance. The output of oracle is a super arm achieves the maximum of $f_t(\mathbf{S})$. In particular, the oracle can solve the combination optimization problem with $1-1/e$ approximation. We introduce three oracle approaches in this section.

1) UCB with Maximum Margin Relevance: Specifically, the reward function with MMR can be described as:

$$f_t(\mathbf{S}) = \sum_{i_n \subseteq \mathbf{S}} \mathbf{x}_{t,i_n}^\intercal \mathbf{z}_t + \lambda_m \sum_{i_n, j_n \subseteq \mathbf{S}} \|\mathbf{x}_{t,i_n} - \mathbf{x}_{t,j_n}\|$$

And the expected reward function can be defined as $E[R_{(t)}] = f_r {S_t}^*$.

2) BiUCB with Entropy Regularization: Formally, we extend the reward function of BiUCB with an entropy regularization as [12], which is described as $h(\mathbf{S}, \mathbf{X}) = \frac{1}{2}|\mathbf{S}| \log(2\pi e) + \frac{1}{2} \log\det(\mathbf{X}(\mathbf{S})^\intercal \mathbf{X}(\mathbf{S}) + \sigma^2 \mathbf{I}_d)$. *Entropy Regularization* significantly improves the diversity of entire recommendation systems. However, it overlooks the temporal diversity. In the experiment section, it always suggests the same items to the user over and over again.

3) Temporal User-based Switching : Temporal user-based switching algorithm was proposed in [2]. Learning from the idea of hybrid CF, we construct a new switching algorithm which is straightforward to switch the k-ϵ-greedy and BiUCB at each trial. The principle of this method is that the different algorithm always suggest different items. Furthermore, at each trial, k ϵ-greedy randomly selects arms with a probability ϵ, which implicitly reranks [2] the top-k list.

IV. Performance Studies

In this section, we describe the experiment settings about evaluation metrics, feature construction, baseline algorithms, and comparison results. In particular, the experiments can be partitioned into two parts. One applies BiUCB into solve user-cold-start and diversified recommendation problems. The other conducts BiUCB to solve user-item-cold-start problem.

A. Evaluation Metrics

In this paper, we consider 3 main evaluation metrics: precision, temporal-diversity and novelty [2]. Formally, let $\mathbf{S}_t$ be the super arm, $\mathbf{I}_u$ be the item set the user rated and p_t be the precision of user $\mathbf{z}_t$. At the trial t, pr_t can be defined as:

$$pr_t = \frac{|\mathbf{S}_t \cap \mathbf{I}_u|}{\mathbf{S}_t}$$

while the average of precision of all users in T trials can be described as:

$$Pr_t = \frac{1}{T|\mathbf{Z}|} \sum_{\mathbf{z} \in \mathbf{Z}} \sum_{t=1}^{T} pr_t$$

Temporal-diversity measures the difference between two ranked lists recommended to user i at trial t and t+1 [2]. Formally, the diversity between two recommendation lists can be described as:

$$d_t(\mathbf{S}_{t+1}, \mathbf{S}_t, K) = \frac{|\mathbf{S}_{t+1} \backslash \mathbf{S}_t|}{K}$$

where $\mathbf{S}_{t+1} \backslash \mathbf{S}_t = \hat{\mathbf{S}} \in \mathbf{S}_{t+1} \mid \hat{\mathbf{S}} \notin \mathbf{S}_t$. The average temporal diversity of all users in T trials can be described as:

$$D_t = \frac{1}{T|\mathbf{Z}|} \sum_{\mathbf{z} \in \mathbf{Z}} \sum_{t=1}^{T} (\mathbf{S}_{t+1}, \mathbf{S}_t, K) = \frac{|\mathbf{S}_{t+1} \backslash \mathbf{S}_t|}{K}$$

However, temporal diversity only concerns on the local diversity. To observe the entire recommendation diversity, we use novelty [2] to be the metric. Formally, at trial t, the novelty of the user $\mathbf{z}_t$ can be described as :

$$novelty(\mathbf{S}_t, K) = \frac{|\mathbf{S}_t \backslash \mathcal{S}_t|}{K}$$

while the average novelty of all users among T trials can be handled analogously.

B. Data Collection

We experimented on real world dataset from MovieLens [26], which consists 1,000,209 ratings for 3952 movies by 6040 users of online movie recommendation service. The dataset is represented by a range of tuples: $t_{m,n} = (z_m, x_n, r_{m,n})$, where z_m is the userID, x_n is the movieID, and $r_{m,n}$ is the rating user z_m gives for x_n. In addition, the rating is made a 5-star scale.

C. Feature Generation

We divided the dataset into three parts, first is used to generate initial item features, second is used to learning to tract the preference of user, third is testing set. We randomly select 300 users who have at least 100 ratings, and divide it roughly half to be the second dataset and the other to be the testing set. The remaining 5740 users and their rating records consist the first dataset. The initial item features are learned by probabilistic matrix factorization (PMF) with k-ranked on the first dataset. When dealing with the item-user-cold-start problem, we don't use the first dataset, because we assume both of users and items are unknown.

D. Baselines

We conduct BiUCB to solve diversified and cold-start problems in recommendation, and compare with five baselines. The experiments design focus on training precision (NS-precision), testing precision (S-precision), diversity and the novelty.

C2UCB algorithm. C2UCB [10] applies the Entropy Regularization into contextual bandit algorithm which recommends a diversified super arm at each trial. The major difference between C2UCB and BiUCB is that BiUCB considers both of the user and the item as arms, and can solve the item-user cold start problem.

k-LinUCB algorithm. LinUCB [9] recommends one arm for a specific user at each trial. To compare with other approaches, we execute LinUCB k times to guarantee suggesting a super arm with k arms each trial.

MMR-UCB algorithm. As we mentioned in section 3, MMR-UCB applies the MMR approach into the UCB-style algorithm.

k-ϵ-greedy algorithm. At each trial, the best item is suggested with a probability $1-\epsilon$, and a random item is suggested with probability ϵ. As LinUCB, we execute this algorithm k times to achieve a super arm.

Warm Start. Warm-start acts as an initialization on offline estimate for users' preferences. Formally, $w = 2 \times k$ indicates we randomly select w ratings from dataset 2 to lean the preferences of users by PMF offline. In addition, $w = all$ means using all ratings to learn the user preference, which can be treated as the best solution.

E. Quality of the Recommendation

Fig.1 compares the ns-precision, s-precision, diversity and novelty among the algorithms, where ns-precision measures the precision of training set, and s-precision evaluates the testing set.

TABLE I
RELATIVE PRECISION, DIVERSITY AND NOVELTY ON MOVIELENS DATA

Algorithm	ns-precision	s-precision	diversity	novelty
ϵ-greedy(0.1)	0.435	0.724	0.266	0.218
ϵ-greedy(0.3)	0.396	0.633	0.457	0.400
ϵ-greedy(0.5)	0.303	0.440	0.520	0.533
k-LinUCB(0.5)	0.515	0.776	0.345	0.246
k-LinUCB(0.25)	0.481	0.883	0.118	0.153
k-LinUCB(0.1)	0.514	0.778	0.332	0.261
C2UCB(0.5)	0.490	0.904	0.153	0.150
C2UCB(0.3)	0.508	0.786	0.327	0.264
C2UCB(0.1)	0.516	0.784	0.330	0.266
MMR-UCB(0.6)	0.350	0.429	0.537	0.425
MMR-UCB(0.4)	0.352	0.443	0.536	0.421
MMR-UCB(0.2)	0.371	0.587	0.183	0.144
BiUCB(0.5)	0.542	**0.966**	0.148	0.154
BiUCB(0.3)	**0.580**	0.827	0.377	0.360
BiUCB(0.1)	0.579	0.828	0.375	0.360
TS(0.5)	0.535	0.713	0.396	0.301
TS(0.33)	0.539	0.612	**0.650**	**0.540**
TS(0.2)	0.543	0.653	0.639	0.504

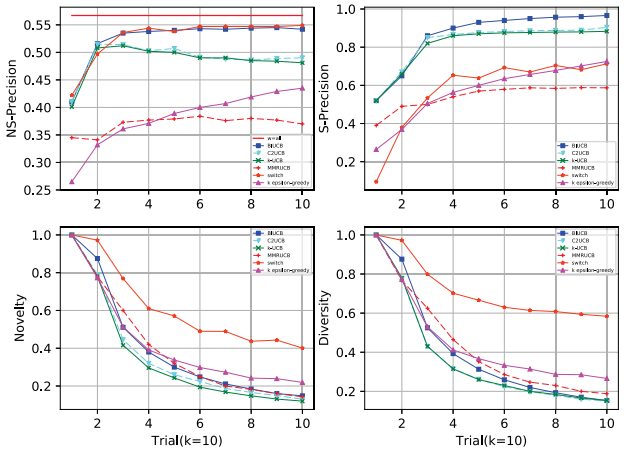

Fig. 1. Baselines Comparison on Different Metrics

As the trial number increase, the precision gradually raises, but the diversity and novelty show a negative trend. From our experimental results, we establish that:

- C2UCB and k-LinUCB learn the preference of users effectively.
- But results from C2UCB and k-LinUCB are not comparative with BiUCB in testing dataset.
- The hybrid algorithm significantly improves the novelty and temporal diversity.

1) Performance Comparison for Precision: As shown in Fig. 1, BiUCB achieves the highest precision among the baselines. Tracking both the dynamics preferences of users and item features enables BiUCB show a good performance. Simply considering the Euclidean distance between items to make diversified recommendation, causes MMR-UCB to achieve lower precision. k-ϵ-greedy randomly selects arms to obtain users' preference and causes an undesirable precision at the first 3 trials. While obtaining more information, k-ϵ-greedy gets better.

C2UCB and BiUCB both outperform k-LinUCB by considering the diversity items in the recommendation lists. By tracking both dynamic states of users and items, BiUCB describes the users and items more precisely. This enables BiUCB to achieve higher performance than C2UCB.

Temporal Switch (TS) proves higher NS-precision than S-precision, since it always try to suggest fresh items to the user, which decreases the reward of TS in the training process. While in testing process, the agent tries some new items for the user. This enables TS performs better.

2) Performance Comparison for Diversity and Novelty: As shown in Table 1, algorithms with high precision achieves low diversity and novelty. Because, it is risky to explore the diversified and fresh items for users, but much exploiting will limit the diversity of items. The switching algorithm well balances the exploration-exploitation dilemma. Because randomly suggesting items for users with ϵ probability enables to rerank the recommendation lists. As shown in Fig. 1, both of diversity and novelty show a negative trend. That is, the more items user select, the less diversity will the algorithm suggests.

TS greatly increases the diversity and novelty of the recommendation and it is better than any of the two single algorithm. Because different algorithms may suggest variant items, and TS can rerank the top-k items. This enables to improve the diversity and novelty of the recommendation.

3) Sensitivity to Parameters: We conduct each algorithm with different parameter setting. Finally, all baseline algorithms are configured with their best parameters (s-precision) provided by Table 1. After 10 trials of each user, the algorithm BiUCB (0.5) outperforms the other algorithms on precision. and TS (0.33) achieves the most diversity and novelty.

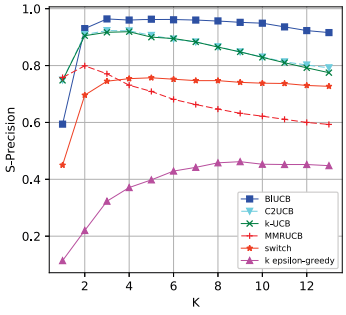

Fig. 2. Relative Precision on different Size of Super Arm

4) Impact of the Super Arm Size: An important parameter of the baselines is the number of super arms K, ($K \in [1, 13]$). Fig. 2 reports the precision performance in variant K. Intuitively, with K increasing, BiUCB achieves higher precision over C2UCB and k-LinUCB, because BiUCB can help track the diversified preferences of users, which makes it always recommends true arms even in large candidates.

5) Performance on item-user-cold start: Another important difference between BiUCB and the other algorithms is that it

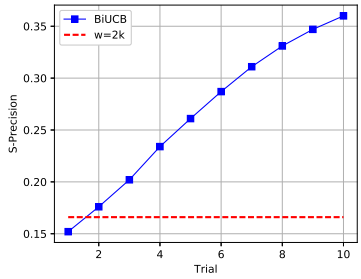

Fig. 3. Precision of the BiUCB on Item-User-cold-start

can deal with the item-user-cold problem, while the others can only deal with single-cold-start problem. We experiment BiUCB on the second dataset to discover the performance of precision. As the result shown in Fig.3, it learns the user's preference with a fast rate, and the precision exceeds the warm start with 2k ratings of each user.

V. Conclusion

This paper investigates the cold-start and diversified problems in recommendation. We propose a novel contextual multi-armed bandit model BiUCB for these issues. In particular, we assume the states of both users and items are dynamic. To handle these dynamics, BiUCB employs a binary UCB to respectively adjust the item-selection policy. Furthermore, BiUCB can also deal with item-user-cold-start problem. Combining BiUCB with k-ϵ-greedy as a switching algorithm enables significant improvement of the temporal diversity of entire recommendation. Extensive experiments demonstrate that the precision of BiUCB and the diversity of switching algorithm combined BiUCB is much better than the existing bandits algorithms.

Acknowledgements

This work is supported by the National Key Research and Development Program of China under Grant No. 2016YFB1000904.

References

[1] Andrew I Schein, Alexandrin Popescu, Lyle H Ungar, and David M Pennock. Methods and metrics for cold-start recommendations. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 253–260. ACM, 2002.

[2] Neal Lathia, Stephen Hailes, Licia Capra, and Xavier Amatriain. Temporal diversity in recommender systems. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 210–217. ACM, 2010.

[3] Michael J Pazzani and Daniel Billsus. Content-based recommendation systems. In *The adaptive web*, pages 325–341. Springer, 2007.

[4] Zi-Ke Zhang, Chuang Liu, Yi-Cheng Zhang, and Tao Zhou. Solving the cold-start problem in recommender systems with social tags. *EPL (Europhysics Letters)*, 92(2):28002, 2010.

[5] Suvash Sedhain, Scott Sanner, Darius Braziunas, Lexing Xie, and Jordan Christensen. Social collaborative filtering for cold-start recommendations. In *Proceedings of the 8th ACM Conference on Recommender systems*, pages 345–348. ACM, 2014.

[6] Shaghayegh Sahebi and William W Cohen. Community-based recommendations: a solution to the cold start problem. In *Workshop on recommender systems and the social web, RSWEB*, 2011.

[7] Asela Gunawardana and Christopher Meek. A unified approach to building hybrid recommender systems. In *ACM Conference on Recommender Systems, Recsys 2009, New York, Ny, Usa, October*, pages 117–124, 2009.

[8] R. S Sutton and A. G Barto. Reinforcement learning : an introduction. *IEEE Transactions on Neural Networks*, 9(5):1054, 1998.

[9] Lihong Li, Wei Chu, John Langford, and Robert E Schapire. A contextual-bandit approach to personalized news article recommendation. In *International Conference on World Wide Web*, pages 661–670, 2010.

[10] Lijing Qin, Shouyuan Chen, and Xiaoyan Zhu. Contextual combinatorial bandit and its application on diversified online recommendation. In *Proceedings of the 2014 SIAM International Conference on Data Mining*, pages 461–469. SIAM, 2014.

[11] Azin Ashkan, Branislav Kveton, Shlomo Berkovsky, and Zheng Wen. Optimal greedy diversity for recommendation. In *International Conference on Artificial Intelligence*, pages 1742–1748, 2015.

[12] Lijing Qin and Xiaoyan Zhu. Promoting diversity in recommendation by entropy regularizer. In *International Joint Conference on Artificial Intelligence*, pages 2698–2704, 2013.

[13] Chaofeng Sha, Xiaowei Wu, and Junyu Niu. A framework for recommending relevant and diverse items. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, pages 3868–3874. AAAI Press, 2016.

[14] Fuzheng Zhang, Kai Zheng, Nicholas Jing Yuan, Xing Xie, Enhong Chen, and Xiaofang Zhou. A novelty-seeking based dining recommender system. In *Proceedings of the 24th International Conference on World Wide Web*, WWW '15, pages 1362–1372. International World Wide Web Conferences Steering Committee, 2015.

[15] Gueorgi Kossinets and Duncan J Watts. Empirical analysis of an evolving social network. *science*, 311(5757):88–90, 2006.

[16] Fabrizio Sebastiani. Machine learning in automated text categorization. *ACM computing surveys (CSUR)*, 34(1):1–47, 2002.

[17] Raymond J Mooney and Loriene Roy. Content-based book recommending using learning for text categorization. In *Proceedings of the fifth ACM conference on Digital libraries*, pages 195–204. ACM, 2000.

[18] Nira Dyn. *Multivariate approximation and applications*. Cambridge university press, 2001.

[19] Tze Leung Lai and Herbert Robbins. Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics*, 6(1):4–22, 1985.

[20] Peter Auer. Using confidence bounds for exploitation-exploration tradeoffs. *Journal of Machine Learning Research*, 3(Nov):397–422, 2002.

[21] Wei Chu, Lihong Li, Lev Reyzin, and Robert E Schapire. Contextual bandits with linear payoff functions. In *AISTATS*, volume 15, pages 208–214, 2011.

[22] Yisong Yue, Sue Ann Hong, and Carlos Guestrin. Hierarchical exploration for accelerating contextual bandits. *arXiv preprint arXiv:1206.6454*, 2012.

[23] Dhruv Kumar Mahajan, Rajeev Rastogi, Charu Tiwari, and Adway Mitra. Logucb: an explore-exploit algorithm for comments recommendation. In *ACM International Conference on Information and Knowledge Management*, pages 6–15, 2012.

[24] Liang Tang, Yexi Jiang, Lei Li, and Tao Li. Ensemble contextual bandits for personalized recommendation. In *ACM Conference on Recommender Systems*, pages 73–80, 2014.

[25] Chunqiu Zeng, Qing Wang, Shekoofeh Mokhtari, and Tao Li. Online context-aware recommendation with time varying multi-armed bandit. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 2025–2034, 2016.

[26] Movielens. http://movielens.org.