Multi-task based Sales Predictions for Online Promotions

SHEN Xin*
Zhejiang University
Hangzhou, Zhejiang, China
sxstar@zju.edu.cn

Martin Ester
Simon Fraser University
Burnaby, B.C., Canada
ester@sfu.ca

Jiajun Bu
Zhejiang University
Hangzhou, Zhejiang, China
bjj@zju.edu.cn

Chengwei Yao
Zhejiang University
Hangzhou, Zhejiang, China
yaochw@zju.edu.cn

Zhao Li
Alibaba Group
Hangzhou, Zhejiang, China
lizhao.lz@alibaba-inc.com

Xun Zhou
Zhejiang University
Hangzhou, Zhejiang, China
21821201@zju.edu.cn

Yizhou Ye
Alibaba Group
Hangzhou, Zhejiang, China
yizhou.yyz@alibaba-inc.com

Can Wang
Zhejiang University
Hangzhou, Zhejiang, China
wcan@zju.edu.cn

ABSTRACT

The e-commerce era is witnessing a rapid development of various annual online promotions, such as Black Friday, Cyber Monday, and Alibaba's 11.11, etc. **S**ales **P**redictions for **O**nline **P**romotions (SPOP) are a set of sales related forecasts for the promotion day, including gross merchandise volume, sales volume, best selling products, etc. SPOP is highly important for e-commerce platforms to efficiently organize merchandise and maximize business values. However, sales patterns during the promotions are varied according to different scenarios, each model of which is designed with different features, static or dynamic, for one task in particular. Therefore, several models are proposed with part of features that are possibly beneficial to other tasks, which indicates the universal representation for the items needs to be learned across different promotion scenarios.

To address this problem, this paper proposes a **D**eep **I**tem **N**etwork for **O**nline **P**romotions (DINOP). In DINOP, we design a novel **T**arget **U**sers **C**ontrolled **G**ated **R**ecurrent **U**nit (TUC-GRU) structure for dynamic features, and provide a new attention mechanism introducing static users profiles. In contrast to traditional prediction models, the network we proposed can effectively and efficiently learn universal item representation by incorporating users' properties as controllers. Furthermore, it can successfully discover the static and dynamic features guided by the multi-task learning, and is easily extended to other sales related prediction problems without retraining. Empirical results show that performance of DINOP in the real data set of Alibaba's Global Shopping Festival is superior to other state-of-the-arts practical methodologies in terms of the convergence rate and prediction accuracy.

CCS CONCEPTS

• **Information systems → Electronic commerce**; • **Computing methodologies → Multi-task learning**; • **Applied computing → Marketing**.

KEYWORDS

Multi-task learning, Sales prediction, Online promotion, Representation learning

ACM Reference Format:
SHEN Xin, Martin Ester, Jiajun Bu, Chengwei Yao, Zhao Li, Xun Zhou, Yizhou Ye, and Can Wang. 2019. Multi-task based Sales Predictions for Online Promotions. In *The 28th ACM International Conference on Information and Knowledge Management (CIKM '19), November 3–7, 2019, Beijing, China.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3357384.3357823

1 INTRODUCTION

Alibaba's 11.11 Global Shopping Festival, hosted annually on 11 November, is one of the world's largest 24-hour online shopping event. This event began in 2009, generating $7.6 million sales in that year. It has evolved from a 24-hour sales event to a 23-day global shopping and entertainment festival, generating $30.8 billion **G**ross **M**erchandise **V**olume (GMV) in 2018.

Various interactive promotion events are held during the 22-day pre-sales period. Alibaba and its brand partners encourage consumers to add merchandise to their shopping carts before 11 November. Meanwhile, shoppers are able to get a discount if they place an order and make a deposit during the pre-sales period. Hidden in the massive amount of user-item interactions during this period are valuable users and items characteristics, which affect transactions on 11 November directly.

Multiple sales predictions tasks, including GMV, **S**ales **V**olume (SV), **B**est **S**elling **P**roducts (BSP), and **S**ale **S**lot (SS) forecasts are significant to both Alibaba and sellers. The GMV is a term used in online retail to indicate a total sales dollar value for merchandise sold through a particular marketplace over a certain time frame.

*SHEN Xin would like to capitalize all letters of his family name.

CIKM '19, November 3–7, 2019, Beijing, China

ACM ISBN 978-1-4503-6976-3/19/11…$15.00
https://doi.org/10.1145/3357384.3357823

It is one of the most important evaluation metrics for the Global Shopping Festival. According to the GMV forecast, Alibaba could adjust promotion events during the festival. The SV shows the number of one specific item which will be sold on 11 November. Based on the SV prediction, sellers are able to optimize inventories. The BSP suggests the popularity of items, and the SS implies the ranking of each item's market share. In addition, we can introduce user features, or activity features to predict whether a user will buy an item, or whether an item should be participated in an activity during the festival. In terms of above forecasts, Alibaba could optimize activities and improve personalization recommendations in the online promotion.

At the first glance, it seems that the problem of **S**ales **P**redictions for **O**nline **P**romotions (SPOP) can be solved by simply training models for multiple years using sales data from historical promotion days. Plenty of existing recommendation methods in e-commerce, like regression, classification, clustering, time series prediction, etc., can be applied to each task respectively. More recently, **D**eep **N**eural **N**etworks (DNNs) and **R**ecurrent **N**eural **N**etworks (RNNs) are developed for a better performance. In particular, the **L**ong **S**hort-**T**erm **M**emory (LSTM) [23] and **G**ated **R**ecurrent **U**nit (GRU) [13] are very competitive in time series predictions.

However, the SPOP problem differs from existing problems in the following important aspects:

(1) We can't utilize historical data in the promotions of previous years to train a model for two main reasons: one is the lack of data, the other is the variability of online promotions. Most new e-commerce companies have insufficient data from promotions because of the cold-start problem. Even Alibaba, which organizes the largest online shopping event for 10 years, doesn't have enough historical data because of the database storage limits. For example, the data table containing the user-item interactions on one day is nearly 300TB. It is infeasible for a company to preserve all data forever. The life circle of most data tables in Alibaba is no more than 2 years. In addition, customer structure and preferences change every year along with promotion themes, therefore forecast accuracy would be compromised even if we had historical data from peak sales days.
(2) Using only parts of static features, dynamic features, or a single day's features according to the items' profiles and sales won't work in SPOP. The transaction volume on a promotion day could be 20 times higher than on a normal day. Traditional machine learning methods, like regression and classification, can't achieve good results when using only static data or a single day's features. Meanwhile, only applying dynamic data during the pre-sales period in time series predictions won't work either. According to our statistics, more than 50% transactions on the promotion day occur without relevant pre-orders during the pre-sales period.
(3) Single task learning is not a good option for solving SPOP. In the e-commerce industry, various sales relevant requirements can be proposed anytime, especially during online promotions. A lot of time and resources could be wasted in designing and training models for numerous new tasks. Learning universal and transferable item representation by multi-task is valuable, especially for some urgent or real-time demands.
(4) Items sales features and target users features can't be considered equally. Although features of target users impact transactions, they are indirect factors; whereas item features in sales related tasks impact transactions more directly. We have daily data on these two kinds of features during the pre-sales stage, however it is not wise to feed both of them into an RNN network. Because widely used RNN structures, like LSTM and GRU, treat every input fairly, i.e. all inputs affect the gates and hidden states. Thus, a new RNN structure is required to handle heterogeneous inputs differently.

Based on these characteristics, we aim to learn a universal and transferable sales related item representation which depicts states of one item during the online promotions from multiple tasks. Due to the limit of database storage, we can only utilize data in 2017 to train the model, and data in 2018 to test. To process heterogeneous features, we propose the **D**eep **I**tem **N**etwork for **O**nline **P**romotions (DINOP) which integrates the data of both static features and dynamic features in one structure. In addition, to treat items sales features and target users features differently, we design a novel RNN cell with a new attention network by incorporating target users features as the controllers rather than normal inputs. We implement and evaluate our models on a real-world data set, Alibaba's 11.11 Global Shopping Festival. Experimental results show that our method outperforms all compared algorithms, and the sales related item representation learned from multi-task is more practical than those learned from single tasks. Furthermore, one new sales relevant task is introduced to demonstrate that our item representation can be transferred to a new problem effectively and efficiently.

We summarize major contributions of this paper as follows:

(1) We present the first study on the **S**ales **P**rediction for **O**nline **P**romotion (SPOP), the problem of multiple tasks of sales related forecast utilizing heterogeneous inputs, including static features, dynamic features, item features and target users features. SPOP exists widely in the e-commerce industry.
(2) We propose a **D**eep **I**tem **N**etwork for **O**nline **P**romotions (DINOP), a general transferable item representation learning method based on multi-task learning. DINOP discovers the knowledge from both static properties and dynamic time series.
(3) We design a novel RNN based deep architecture, the **T**arget **U**sers **C**ontrolled **G**ated **R**ecurrent **U**nit (TUC-GRU), with a new attention mechanism, for modeling items based on the dynamic features. Different from existing methods, the universal item representation incorporates the target users' properties as the controllers in TUC-GRU.
(4) We show results of our experimental evaluation on a real-world industry data set, Alibaba's 11.11 Global Shopping Festival, which demonstrate the superiority and transferability of our methods compared to existing methods.

The rest of the paper is organized as follows. In Section 2, we give a brief review of related work. Section 3 shows the problem definition of the SPOP. The DINOP structure is described in Section 4. Section 5 presents the experimental methodology, results and analysis. Finally, we summarize the paper and suggest directions for future work in Section 6.

2 RELATED WORKS

2.1 Recommendation systems

There are increasing number of research on the recommendation during the e-commerce era. Typically, recommendation system works in two main ways, content-based and collaborative filtering (CF). The content-based approaches use description of items and profiles of users' preferences [4]. The CF predicts the ratings of items to a particular user based on the user-item interaction history [8]. Nowadays, CF is out of question the most widely adopted recommendation approach [1]. Most of these studies focus on users' interests and behaviors, such as matrix factorization with additional information inferring user preferences by observing user behaviors [26], modeling user preferences for point-of-interest recommendation [28], a latent factor model with regression prior to predict the response [2], grouping mobile access records for characterizing user behavioral patterns [39], and a user-item co-clustering framework to improve CF [9]. However, there are still some works on the item feature-based matrix factorization. For instance, a model based on Latent Dirichlet Allocation is introduced to predict dyadic response by bag-of-words like representation [3]. Recently, DNNs are widely used to improve recommendation systems. Restricted Boltzmann Machine first models the user-item interactions with neural network for CF [36]. Besides, auto-encoders are also introduced to CF by learning two fully connected layers [37]. It is a remarkable fact that a lot of Internet-related technology companies have participated in the researching on and implementing DNNs on recommendations. Google proposed a wide and deep network to memorize sparse feature interactions and generalize to previously unseen feature interactions [12]. LinkedIn unitized the modified wide and deep network for a job redistribution model [6]. Microsoft presented a series of latent semantic models with deep structures for document ranking [24]. Youtube described a deep neural network for recommending videos in candidate generation and ranking [15].

2.2 Time series prediction

Time series forecasting can be done by classical statistical techniques, such as autoregression (AR), moving average (MA), autoregressive moving average (ARMA) and autoregressive integrated moving average (ARIMA) [7], and generalized autoregressive conditional heteroskedastic (GARCH) [5]. However, these models predict the value changes from the sequence itself without considering the impact from other factors. Because of the nonlinear modeling capabilities, a lot of machine learning methods are proposed to solve time serial prediction. For instance, artificial neural network [34], and support vector machine [38]. Deep learning, especially RNN model, is also applied for improving the prediction using sequential characteristics [27]. For example, we can forecast the sales of a supermarket using a long shot-term memory network [30], and infer user's future intents according to their previous behavior sequences [22].

2.3 Multi-task representation learning

Multi-task learning is a solution when there are multiple related tasks that have limited training samples [10]. Single-task representation learning only obtains features associated with one task, so plenty of information could be discarded. On the contrary, multi-task representation learning allows knowledge sharing and transferring. The representation obtained by a multi-task learning contains more underlying information. A lot of work has confirmed the effectiveness of multi-task representation learning. Multi-task learning has been widely and successfully used across various applications of machine learning [35], such as natural language processing [14], speech recognition [16], and computer vision [19]. Recently, plenty of technology companies became interested in multi-task representation learning. Google developed a multilingual neural machine translation system to enable zero-shot translation [25]; and adapted multi-gate mixture-of-experts structure to multi-task learning by sharing the expert submodels for the content recommendation system [31]. Microsoft proposed a multi-task deep neural network for representation learning, in particular focusing on semantic classification and semantic information retrieval [29]. Alibaba utilized multi-task learning to obtain users representation based on user behavior sequences [33].

To the best of our knowledge, our work is the first one that utilizes multi-task learning to capture universal item representation for online promotion predictions.

3 THE SALES PREDICTIONS FOR ONLINE PROMOTIONS PROBLEM

Online promotions widely exist in e-commerce. Black Friday and Cyber Monday have been famous in the United States since 2005. Alibaba's 11.11 Global Shopping Festival began in 2009, and generated US$30.8 billion in GMV in 2018.

The objective of the SPOP problem is to accurately predict multiple sales related tasks, aiming to provide guidance for both e-commerce platform and sellers. Different from most current problems, such as personalization search, recommendation, and advertising, sales related tasks are particularly item-based.

However, heterogeneous factors contribute to the transactions during the online promotion. For instance:

1. **S**tatic **I**tem **I**nherent profiles (*SII*), which indicate the features of the item, including item id, seller id, brand, category, seller address, etc.
2. **S**tatic **I**tem **S**ales data (*SIS*), which represent usual sales situations of the item, including GMV, SV, price, etc.
3. **S**tatic target **U**sers profiles (*SU*), which suggest the distribution of features of users who have bought the item during a long-term time, including age, gender, horoscope, etc.
4. **D**ynamic **I**tem **S**ales data during pre-sales period (DIS_t), which show the daily pre-sales situations of the item on the t-th pre-sales day.
5. **D**ynamic target **U**sers profiles during pre-sales period (DU_t), which imply the daily distribution of features of users who buy the item on the t-th pre-sales day.

The first three types of factors are regarded as static features, which are stable and represent the normal states, while the last two types of factors are regarded as dynamic features which denote the daily pre-sales states. In our experiments, we set the interval as 90 days for static features, and set the time series data for pre-sales as 20 days, between 20 October and 8 November.

Meanwhile, the e-commerce industry also faces multiple real-world sales related problems, as follows:

(1) The GMV task, a regression problem, which predicts the GMV of one item on 11 November. According to the results of this task, we can obtain multiple granularities of GMVs, such as shop GMV, industry GMV, and all-platform GMV. In addition, we can achieve the SV of each item based on the GMV forecast to instruct the seller to adjust the stocking volume before the online promotion day.

(2) The BSP task, a binary classification problem, which judges whether one item belongs to the list of best selling products. Specifically, we define the BSP as follows:

$$y_{BSP_i} = \begin{cases} 1 & \text{if } \sum_{j \in \{j|G_j \geq G_i\}} G_j/G_T < 0.2, \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

where G_i indicates the GMV of $item_i$, and G_T indicates the all-platform GMV.

(3) The SS task, a multiclass classification, obtains an in-depth index of the BSP for each industry instead of all-platform. In particular:

$$y_{SS_i} = R \in \{1, 2, ..., 10\} \tag{2}$$

$s.t.$

$$\frac{R-1}{10} < \sum_{i,j \in \text{industry}_k, j \in \{j|G_j \geq G_i\}} G_j/G_{T_k} \leq \frac{R}{10}$$

(4) The **B**uying through **O**nline **P**romotion (BOP) task, a special personlization recommendation problem, predicts whether one user will purchase one item during the promotion.

(5) The **J**oining an **A**ctivity (JA) task, a binary classification problem, which decides whether one item should be included in a promotional activity.

To tackle the highly challenging problem of SPOP, we propose the DINOP. Both static features and dynamic features are utilized to generate the presentation for items through multi-task learning. Both static target users profiles and dynamic target users profiles are introduced as the controllers instead of ordinary inputs in the novel TUC-GRU cell and the attention network. The universal sales related item representation contains more features extracted from multiple tasks, and enable the possibility of transferring to new related tasks.

4 DEEP ITEM NETWORK FOR ONLINE PROMOTIONS

Figure 1 shows the general overview of the DINOP. In terms of various types of features, we design two key components: a TUC-GRU network and an attention based pooling. The novel RNN network takes dynamic features as inputs and helps to model the daily information during the pre-sales period. The new attention based pooling integrates not only the RNN's outputs by learning different weights controlled by target users profiles, but also the static features. By sharing item representation among multiple tasks, our model can achieve a better performance and be transferred to other relevant tasks.

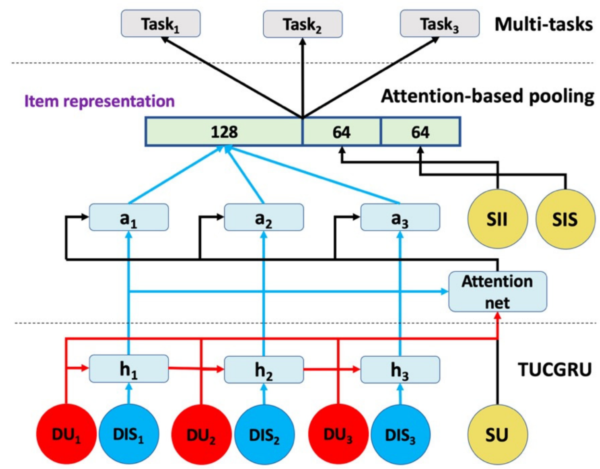


Figure 1: General model architecture of the DINOP. Circle nodes indicate the inputs. Yellow nodes indicate the static features. Red lines denote the influence from dynamic target users features, and blue lines reveal how daily sales data contribute to sales related item representation.

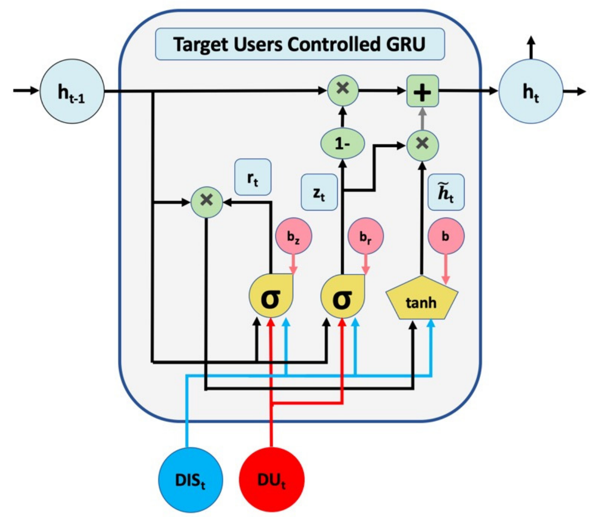


Figure 2: The TUC-GRU cell. Blue lines show dynamic item sales data as the normal inputs of TUC-GRU, and red lines indicate dynamic target users properties as controllers.

4.1 Target users controlled GRU

Since pre-sales stage plays an important role in online promotions, one component of the encoder is designed to process the daily time series data. Two kinds of the dynamic features, DIS_t and DU_t

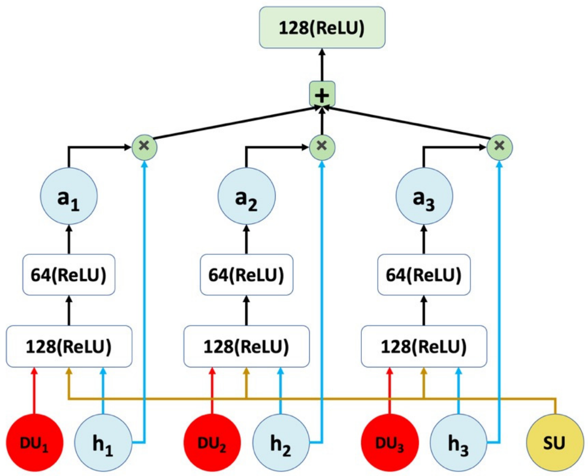


Figure 3: The attention net on the top of the TUC-GRU. The h_t indicates the sales related item representation of the t-th day of the pre-sales stage. Both static and dynamic target users properties are introduced to learn the weights of h_t.

during the pre-sales period are fed to an RNN network. We propose the TUC-GRU to treat these two features differently. In particular, DIS_t directly reflects the sales states, so we treat it as an ordinary input of the TUC-GRU cell. In contrast, DU_t implies the features of target users on one day instead of any sales-related features, so it is regarded as a strong control signal in the gates of TUC-GRU. We formulate the TUC-GRU model as follows:

$$\begin{aligned} z_t &= \sigma(w_{DIS,z}DIS_t + w_{DU,z}DU_t + w_{hz}h_{t-1} + b_z) \\ r_t &= \sigma(w_{DIS,r}DIS_t + w_{DU,r}DU_t + w_{hr}h_{t-1} + b_r) \\ \tilde{h_t} &= \tanh(w_h r_t h_{t-1} + w_{DIS}DIS_t + b) \\ h_t &= (1 - z_t)h_{t-1} + z_t\tilde{h_t} \end{aligned} \tag{3}$$

where z_t and r_t indicate the *update* and *reset* gates of the t-th object respectively, $\tilde{h_t}$ is the cell activation vector.

Figure 2 shows the cell structure of TUC-GRU. The output of TUC-GRU is a sequence $h = h_t$, containing daily sales related item representation during the pre-sales period.

4.2 Attention based pooling

It is a fact that the daily target users are easily affected by various promotional activities. The target users properties on a single day may far differ from the normal case. So we apply a new attention mechanism on the top of the TUC-GRU by introducing static target users profiles, SU. As the output of TUC-GRU, we consider each h_t as a part of item representation. The attention weights assign proper credit to daily sales states according to the static target users features. The item representation generated from the daily features, rep_d, takes the form as follows:

$$\begin{aligned} rep_d &= \sum a_t h_t \\ a_t &= \frac{\exp\left(attention(h_t, DU_t, SU)\right)}{\sum_{t=1}^{T}\exp\left(attention(h_t, DU_t, SU)\right)} \end{aligned} \tag{4}$$

where a_t is the weight for the hidden state h_t, T is the number of days of pre-sales, *attention*() is shown in Figure 3. Both static and dynamic target users information contribute to the attention weights.

The final item representation rep, a 256 dimensional vector, is the concatenation of rep_d, emb_{SII}, and emb_{SIS}, where emb_{SII} and emb_{SIS} are both 64 dimensional vectors embedded from the *SII* and *SIS*, respectively.

4.3 Multiple tasks

Mentioned in Section 3, the SPOP has at least five practical industry prediction tasks, which are GMV, BSP, SS, BOP, and JA. We aim to learn the universal item representation for online promotions, so the general model trains these tasks simultaneously after obtaining the item representation from attention-based learning, as shown in Figure 1. For each task, others are regarded as regularizations. We give the net structures for these five tasks in Figure 4.

The GMV task is a regression problem directly connected with item representation, because both pre-sales data and normal sales data are the inputs of the DINOP. Mean Square Error (MSE) is used as the loss function:

$$Loss_{GMV} = \frac{1}{N}\sum_{i=1}^{N}(y_i - f_{GMV}(rep_i))^2 \tag{5}$$

where rep_i indicates the representation of $item_i$ outputted from the attention-based pooling, y_i is the label, N is the number of items, $f_{GMV}(item_i)$ is a function learning a real number from the $item_i$ shown in Figure 4(a).

The BSP task is a binary classification problem also directly associated with item representation. We define the loss function as follows:

$$\begin{aligned} Loss_{BSP} = -\frac{1}{N}\sum_{i=1}^{N}&[y_i \log(f_{BSP}(rep_i)) \\ &+ (1 - y_i)\log(1 - f_{BSP}(rep_i))] \end{aligned} \tag{6}$$

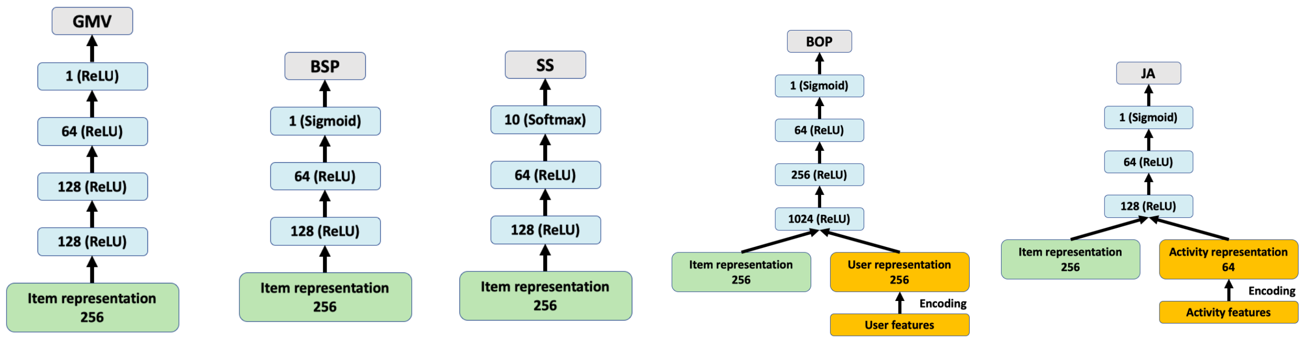
where $f_{BSP}(rep_i)$ is a function shown in Figure 4(b) mapping rep_i to a real score $\in (0, 1)$, indicating the probability whether $item_i$ belongs to the best selling product list.

The SS task is a 10-class classification problem. Each merchandise item is classified into one slot based on its rank of GMV among other similar items. There are 10 slots in total and the first slot indicates the best popularity. Empirically, results of this task follow long tail distribution. We define the loss function as:

$$\begin{aligned} L_{SS} &= -\sum_{i=1}^{N}\log(\sigma_{y_i}(f_{SS}(res_i))) \\ s.t. \sigma_j(f_{SS}(res_i)) &= \frac{\exp(f_{SS}(res_i, j))}{\sum_{k=1}^{10}\exp(f_{SS}(res_i, k))} \end{aligned} \tag{7}$$

where $y_i \in \{1, 2, ..., 10\}$ is the label of $item_i$, $f_{SS}(res_i, j)$ indicates the prediction of the probability that $item_i$ belongs to the j-th slot. Figure 4(c) presents the net structure for SS task.

The BOP task is a special personlization recommendation problem, which can be regarded as a binary classification. This task is aimed to forecast whether one user will buy one item during the promotion. To address this issue, we design a network shown in



(a) GMV (b) BSP (c) SS (d) BOP (e) JA

Figure 4: Net architectures for multiple tasks using the universal item representation

Table 1: Statistics of inputs

Factors	Raw features	Dimension after encoding
SII	84	128
SIS	99	319
SU	103	256
DIS	99	128
DU	103	256
ACT	269	64

Figure 4(d) introducing both item representation and features of users. The loss function is similar to the BSP task.

The JA task is also treated as a binary classification problem. There are various activities in the e-commerce industry, for example, "lightning deals" in Amazon, "juhuasuan" in Taobao. This task predicts whether one item should be included in one activity. And it takes item representation and features of **ACT**ivities (*ACT*) as inputs. The net structure is shown in Figure 4(e).

5 EXPERIMENTS AND RESULTS

We implement and evaluate our DINOP on a real-world data set of Alibaba, a world-leading e-commerce company. In this section, we first describe the data set and experimental configurations. Then we demonstrate that the proposed DINOP achieves superior performance over some comparable and practical methods. In addition, we compare multi-task learning with single task to validate the performance of the proposed method in learning representation. Finally, we verify the transferability of the item representation trained by the multi-task.

5.1 Data set and experimental settings

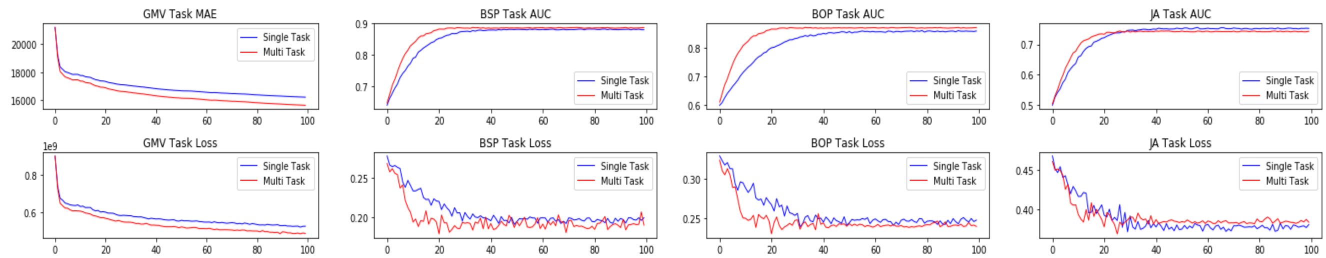
5.1.1 Data description. Our experiments are performed on a large-scale industry data set with the real logs of the world's largest online promotion, Alibaba's 11.11 Global Shopping Festival. The most recent data are used in our experiments, i.e. we train our model on the 2017's 11.11 Festival, and test on the 2018's. Fortunately, pre-sales stage began on 20 October, the same day for both years. So these two time series just match naturally. As mentioned in Section 3, the SPOP's inputs contain two main types of data: the static features reflecting the inherent profiles or the stable states for a long period before the promotions, and the dynamic features indicating the daily information during the pre-sales period. For static features, we use the 90 days information before the pre-sales, i.e. 22 July - 19 October. For example, the price of an item, one static feature, sometimes changes according to the market. We treat the average price during the 90 days as one feature of *SIS*. For dynamic features, we use the daily data from the beginning of the pre-sales to three days before 11 November, i.e. 20 October - 8 November, 20 days totally. Raw data are far from perfect for experiments, so we remove inactive users and items. Most features are categorical values or string values, especially in *SII*. We empirically select some relevant features, adopt one-hot encoding [21] to convert them to binary features, multiple correspondence analysis [20] is utilized to reduce the dimensions. Table 1 shows the statistics of all inputs.

5.1.2 Experimental settings. We deploy our algorithm on Alibaba's server clusters comprising of 800 computing nodes with 10 GPUs. During pre-sales period, our algorithm continually updates the predictions for GMV, BSP and SS every day, which are important guidance for both Alibaba and sellers. In this paper, we apply a fixed data set mentioned above for offline comparisons. L2-norm is used for regularization. The network is trained by Adagrad [17]. Empirically, we have tried the dropout as 0.3, 0.5, and 0.8, the learning rate as 10^{-1}, 10^{-3}, and 10^{-6}. According to the results, we finally set the dropout and learning rate as 0.5, and 10^{-3} in our experiments, respectively.

5.2 Performance comparison for single task

First, we compare our DINOP with several practical and state-of -the-art methods to reveal the effectiveness of our model. To the best of our knowledge, no existing models can deal the SPOP through a universal structure. So, multiple tasks are learned separately in this part. We start with a description of the compared algorithms:

(1) DNN: The fundamental network, a feed forward network composed of fully connected layers, is treated as our baseline.



(a) GMV (b) BSP (c) BOP (d) JA

Figure 5: Comparisons of single task and multi-task on the GMV, BSP, BOP, and JA task. X-coordinate stands for the learning interval of an epoch. Y-coordinate stands for the evaluation metrics and the loss. The evaluation metrics are MAE for the GMV task, and AUC for the other three tasks.

Table 2: Comparisons of different algorithms

	GMV MAE	BSP AUC	SS ACC	BOP AUC	JA AUC
DNN	20375	0.7217	0.5912	0.8139	0.7184
Wide&Deep	20098	0.7924	0.6026	0.8243	0.7315
LSTM	16995	0.8617	0.6247	0.8313	0.7324
GBDT	28186	0.8774	0.6657	0.8351	0.7398
XGBoost	23960	0.8611	0.6472	0.8437	0.7412
DINOP-su	16876	0.8765	0.6963	0.8413	0.7432
DINOP-du	16962	0.8788	0.6862	0.8517	0.7396
DINOP-u	17002	0.8697	0.6548	0.8276	0.7317
DINOP	**16219**	**0.8811**	**0.7140**	**0.8578**	**0.7513**

Adagrad is used as the optimizer. It has three hidden layers with the sizes of 1024, 512, 256.

(2) Wide & Deep [12]: The combination of a logistic regression model and a DNN model maintains the benefits of memorization and generation. Adagrad and follow-the-regularized-leader (FTRL) [32] are used as optimizers for deep part and wide part respectively. To keep consistency, the deep part is the same as DNN.
(3) LSTM [23]: The network is well-suited for classifying, processing and making predictions based on time series data, since there can be lags of unknown duration between important events in a time series.
(4) Gradient Boosting Decision Tree (GBDT) [18]: As a machine learning method for regression and classification problems, it produces a prediction model in the form of an ensemble of weak prediction models. It is widely used in industry applications.
(5) XGBoost [11]: A solution that combines boosted trees with conditional random field. It is also widely utilized in industry areas and data mining competitions like KDDCup.
(6) DINOP-su/DINOP-du/DINOP-u: Three submodels of DINOP ignore the static target users profiles, dynamic daily target users profiles, and both of two, respectively.

We evaluate the performance of each model on five tasks separately, i.e. the GMV, BSP, SS, BOP and JA task. The GMV is a regression task, the SS is a multiclass classification task, and others are all binary classification tasks. We apply mean absolute error (MAE), accuracy (ACC), and area under curve (AUC) as the evaluation metrics, respectively. Table 2 summarizes the experimental results for the DINOP and the compared models. The following conclusions can be drawn from the experimental results:

(1) The DINOP network proposed in this paper consistently outperforms all compared methods for all five tasks. It is capable of all the forecasts because of the TUC-GRU structure for time series data and the introduction of the target users profiles as control signals in two components.
(2) The first three rows are all neural network based methods. Wide & Deep performs better than DNN since it has an extra logistic regression model. LSTM is much better than others because it is designed to deal with the time series data, which are dynamic features in SPOP.
(3) There is no doubt that GBDT and XGBoost are more competitive since they are widely used in the industry. Even though they achieve good results for all classification tasks, they fail in the GMV task. Different from other four tasks predicting relative values, the GMV task forecasts absolute values. Features contributing to the absolute GMV prediction are mostly embedded in the time series features, which GBDT and XGBoost are not able to deal with. It is not surprising that these two practical methods fail in this task.
(4) The results of DINOP's submodels demonstrate the effectiveness of introducing the target users profiles as a strong signal to control both the TUC-GRU and the attention. It reflects the fact that target users contribute to the sales predictions.
(5) LSTM appears to be the best competitor, since it is designed to process time series data. However, the results show that LSTM doesn't perform well in SS task and BOP task compared with GBDT, XGBoost, and our method. The reason is that LSTM treat both *DIS* and *DU* as inputs. The representation it learned contains both features of *DIS* and *DU*. In contrast, our method treats only *DIS* as input and introduces *DU* as a signal to obtain a better representation for *DIS*. It is

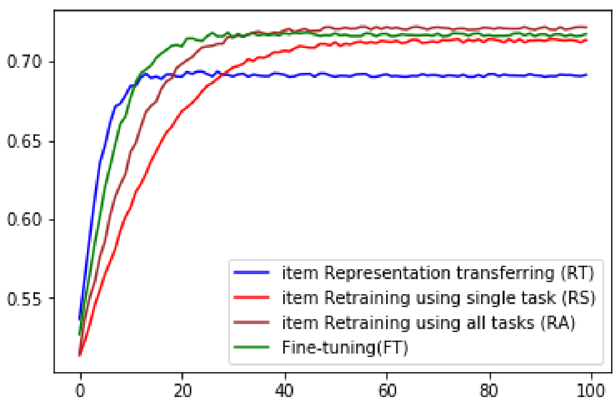


Figure 6: Comparison of 4 methods of training a new related task, the SS task. X-coordinate stands for the learning interval of an epoch. Y-coordinate stands for the evaluation metric of the SS task, ACC.

an evidence showing that using target users profiles as controllers, instead of ordinary inputs, can improve the training results for the item representation.

5.3 Evaluation in multi-task learning

In this section, we compare multi-task learning with single task on the SPOP. By sharing features and representation among relevant tasks, DINOP can generalize better for muliple tasks. We implement a multi-task learning with four tasks, which are the GMV, BSP, BOP and JA.

As shown in Figure 5, each learning process can be divided into two parts. In the first stage, the loss decreases rapidly. On the contrary, the loss changes slowly during the second stage. The reason is that some frequent features can be easily learned by the network, while the sparse features are difficult to be perceived.

Our experimental results show that multi-task learning converges faster than single task learning for any of the four tasks. Multi-task learning also reduces the length of the first stage of learning. Compared with the single task, the predictions can be improved in the GMV, BSP and BOP forecasts by sharing representation and features with other tasks. Different from other three sales related tasks which are directly affected by transactions, JA task is more relevant to the item profiles. It is the main reason why multi-task fails in JA task compared with the single task.

Moreover, in industry, one general model for multiple tasks is more economical than several task-specific models.

5.4 Representation transferability

DINOP generates universal item representation about the online promotions. In Section 5.3, we obtain the item representation through learning from four tasks: the GMV, BSP, BOP, and JA. If we want to further predict the SS, a new related task, we have four methods as follows:

(1) **R**epresentation **T**ransferring (RT): The new task net treats obtained item representation as inputs, and trains itself without any connections with the origin model.
(2) **R**etraining using **S**ingle task (RS): Ignore the existing results, retrain the new task with DINOP as a single task.
(3) **R**etraining using **A**ll tasks (RA): Connect the task net to the DINOP, ignore the achieved results, and retrain the whole network together with all other tasks.
(4) **F**ine-**T**uning (FT): Add on the new task's net to the item representation layer of DINOP. The current learned network and item representation are used as the initialization for the modified structure. And the whole network is kept on training for multiple tasks including the new one.

We implement all four methods above to train a new related task, the SS. We draw the conclusions from the experimental results shown in Figure 6 as follows:

(1) All the RS, RA and FT methods achieve similar forecast abilities for the SS. Using ACC as the evaluation metric, RA outperforms FT by 0.17%, while FT performs better than RS by 0.24%. The differences between these three methods are not significant. However, the convergent speeds are different from each other. Avoiding retraining the whole network, there is no doubt that FT converges faster than RS and RA. Thus, FT is a better solution to transfer our well-trained model to new relevant tasks.
(2) RT method proves itself as a good algorithm on some specific occasions. RT is only 3.19% worse than RS in ACC. Because RT just utilizes the current learned item representation instead of training or retraining the whole network, undoubtedly RT has the highest rate of convergence among all four methods. Without integrating various types of inputs and complex network structures, RT can be easily deployed in the system. The simplicity and efficiency make RT welcomed in a lot of industry scenes, especially in the online system.

The good performance of both FT and RT verifies the transferability of our item representation. After balancing the effectiveness and efficiency, it is simple to choose a proper method to transfer our DINOP to a new related task, without any kind of retraining.

6 CONCLUSIONS

In this paper, we introduce the SPOP problem and propose a novel deep neural network, called DINOP, for tackling the multi-task SPOP problem by learning a universal item representation. During the pre-sales preiod, we train an RNN-based architecture to discover dynamic features of the items. In order to consider the influence on the item sales by the target users, we design a novel TUC-GRU cell and a new attention mechanism that considers the target users profiles as controllers. Extensive experimental results on the large-scale data set of the world's largest online promotion, Alibaba's 11.11 Global Shopping Festival, validate the effectiveness, efficiency, and transferability of DINOP. Furthermore, the transferability makes the universal item representation useful to other parts of the Alibaba online system, such as search box recommendation, click-through rate prediction, fraud transaction detection, and coupon push system.

There are several interesting problems to be investigated in our future work:

- Our proposed DINOP may be applied in various fields, such as finance and marketing, in addition to e-commerce. Given well-defined static features, dynamic features and control signals, the DINOP can obtain universal representation for multiple tasks. We can further investigate the generalization of DINOP.
- The BOP is a task in our SPOP problem. According to the pre-sales data, the difficulty of this task has decreased significantly. A more generalized problem of this task widely exists in the e-commerce, that is to predict whether a specific user will buy a specific item on a given day. All the users, sellers, and e-commerce platform will benefit from tackling this very challenge issue.

ACKNOWLEDGMENTS

This work is supported by Alibaba-Zhejiang University Joint Institute of Frontier Technologies, and National Natural Science Foundation of China (Grant No: U1866602).

SHEN Xin would like to express the deepest appreciation to Jingyi Zhang for her continued assistance and encouragement.

REFERENCES

[1] Gediminas Adomavicius and Alexander Tuzhilin. 2005. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge & Data Engineering* 6 (2005), 734–749.
[2] Deepak Agarwal and Bee-Chung Chen. 2009. Regression-based latent factor models. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 19–28.
[3] Deepak Agarwal and Bee-Chung Chen. 2010. fLDA: matrix factorization through latent dirichlet allocation. In *Proceedings of the third ACM international conference on Web search and data mining*. ACM, 91–100.
[4] Charu C Aggarwal et al. 2016. *Recommender systems*. Springer.
[5] Luc Bauwens, Sébastien Laurent, and Jeroen VK Rombouts. 2006. Multivariate GARCH models: a survey. *Journal of applied econometrics* 21, 1 (2006), 79–109.
[6] Fedor Borisyuk, Liang Zhang, and Krishnaram Kenthapadi. 2017. LiJAR: A system for job application redistribution towards efficient career marketplace. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 1397–1406.
[7] George EP Box and David A Pierce. 1970. Distribution of residual autocorrelations in autoregressive-integrated moving average time series models. *Journal of the American statistical Association* 65, 332 (1970), 1509–1526.
[8] John S Breese, David Heckerman, and Carl Kadie. 1998. Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., 43–52.
[9] Jiajun Bu, Xin Shen, Bin Xu, Chun Chen, Xiaofei He, and Deng Cai. 2016. Improving collaborative recommendation via user-item subgroups. *IEEE Transactions on Knowledge and Data Engineering* 28, 9 (2016), 2363–2375.
[10] Rich Caruana. 1997. Multitask learning. *Machine learning* 28, 1 (1997), 41–75.
[11] Tianqi Chen and Carlos Guestrin. 2016. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*. ACM, 785–794.
[12] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, et al. 2016. Wide & deep learning for recommender systems. In *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems*. ACM, 7–10.
[13] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078* (2014).
[14] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of Machine Learning Research* 12, Aug (2011), 2493–2537.
[15] Paul Covington, Jay Adams, and Emre Sargin. 2016. Deep neural networks for youtube recommendations. In *Proceedings of the 10th ACM Conference on Recommender Systems*. ACM, 191–198.
[16] Li Deng, Geoffrey Hinton, and Brian Kingsbury. 2013. New types of deep neural network learning for speech recognition and related applications: An overview. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 8599–8603.
[17] John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research* 12, Jul (2011), 2121–2159.
[18] Jerome H Friedman. 2001. Greedy function approximation: a gradient boosting machine. *Annals of statistics* (2001), 1189–1232.
[19] Ross Girshick. 2015. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*. 1440–1448.
[20] Michael Greenacre. 2017. *Correspondence analysis in practice*. Chapman and Hall/CRC.
[21] David Harris and Sarah Harris. 2010. *Digital design and computer architecture*. Morgan Kaufmann.
[22] Balázs Hidasi and Alexandros Karatzoglou. 2018. Recurrent neural networks with top-k gains for session-based recommendations. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. ACM, 843–852.
[23] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.
[24] Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. 2013. Learning deep structured semantic models for web search using clickthrough data. In *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management*. ACM, 2333–2338.
[25] Melvin Johnson, Mike Schuster, Quoc V Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, et al. 2017. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics* 5 (2017), 339–351.
[26] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* 8 (2009), 30–37.
[27] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. *nature* 521, 7553 (2015), 436.
[28] Bin Liu, Yanjie Fu, Zijun Yao, and Hui Xiong. 2013. Learning geographical preferences for point-of-interest recommendation. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 1043–1051.
[29] Xiaodong Liu, Jianfeng Gao, Xiaodong He, Li Deng, Kevin Duh, and Ye-Yi Wang. 2015. Representation learning using multi-task deep neural networks for semantic classification and information retrieval. (2015).
[30] Xin Liu and Ryutaro Ichise. 2017. Food sales prediction with meteorological dataâĂŤa case study of a japanese chain supermarket. In *International Conference on Data Mining and Big Data*. Springer, 93–104.
[31] Jiaqi Ma, Zhe Zhao, Xinyang Yi, Jilin Chen, Lichan Hong, and Ed H Chi. 2018. Modeling task relationships in multi-task learning with multi-gate mixture-of-experts. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 1930–1939.
[32] H Brendan McMahan, Gary Holt, David Sculley, Michael Young, Dietmar Ebner, Julian Grady, Lan Nie, Todd Phillips, Eugene Davydov, Daniel Golovin, et al. 2013. Ad click prediction: a view from the trenches. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 1222–1230.
[33] Yabo Ni, Dan Ou, Shichen Liu, Xiang Li, Wenwu Ou, Anxiang Zeng, and Luo Si. 2018. Perceive Your Users in Depth: Learning Universal User Representations from Multiple E-commerce Tasks. *arXiv preprint arXiv:1805.10727* (2018).
[34] Muhammad Qamar Raza and Abbas Khosravi. 2015. A review on artificial intelligence based load demand forecasting techniques for smart grid and buildings. *Renewable and Sustainable Energy Reviews* 50 (2015), 1352–1372.
[35] Sebastian Ruder. 2017. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098* (2017).
[36] Ruslan Salakhutdinov, Andriy Mnih, and Geoffrey Hinton. 2007. Restricted Boltzmann machines for collaborative filtering. In *Proceedings of the 24th international conference on Machine learning*. ACM, 791–798.
[37] Suvash Sedhain, Aditya Krishna Menon, Scott Sanner, and Lexing Xie. 2015. Autorec: Autoencoders meet collaborative filtering. In *Proceedings of the 24th International Conference on World Wide Web*. ACM, 111–112.
[38] Nitin Anand Shrivastava, Abbas Khosravi, and Bijaya Ketan Panigrahi. 2015. Prediction interval estimation of electricity prices using PSO-tuned support vector machines. *IEEE Transactions on Industrial Informatics* 11, 2 (2015), 322–331.
[39] SHEN Xin, Hongxia Yang, Weizhao Xian, Martin Ester, Jiajun Bu, Zhongyao Wang, and Can Wang. 2018. Mobile Access Record Resolution on Large-Scale Identifier-Linkage Graphs. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 886–894.