Modeling Consumer Buying Decision for Recommendation Based on Multi-Task Deep Learning

Qiaolin Xia¹, Peng Jiang², Fei Sun², Yi Zhang³, Xiaobo Wang², and Zhifang Sui¹.
¹MOE Key Laboratory of Computational Linguistics, EECS, Peking University
²Alibaba Group, Beijing, China
³University of California, Santa Cruz, USA
{xql,szf}@pku.com, {jiangpeng.jp, ofey.sf, yongshu.wxbl}@alibaba-inc.com, yiz@soe.ucsc.edu

ABSTRACT
Although marketing researchers and sociologists have recognized the importance of buying decision process and its significant influence on consumer’s purchasing behaviors, existing recommender systems do not explicitly model the consumer buying decision process or capture the sequential regularities of what happens before and after each purchase. In this paper, we try to bridge the gap and improve recommendation systems by explicitly modeling consumer buying decision process and corresponding stages. In particular, we propose a multi-task learning model with long short-term memory networks (LSTM) to learn consumer buying decision process. It maps items, users, product categories, and the behavior sequences into real valued vectors, with which the probability of purchasing a product can be estimated. In this way, the model can capture user intentions and preferences, predicts the conversion rate of each candidate product, and makes recommendations accordingly. Experiments on real world data demonstrate the effectiveness of the proposed approach.

CCS CONCEPTS
• Information systems → Recommender systems; Users and interactive retrieval; Decision support systems;

KEYWORDS
User Modeling; Recommendation; Multi-Task Learning

ACM Reference Format:

1 INTRODUCTION
Marketing researchers and sociologists have studied the implicit buying decision process on consumer’s purchasing behaviors for many years [1, 6]. A buying decision process describes a number of stages a consumer goes through before and after buying a particular product. It usually involves several stages, e.g., need recognition stage, product research, and information gathering stage, consideration stage, purchasing stage, and post-purchase feedback stage. Understanding such stages is of paramount importance to providing more personalized experiences, which can lead to greater improvement for recommendation systems.

As an example, Figure 1 shows how the conversion rate¹ is affected by the user behaviors at the product research stage. It reveals that the conversion rate increases as the count and days of research stage increase. However, too many research behaviors will lead to a poor conversion rate. For a consumer at this stage, the recommender system may want to provide the products that satisfy the similar need with competitive prices and good reviews for customers to compare with, so that the consumers are more likely to move forward to the next stage. Besides, the behavior at one purchasing stage greatly influences the behavior at the next stage and the probability of continuing to the next stage.

Based on these observations, in this paper, we try to explicitly model the consumer’s buying decision process and stages for helping predict the conversion rate of products, which is a core of e-commerce recommendation algorithms. Recently, there is a lot of work focusing on modeling the effect of time [5, 8, 9, 11] or user behaviors [4, 10] to improve recommendation systems. These works study the drift of user preferences over time via factorization models or neural sequence models. However, none of them try to explicitly model the buying decision process for product recommendation, especially in the context of a large scale e-commerce system. To the best of our knowledge, this is the first work focusing on modeling the buying decision process in recommendation systems.

¹Conversion rate is the probability of a click action leading to a purchasing action.
In this work, we explore several deep learning based recommendation solutions that explicitly model the consumer buying decision process while predicting product conversing rate. In particular, we adopt the Long Short-Term Memory Network (LSTM) [3] to learn the representations of the current stage of the consumer based on his/her behavior sequences. LSTM is designed to model sequential data (e.g., natural language), and has achieved great success on various sequential prediction tasks such as machine translation and text summarization. In this paper, we propose several variations of LSTM architecture that are capable of integrating different consumer buying behaviors and stages.

To evaluate the effectiveness of the proposed approach, we conduct extensive experiments with real user data collected from Taobao.com. The results demonstrate that the proposed approach can significantly improve the performance in terms of conversion rate prediction accuracy, a key metric of recommendation algorithms.

2 PROBLEM DEFINITION

To integrate buying decision process into recommendation models, we first introduce the problem definition of the core recommendation task: predicting the direct buying conversion rate of each candidate product at each time point. Then we describe the augmented problem definition: modeling the buying decision process and predicting the stage at each time point.

2.1 Direct Buying Prediction

Let $U = \{u_1, u_2, ..., u_{|U|}\}$ be a set of users, and $I = \{i_1, i_2, ..., i_{|I|}\}$ be a set of items. For a user $u$, the online click sequence $C_u$ can be represented as $[c_1^u, c_2^u, ..., c_{|C_u|}^u]$, and the corresponding time sequence is $([t_1, t_2, ..., t_{|C_u|})]$. After each click, the user may have several behaviors on the detail page of the active product, such as adding to the product to the cart, marking it as a favorite product, or reading some reviews. We denote these behaviors as $A = \{a_1^u, a_2^u, ..., a_{|A|}^u\}$.

In this paper, the behaviors in $A$ occur either on the product detail page after a click\(^7\) or on product cart page. For the latter cases, we associate the buying behaviors with the last click on the same product. These behaviors reflect a customer’s different intentions and buying decision stages.

**Definition 2.1. (Direct Buying)** Given a user $u$, his/her click $c_{tm}^u$ on an item $i$ at time $t_m$, and a buying behavior $a_{tm}^u$ on item $i$ at $t_m$ ($t_m > t_m$), if there is no click on item $i$ in the time span ($t_m, t_n$), we call the buying behavior $a_{tm}^u$ a Direct Buying for the click $c_{tm}^u$.

Besides, there is an effective complementary technology used in the industry to increase the conversation, which is predicting the will buy conversation rate, which is defined as follows:

**Definition 2.2. (Will Buy Predicting)** Given a user $u$, his/her click $c_{tm}^u$ on an item $i$ at time $t_m$, and a buying behavior $a_{tm}^u$ on item $i$ at $t_m$ ($t_m > t_m$), if there exists other clicks on the item $i$ in the time span ($t_m, t_n$), we call the behavior $a_{tm}^u$ as a Will Buy for the click $c_{tm}^u$.

Some recommendation systems use will buy conversion rate instead of direct buy conversion rate. Therefore, we also conduct further experiments on will buy predicting in Section 4.

2.2 Buying Decision Stage Prediction

The conversion rate is affected by the decision stages in the whole buying decision process. Therefore, we introduce the task of predicting which buying decision stage a user is when the user clicks a product $i$ at time $t_m$, which is represented as $[<u, i, t_m>]$ triplet.

However, a user’s buying decision stage is a hidden variable which is not observable for an e-commerce system, thus we cannot directly acquire this information. Fortunately, after clicking a product, a user’s behaviors related to this product tells a lot about which stage the user is at for a particular time point. For simplicity, we write the following set of heuristic rules to automatically assign a stage value to time point $t_m$ when a user clicks on product $i$.

**Need-recognition stage:** Given a click $[<u, i, t_m >]$, if this is the first time the user clicks the product, we assume the user is at need-recognition stage at time $t_m$.

**Research stage:** Given a product click $[<u, i, t_m >]$, if there is a look-at-comment, ask-the-seller or look-at-question All behavior by user $u$ at a later time point $t_n$, and the user has no other click on the same item during $t_m$ to $t_n$, then we assume the user is at research stage at time $t_m$.

**Consideration stage:** Given a click $[<u, i, t_m >]$, if there is add-to-cart or mark-as-favorite behavior on item $i$ by user $u$ at a later time point $t_n$, and the user has no other click on item $i$ between $t_m$ and $t_n$, we assume the user is at consideration stage at time $t_m$.

**Buying stage:** Given a product click $[<u, i, t_m >]$, if the user clicks the “buy” button on item $i$ later at time $t_n$, and the user has no other click on item $i$ between $t_m$ and $t_n$, we assume the user is at buying stage at time $t_m$.

**Feedback stage:** Given a product click $[<u, i, t_m >]$, if the user comments on the product $i$ at later time point $t_n$, and the user has no other click on item $i$ between $t_m$ and $t_n$, we assume the user is at feedback stage at time $t_m$.

The above rules give us a large amount of pseudo labeled data that include product click sequences and the buying stage at each time point.

3 MODELING BUYING DECISION WITH MULTI-TASK LSTM

Given a product click $[<u, i, t_m >]$ and all user behaviors until time $t_m$, the task of predicting whether the user is at the buying stage at $t_m$ is almost the same as the task of predicting “direct buying” conversion rate defined in Section 2.1, which is the core problem for recommendation systems. On the other hand, predicting whether a user is at Research, Consideration, or Feedback stage can be viewed as related tasks. This view motivates us to use a multi-task sequence modeling approach for the product recommendation.

The general architecture is a four-layer neural network as illustrated in Figure 2. The first layer is the input layer. Give a user $u$, for each click $c_{tm}^u$ on item $i$, we extract the user feature vector $f_u$, item feature vector $f_i$, and user-item interaction feature vector $f_{ij}$ at time $t$ as the input of the first layer. The above features can be organized into two kinds of vectors: i) numerical feature vector $x_t^u \in \mathbb{R}^N$, where $N$ is the total number of numerical features, and ii) embedding feature vectors $(x_{t, 1}^u, x_{t, 2}^u, ..., x_{t, K}^u)$, where $x_{t, k}^u \in \mathbb{R}^V_k$ is a one-hot-vector representation of the $k^{th}$ embedding feature.
generate a rich feature set to capture that valuable information. In general, all features fall into three feature groups: item features, user features, and user-item interaction features. The item feature group includes the attributes (e.g., category Id, brand Id, and seller Id) and statistics (e.g., click through rate and conversion rate) of a product. The user feature group includes a user’s profile (e.g., gender, age and city) and some statistical information such as purchase level and active level.

4 EXPERIMENTS

We have collected 7-day customer behavior sample data that includes more than 10 million user clicks from Taobao.com. The total number of customers is 24,634 and the average user clicks and buy is around 442.7 and 4.3. For each user in the dataset, almost all types of behaviors (e.g., click, add-to-cart, buy, and look-at-comments) are collected. Besides, user’s profile (e.g., gender, age, and city) and item’s attribute information (e.g., category, seller, brand, etc) are also collected. The records of off-the-shelf items without any attributes and users with less than 10 behavior records are excluded. We split the dataset into three non-overlapping sets: the training set, the validating set, and the test set. The training set contains the 1th day to 6th day’s data of 90% users randomly sampled. The test set contains the 7th day’s data of the above 90% users. The validation set contains the whole 7-day period data of the rest 10% users.

For the task of direct buying conversion rate prediction, we evaluate our model by comparing it with several state-of-the-art recommendation approaches:

- **BPR-MF**: Bayesian Personalized Ranking based Matrix Factorization [7] is a collaborative filtering method based on users’ pairwise preferences. In our experiments, we specify a user’s pairwise preference by labeling items bought as positive. We also discretize all statistic features mentioned in Section 3 as the global features used in BPR-MF.

- **GBDT**: Gradient Boosted Decision Tree [2] is a widely used nonlinear learning algorithm with strong generalization ability. All features are transformed into continuous values in our experiment. GBDT cannot handle too many features, thus ItemId and categoryId features are not included.

- **DNN**: Deep Neural Network model doesn’t have inside memory so it is hard to capture temporal information. So it is also included as a baseline method to help us figure out whether modeling decision process by LSTMs really helps. It has exactly same number of layers and parameters as our LSTM model.

4.1 Results and Discussion

**Comparison with Baselines** Table 1 shows the final results obtained from different approaches on the testing set. +behavior_onehot / +behavior_emb means adding one-hot/embedding behavior features. +itemid_emb / +catId_emb means adding item-IId/categoryId embedding. +multi means multi-task learning. To be fair, we carefully tune the parameters of each model with a grid search method to ensure that each model achieves its best performance on the validation set. Table 1 demonstrates that the proposed approach can significantly improve the performance compared with all baseline approaches. Our model +catId_emb + behavior_emb + multi
achieves the best performance on all metrics except a slight decrease P@5 at 0.03%. It achieves a 16.5% relative improvement of AUC over BPR-MF, 13.3% over GDBT with behavior features, and 3.7% over DNN model with catId and behavior embeddings. The results are not surprising, as it is hard for BPR-MF and GDBT to capture the characters of various behaviors and their temporal relations. It’s also hard for DNN models to capture the transition of user buying decision stages without the inside memory. Even without behavior features, the performance of our approach is much better than that of the GBDT and DNN.

We also study the effect of different features for both DNN and our models. Table 1 shows the introducing of the category ID embedding features can achieve a better performance than that of item ID. A possible reason might be that item ID is too sparse and hard to generalize, while category ID is more general since the model complexity is much lower compared with using item ID, meanwhile offer enough discriminative power. After introducing the behavior embedding features, both our approach and DNN improve significantly. This indicates that explicitly introducing user behaviors is valuable for predicting the buying stages. Introducing multi-task learning achieves a further improvement of 1.1%, 2.8%, 3.7% and 2.7% in terms of AUC, MAP@5, MAP@10 and MAP@20, though it causes a slight decrease of 0.15% in P@5. The result demonstrates that the other three tasks not only estimate the probability of a user being at need-recognition, consideration and feedback stage, but also regularize the buying stage prediction model.

Comparison across Number of Target Classes As mentioned in Section 2.1, the main task can be extended to predict the probability of three classes (direct buy, will buy and other). With 2-class setting, the performance is 0.8340, 0.1865, 0.4497 in terms of AUC, P@5, MAP@10, and with 3-class setting, the performance is 0.8496, 0.1955, 0.5002. After introducing the will buy target class, the performance increases 1.9%, 4.8% and 11% in terms of AUC, P@5, and MAP@10. In addition, the convergence speed of the three classes setting is much faster than that of the two class setting (shown in Figure 3). According to common practice in industry, will buy is also an important target that reflects the delayed reward in future. More target classes mean higher positive label density, which makes the training data easier to learn, which in return might leads to a better performance. Therefore, we adopt the three classes setting as the default in all experiments.

5 CONCLUSION

Consumer buying decision process is a widely recognized concept in marketing research, and unfortunately it is ignored by the recommendation systems community. In this paper, we explicitly model the consumer’s buying decision process to help the predicting of the conversion rate in e-commerce recommendation system. With the introduced LSTM modeling approach, we demonstrated how recommendation systems can benefit from this concept.

This is a first step towards introducing consumer buying decision process modeling into recommendation systems. In particular, we have made strong assumptions and used rules to specify which purchasing decision stage the user is at based on user click behaviors. We can introduce the stages as hidden variables in the future, which can also be modeled with the deep learning framework.

ACKNOWLEDGMENTS

The authors would also like to thank the anonymous referees for their valuable comments and helpful suggestions. This paper is supported by NSFC project 61772040 and 61751201. The corresponding author is Zhifang Sui.

REFERENCES