Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

A novel confidence estimation method for heterogeneous implicit feedback*

Jing WANG†‡, Lan-fen LIN, Heng ZHANG, Jia-qi TU, Peng-hua YU

(*College of Computer Science and Technology*, *Zhejiang University*, *Hangzhou 310027*, *China*)

†E-mail: cswangjing@zju.edu.cn

Received Aug. 11, 2016; Revision accepted Nov. 9, 2016; Crosschecked Nov. 24, 2017

Abstract: Implicit feedback, which indirectly reflects opinion through user behaviors, has gained increasing attention in recommender system communities due to its accessibility and richness in real-world applications. A major way of exploiting implicit feedback is to treat the data as an indication of positive and negative preferences associated with vastly varying confidence levels. Such algorithms assume that the numerical value of implicit feedback, such as time of watching, indicates confidence, rather than degree of preference, and a larger value indicates a higher confidence, although this works only when just one type of implicit feedback is available. However, in real-world applications, there are usually various types of implicit feedback, which can be referred to as heterogeneous implicit feedback. Existing methods cannot efficiently infer confidence levels from heterogeneous implicit feedback. In this paper, we propose a novel confidence estimation approach to infer the confidence level of user preference based on heterogeneous implicit feedback. Then we apply the inferred confidence to both point-wise and pair-wise matrix factorization models, and propose a more generic strategy to select effective training samples for pair-wise methods. Experiments on real-world e-commerce datasets from Tmall.com show that our methods outperform the state-of-the-art approaches, considering several commonly used ranking-oriented evaluation criteria.

Key word: Recommender systems; Heterogeneous implicit feedback; Confidence; Collaborative filtering; E-commerce
https://doi.org/10.1631/FITEE.1601468 **CLC number:** TP391

1 Introduction

E-commerce has grown rapidly in recent years and this has resulted in a huge volume of products and services. As users are provided with more options, it becomes, in turn, more difficult for users to make the right choice. These developments highlight the importance of recommender systems, which aim at helping users to find products and services that best meet their needs and interests (Ricci *et al*., 2011).

Recommender systems rely on different types of input. The most convenient is explicit feedback, which directly tells us the user preference, such as 5-star ratings, thumbs-up/down, or like and dislike. Collaborative filtering (CF), which is one of the most successful recommendation techniques, has been well studied to exploit explicit feedback (Tuzhilin and Adomavicius, 2005; Park *et al*., 2012; Bobadilla *et al*., 2013). However, explicit feedback is not always available in real-world applications. Implicit feedback, which indirectly reflects opinion through user behaviors, such as purchase and click, is easy to gather, without incurring into any overhead on users. Recently, implicit feedback has gained wider attention in recommender system communities due to its availability and richness.

One solution for dealing with implicit feedback is treating the data as an indication of positive and negative preferences associated with vastly varying confidence levels (Hu *et al*., 2008). Such algorithms

‡ Corresponding author

* Project supported by the National Basic Research Program (973) of China (No. 2015CB352400) and the National Key Research and Development Program of China (No. 2016YFB1200203-03)

ORCID: Jing WANG, http://orcid.org/0000-0002-0149-2049

assume that the numerical value of implicit feedback indicates confidence, rather than degree of preference, and a larger value indicates higher preference, which is workable only when just one type of implicit feedback is available. However, there are many types of implicit feedback in most real-world applications, and different types of implicit feedback have different abilities to indicate confidence, which is referred to as heterogeneous implicit feedback (Pan *et al*., 2015). For example, in online retail platforms: (1) a user clicks an item when he/she wants to know more details about that item, but we do not know whether the user is satisfied with the item or not; (2) a user collects an item (adds an item into favorites) indicating that he/she might review the item later, which shows higher confidence in the preference than a click; (3) a user adds an item into his/her shopping cart indicating that he/she may want to buy the item. Yet, sometimes a user puts competitive items into a shopping cart but purchases only one of them, or finds the one not interesting enough later on; (4) a user buys an item and pays for it. Since the ultimate goal of a recommender system for an online retail platform is to prompt users to make a purchase decision, it is natural to think that purchased items are preferred by users. Even if the user is unsatisfied with the product received, it shows his/her preference for that type of product. Summing up the implicit feedback discussed above, we can see that it can be classified into two categories: certain feedback such as purchase, which indicates certain preference, and uncertain feedback, such as click, collect, and cart, which indicates uncertain preference. Certain feedback is high-quality user feedback, but usually very sparse. It is important to incorporate uncertain feedback to ease the sparsity problem, but it is also challenging to characterize the confidence in a user preference from uncertain feedback.

In this research, we study the problem of heterogeneous implicit feedback, where both certain feedback and multiple types of uncertain feedback are available. First, we study the characteristics of different types of implicit feedback and propose a novel confidence estimation approach to infer the confidence level from heterogeneous implicit feedback. Then we apply the inferred confidence to both point-wise and pair-wise matrix factorization models, and propose a more generic strategy to select training samples for pair-wise methods. Experiments on real-world e-commerce data show that our methods outperform the state-of-the-art approaches, considering several commonly used ranking-oriented evaluation criteria. The contributions of this research are summarized as follows:

1. We propose a novel confidence estimation approach to quantify the confidence of user preference based on heterogeneous implicit feedback.

2. We apply the inferred confidence to both point-wise and pair-wise matrix factorization models, and propose a more generic strategy to construct training samples for pair-wise methods.

3. We conduct extensive experiments on real-world e-commerce data from Tmall.com. The results show that our approach can greatly improve the original point-wise and pair-wise methods, considering several commonly used ranking-oriented evaluation criteria.

This paper is an extension of our work originally reported in Proceedings of the 18th Asia-Pacific Web Conference (Wang *et al*., 2016). The main changes are: (1) A formal definition of the recommendation problem studied is given; (2) Many details about the experiments are included; (3) Detailed features in the engineering work are given; (4) Additional experiments are conducted on confidence estimation, and the distribution of the confidence learned by our method and adaptive Bayesian personalized ranking (ABPR) is analyzed. In addition to these important improvements, most of the content is modified to make it easier to read.

2 Related work

Collaborative filtering is one of the most popular recommendation techniques (Ricci *et al*., 2011; Park *et al*., 2012; Bobadilla *et al*., 2013; Shi *et al*., 2014). Collaborative filtering methods are based on the assumption that users agreed previously and will also agree in the future, and they will like similar kinds of items to the ones they liked in the past. Explicit-feedback-based collaborative filtering methods have been well studied (Tuzhilin and Adomavicius, 2005; Park *et al*., 2012; Bobadilla *et al*., 2013). However, explicit feedback is not always available. Recently, implicit feedback has received increasing attention in recommender system communities due to its availability and richness, and many algorithms have been proposed.

A major solution to exploit implicit feedback is to treat the data as an indication of positive and negative preferences associated with vastly varying confidence levels, instead of treating implicit feedback as a degree of preference. For example, one of the earliest solutions for handling implicit feedback, ImplicitALS (Hu *et al.*, 2008), consists of a matrix factorization model, and a point-wise-based (for each user-item instance) objective function that is weighted by the confidence derived from the observations of behaviors. ImplicitALS assumes that the numerical value of implicit feedback, such as time of watching, indicates confidence, and a larger value indicates a higher confidence. However, such a strategy is workable only when just one type of implicit feedback is available. When dealing with heterogeneous implicit feedback, such simple rules cannot exploit well the information contained in various data. Another popular algorithm, Bayesian personalized ranking (BPR) (Rendle *et al.*, 2009), is based on the assumption that a user prefers a consumed item to an unconsumed item. BPR with confidence (BPRC) (Wang *et al.*, 2012) extends BPR by adding a confidence obtained from external context information for each sample, and optimizes a confidence-weighted objective function. However, in most applications the confidence is not available. All of these methods consider only one type of implicit feedback. Pan *et al.* (2015) was the first to study the problem of heterogeneous implicit feedback. They took users' transaction records as a certain feedback and examination records as uncertain feedback, and then proposed an ABPR that learns a confidence weight for each examination record. However, ABPR still treats all cases of uncertain feedback equally.

Heterogeneous one-class collaborative filtering (HOCCF) (Pan *et al.*, 2016) takes positive feedback as the target data and implicit examinations as the auxiliary data. HOCCF not only learns a similarity between candidate item i and a preferred item, but also learns an additional similarity between item i and an examined item j, and then HOCCF can estimate the preference of user u on item i in a similar way to that of item-oriented, memory-based collaborative filtering with all of the items' neighbors. The limitation of HOCCF lies in its generalization ability, and further improvement is required to apply it to pair-wise methods.

In this research, we propose a novel confidence estimation method that can deal with multiple types of uncertain feedback to characterize the confidence such that we can believe a user prefers an item. We also apply the confidence to both point-wise and pair-wise matrix factorization models.

3 Framework

Let U be the set of all users and I the set of all items. We reserve special indexing letters for distinguishing users from items: for users u, v and for items i, j. Let $C=\{c_{ui}\}$ be the confidence of user preference, where a higher value means a stronger confidence. Certain feedback and uncertain feedback are denoted as $T=\{(u, i)\}$ and $E=\{(u, i)\}$, respectively, and there are the following cases for (u, i) pairs in the system (Fig. 1):

1. $(u, i)\in U\times I$, including all (u, i) pairs in the system.

2. $(u, i)\in E$, meaning that user u has uncertain implicit feedback for item i.

3. $(u, i)\in T$, meaning that user u has certain feedback on item i. We assume that $(u, i)\in T$ is a precondition of $(u, i)\in T$. For example, there are always click records before a user purchases something, and reading actions before forwarding micro-blogs.

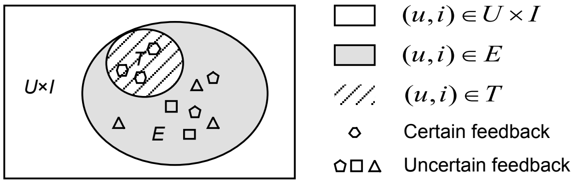

Fig. 1 Heterogeneous implicit feedback

Given (u, i) and the corresponding implicit feedback, the recommendation framework is as described in Fig. 2. When $(u, i)\in T$, we consider that user u likes item i, so we can directly set $c_{ui}=1$, which is the highest confidence in the system. When $(u, i)\in E$ but $(u, i)\notin T$, we infer the confidence from the corresponding implicit feedback using our proposed method. Then the two parts are merged and used in collaborative filtering models. The candidate items are from $(u, i)\notin E$, i.e., the items with which a user has not interacted, exactly as in collaborative filtering methods.

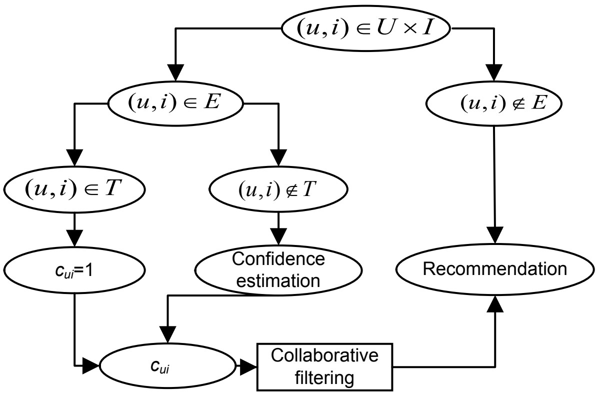

Fig. 2 Recommendation framework

4 Confidence estimation based on heterogeneous implicit feedback

Confidence in a user preference is influenced by many factors. Take online retail platforms as an example: (1) Different types of implicit feedback have different abilities to indicate confidence, such as click and collect; (2) The number of times an event has occurred indicates different levels of confidence (for example, a frequently clicked item is more likely to be preferred than an item that is clicked only once); (3) A user may have multiple behaviors with respect to an item, and it is hard to compare a frequently clicked item i with another item j in the favorites, which is clicked only once; (4) Different users have different habits (for example, some users like to put candidate products into favorites, while others like to put them into shopping carts, and some users click a great deal but purchase rarely, while others tend to purchase a larger portion of their clicked items). These are common factors but not the only factors that influence confidence, so it is complex to design a function to calculate confidence, which is the key idea of ImplicitALS (Hu *et al*., 2008).

As mentioned above, certain feedback can indicate full confidence and we set c_{ui}=1, if we can use certain feedback to represent uncertain feedback, and then we can connect uncertain feedback with confidence. In this section, we propose a novel confidence estimation method to quantify the confidence level based on the heterogeneous implicit feedback.

In real-world applications, certain feedback and uncertain feedback are not independent of each other. For example, there are always click records before a user purchases something, and reading actions before forwarding micro-blogs. So, in our approach, we assume that $(u, i)\in E$ is a precondition of $(u, i)\in T$; i.e., if user u conveys certain feedback on item i, there must also be uncertain feedback. There are relationships between certain feedback and instances of uncertain feedback. For example, a clicked item has a small probability of being purchased; an item in favorites or a shopping cart has a higher probability of being purchased; a frequently clicked item also has a high probability of being purchased. It is natural to consider what the probability of the existence of certain feedback is given the statistics for uncertain feedback. The probability is the connection between uncertain feedback and certain feedback, and certain feedback indicates full confidence, so the probability can be used to represent the degree of confidence for uncertain feedback. The task of confidence estimation is transformed into predicting the probability of the existence of certain feedback, given the statistics from instances of uncertain feedback.

We adopt supervised learning to build a model to predict the probability of the existence of certain feedback. To solve a supervised learning problem, we must perform the following steps:

1. Determine the type of training examples. Our purpose is to predict the probability of the existence of certain feedback, given the uncertain feedback of a user-item pair. Thus, a training example is a user-item pair characterized by uncertain feedback.

2. Determine the input feature representation. The accuracy of the learning function depends strongly on how the input object is represented. The features can be derived from: (1) statistics from uncertain feedback, such as click times and reading time; (2) user profiles, such as age, gender, and behavior bias; (3) item profiles, such as category, brand, price, and popularity. The specific features are related to the application, so we do not go in depth in this section.

3. Determine the labels for supervised learning. There is a problem when constructing the labels. If $(u, i)\in E$ and $(u, i)\in T$, we can label this (u, i) as class 1; if $(u, i)\in E$ and $(u, i)\notin T$, there are two possible situations for this (u, i): user u will have instances of

certain feedback on item i in the future, or not. This is easy to understand in e-commerce settings. For example, the clicked but not purchased items have small probabilities of being purchased later. Therefore, maybe $(u, i)\in T$ will be true in the future, denoted as $(u, i)\in T'$. Both T and T' mean certain feedback, but T can be observed in the training data, while T' is the future data, which cannot be observed in the training data. We are not sure whether $(u, i)\in T'$ is true or not before the user actually does it. Thus, we have only one kind of label, and the labels of user-item pairs in $\{(u, i)|(u, i)\in E, (u, i)\notin T\}$ are unknown. The unlabeled instances contain a great proportion of negative samples, and a small proportion of potential positive samples. We borrow ideas from one-class collaborative filtering (OCCF) (Pan *et al.*, 2008): sample a portion of $\{(u, i)|(u, i)\in E, (u, i)\notin T\}$ as negative examples to balance the extent of treating unlabeled instances as negative examples (class 0). Thus, the task turns into a binary classification problem.

4. Determine the type of learning model. Models for binary classification problems have been well studied, such as logistic regression (Freedman, 2009), random forest (Liaw and Wiener, 2002), and gradient boosting decision tree (GBDT) (Friedman, 2001; 2002). We just need to choose the most suitable one within a specific real-world application.

5. Run the learning algorithm on the assembled training set, and evaluate the accuracy of the learned model on a test set that is separate from the training set.

Once the classification model is learned, we can predict the probability that (u, i) belongs to class 1, i.e., $(u, i)\in T'$, for the user-item pairs in $\{(u, i)|(u, i)\in E, (u, i)\notin T\}$. Finally, we transform the heterogeneous implicit feedback into numeric values to characterize the confidence:

$$c_{ui}=\begin{cases}1, & (u,i)\in T,\\ p_{(u,i)\in T'}, & (u,i)\in E,\ (u,i)\notin T,\end{cases} \tag{1}$$

where $P_{(u,\ i)\in T}$ denotes the probability of $(u, i)\in T'$ predicted by the classification model, and c_{ui} is the confidence inferred from the heterogeneous implicit feedback. By doing so, we do not need to care about the contribution of every type of implicit feedback, and can quantify the confidence in a unified approach.

Our c_{ui} can be directly applied to point-wise models, such as ImplicitALS, by replacing their c_{ui}, or be added as the weight of each sample for other point-wise models. In the next section, we will introduce how to use c_{ui} to generalize BPR. We will briefly introduce the features and classification models in the experimental part, and explore the relationship between the accuracy of the classification and the final accuracy of the recommendation.

5 Confidence estimation based Bayesian personalized ranking

Matrix factorization has become very popular due to its high accuracy and scalability (Koren, 2008; Volkovs and Yu, 2015). Matrix factorization models map both users and items to a joint latent factor space of dimensionality f, such that user-item interactions are modeled as inner products in that space. Accordingly, each item i is associated with a vector $\boldsymbol{y}_i\in\mathbb{R}^f$, and the elements of $\boldsymbol{y}_i$ measure the extent to which the item possesses those latent factors. Similarly, each user u is associated with a vector $\boldsymbol{x}_u\in\mathbb{R}^f$. The resulting dot product $\boldsymbol{x}_u^{\mathrm{T}}\boldsymbol{y}_i$ captures the interaction between user u and item i—the user's overall interest in the item's characteristics. The major challenge is computing the mapping $\boldsymbol{x}_u, \boldsymbol{y}_i\in\mathbb{R}^f$ of each item and user to factor vectors. Considering the objective functions they use, matrix factorization models can be classified into point-wise and pair-wise methods. The objective functions of point-wise methods are designed for each user-item instance (Hu *et al.*, 2008; Lee *et al.*, 2008; Koren, 2010). Pair-wise methods minimize a ranking objective function: if user u prefers item i over item j, it is denoted as $i\succ_u j$, and the goal is to correctly order such item pairs. Pair-wise algorithms are the state-of-the-art methods to deal with implicit feedback, such as BPR (Rendle *et al.*, 2009). We generalize BPR to deal with heterogeneous implicit feedback. A pair-wise, confidence-based matrix factorization model optimizes the objective function:

$$\begin{aligned}\min\sum_{(u,i,j):i\succ_u j}&\left(-\ln\left(1\Big/\left(1+\exp\left(-c_{uij}(\boldsymbol{x}_u^{\mathrm{T}}\boldsymbol{y}_i-\boldsymbol{x}_u^{\mathrm{T}}\boldsymbol{y}_j)\right)\right)\right)\right.\\&\left.+\lambda\left(\sum_u\|\boldsymbol{x}_u\|^2+\sum_i\|\boldsymbol{y}_i\|^2\right)\right),\end{aligned} \tag{2}$$

where $-\ln\left(1/\left(1+\exp\left(-c_{uij}(\boldsymbol{x}_u^{\mathrm{T}}\boldsymbol{y}_i-\boldsymbol{x}_u^{\mathrm{T}}\boldsymbol{y}_j)\right)\right)\right)$ is the loss function designed to encourage pair-wise comparison, $c_{uij}=c_{ui}-c_{uj}$ indicates how much we trust that user u prefers item i over item j, $\boldsymbol{x}_u^{\mathrm{T}}\boldsymbol{y}_i-\boldsymbol{x}_u^{\mathrm{T}}\boldsymbol{y}_j$ is the difference of the predicted preference values between items i and j, and $\lambda\left(\sum_u\|\boldsymbol{x}_u\|^2+\sum_i\|\boldsymbol{y}_i\|^2\right)$ is the regularization term used to avoid overfitting. The challenge is how to choose suitable (u, i, j) triples that satisfy $i\succ_u j$, and whether user u prefers item i over item j. In our approach, (u, i, j) is a triple from $\{(u, i, j)\,|(u, i)\in T\cup E, c_{ui}>c_{uj}\}$, which means user u prefers item i over item j. Choosing triples from $c_{ui}>c_{uj}$ includes the following situations: (1) $(u, i)\in T$, $(u, j)\notin T$; (2) $(u, i)\in E$, $(u, j)\notin E$; (3) $(u, i)\in E$, $(u, j)\in E$, and $c_{ui}>c_{uj}$. BPR (Rendle *et al.*, 2009) contains situation (1), ABPR (Pan *et al.*, 2015) contains situations (1) and (2), and our method contains all of these situations. We are not absolutely sure that user u prefers item i over j, so we use $c_{uij}=c_{ui}-c_{uj}$ to describe the confidence in the $i\succ_u j$ relation. We have a unified way to choose training samples for pair-wise methods, which can bring about more effective, comparable pairs and further relieve the sparsity problem. We name this method confidence estimation based BPR (CL-BPR), because it is an improved version of BPR and ABPR, and is based on a confidence estimation step to quantify the confidence from heterogeneous implicit feedback data.

6 Experiments

6.1 Dataset and statistics

We conduct extensive experiments on two real-world e-commerce datasets provided by Tmall.com. The first one is the REC-TMALL dataset released in the first stage of the Tmall Recommendation Prize 2014 (https://tianchi.shuju.aliyun.com/datalab/index.htm). This dataset is focused on brand recommendation; that is, an item in this dataset means a brand. The second one is the dataset released in the IJCAI-15 Competition (http://tianchi.aliyun.com/datlab/dataSet.htm?spm=5176.100073.888.13.nt1XTA&id=1), which contains interactions between users and products. Both datasets contain data fields listed in Table 1.

Table 1 Data fields

Field	Description	Instruction
User ID	Unique identifier of a user	Sampling and encryption
Product/ brand ID	Unique identifier of a product/brand	Sampling and encryption
Time	The time when interaction occurred	Precision level to the specific day
Action type	Type of action	Buy, click, collect, and cart

This kind of implicit feedback data is very common on online retail platforms. We regard ‘buy’ as certain feedback, and other behaviors as uncertain feedback data. Statistics about the two datasets are shown in Table 2, and we can see that certain feedback is very sparse, while uncertain feedback is more abundant.

We first split each dataset into two parts, Dataset1 and Dataset2, evenly by users. Dataset1 is used to train and evaluate the classification model in the confidence estimation step, and the learning model is used to quantify the confidence for the heterogeneous implicit feedback in Dataset2. Then Dataset2 is used to train and test the collaborative filtering model. Specifically, we split Dataset1 into two parts Dataset1-train and Dataset1-test evenly by users. Dataset1-train is used to train the classification model, and Dataset1-test is used to evaluate the classification model. We split Dataset2 according to time, since recommenders normally predict users’ future preference by exploiting historical data. For the REC-TMALL dataset, we used 0–90 d as Dataset2-train, and 91–122 d as Dataset2-test; for the IJCAI-15 dataset, we used 0–110 d as Dataset2-train, and 111–160 d as Dataset2-test.

Table 2 Statistics about REC-TMALL and IJCAI-15 datasets

Dataset	#User	#Item	#Click	#Buy	#Collect	#Cart	Sparsity(T)	Sparsity($T\cup E$)
REC-TMALL	884	9531	174 539	6984	1204	153	0.08%	0.684%
IJCAI-15	424 170	1 090 390	4 8550 713	3 292 144	3 005 723	76 750	0.0007%	0.0069%

6.2 Experiment settings for confidence estimation

6.2.1 Feature engineering

In the confidence estimation step, features are needed for the classification model. When defining features, our principles are: (1) What kind of behavior can indicate that a user wants to purchase an item; (2) What kind of item is more likely to be purchased (item-bias); (3) What kind of user is more likely to purchase (user-bias). We first consider the following groups of features (note that we have three types of uncertain feedback, so each line has three features):

1. user u's behavior count for each instance of uncertain feedback on item i,
2. user u's average behavior count for each instance of uncertain feedback over items,
3. item i's average behavior count over users,
4. user u's total behavior count, and
5. item i's total behavior count.

The features above are mainly from the perspective of behavior counts. In addition, we think that a user clicking on an item twice a day is different from clicking on the same item once two days. Therefore, we double the number of features by replacing absolute counts with days. Now we have 5×3×2 features to characterize uncertain feedback data, denoted as a 'small feature set'. We also look at various combinations of these features to obtain more complex features, and merge them with the original features, denoted as the 'full feature set'. We use these two feature sets to predict the probability of certain feedback, respectively.

6.2.2 Classification methods and evaluation metrics

After feature extraction, we use logistic regression (Freedman, 2009), random forest (Liaw and Wiener, 2002), and GBDT (Friedman, 2001; 2002) in the confidence estimation step. We take advantage of scikit-learn (http://scikit-learn.org/stable/), which is an open-source machine-learning tool in Python and contains the methods above. For logistic regression, we set penalty=l2, solver=sag, tol=1, and reserve the default values for the other parameters. For random forest, we set n_estimators=500, min_samples_split =5, and reserve the default values for the other parameters. For GBDT, we set learning_rate=0.1, n_estimators=500, max_depth=6, and reserve the default values for the other parameters.

We evaluate the classification accuracy using the area under the curve (AUC), which is equal to the probability that a classifier ranks a randomly chosen positive instance higher than a randomly chosen negative one. We adjust the features and classification methods to obtain different versions of classification models and different AUCs, and then observe the corresponding accuracy of the recommendation step. Thus, we can know whether the accuracy of the recommendation will be improved or not when the accuracy of the classification is improved.

6.3 Experiment settings for collaborative filtering

We first analyze the performances of point-wise matrix factorization models using the confidence generated by our confidence estimation method, and by other strategies. Specifically, we use the ImplicitALS model, and there are three versions according to the confidence generation strategies:

1. ImplicitALS(T) for certain feedback (purchase) only. In ImplicitALS(T), the purchase count is used to compute c_{ui}. The function for computing c_{ui} is $c_{ui}=1+\alpha\cdot\text{count}_{ui}$, where α is a constant for controlling the growth rate of confidence when the behavior count increases. In our experiments, setting $\alpha=50$, num_factors=50, $\lambda=0.001$, and learning_rate=0.001 is found to produce the best results.
2. ImplicitALS($T\cup E$) for the combination of certain feedback and uncertain feedback. In ImplicitALS($T\cup E$), the total interaction count is used to compute c_{ui}. The function for computing c_{ui} is the same as ImplicitALS(T) and the difference is that the count here includes all kinds of implicit feedback. Setting $\alpha=50$, num_factors=50, $\lambda=0.001$, and learning_rate=0.01 is found to produce the best results.
3. CL-ImplicitALS. The c_{ui} is obtained by our confidence estimation method. Setting num_factors= 100, $\lambda=0.01$, and learning_rate=0.01 is found to produce the best results.

Then we analyze the performances of the pairwise matrix factorization models, including:

1. BPR(T) for certain feedback only. In BPR(T), user u prefers item i over item j if $(u, i)\in T$ and $(u, j)\notin T$. Setting num_factors=200, $\lambda=0.1$, and learning_rate= 0.01 is found to produce the best results.
2. BPR($T\cup E$) for the combination of certain feedback and uncertain feedback. In BPR($T\cup E$), user u prefers item i over item j if $(u, i)\in T\cup E$ and $(u, j)\notin$

$T\cup E$. Setting num_factors=300, λ=0.1, and learning_rate= 0.01 is found to produce the best results.

3. In ABPR, user u prefers item i over item j if $(u, i)\in T$ and $(u, j)\notin T$, and user u also prefers item i over item j if $(u, i)\in T\cup E$ and $(u, j)\notin T\cup E$. Setting num_factors=200, λ=0.01, and learning_rate=0.01 is found to produce the best results. Other settings exactly follow those in Pan *et al*. (2015).

We use LibRec (http://www.librec.net/) as the basis to conduct our experiments. LibRec is a GPL-licensed Java library, aimed at solving rating predictions and item ranking problems by implementing a suite of state-of-the-art recommendation algorithms. Specifically, we use LibRec's WRMF (an alias for ImplicitALS) to carry out ImplicitALS(T), ImplicitALS($T\cup E$), and CL-ImplicitALS, and we use LibRec's BPR to carry out BPR(T) and BPR($T\cup E$). Then we implement ABPR and CL-BPR on the basis of LibRec's BPR.

Ranking-oriented evaluation metrics, including Precision@5, Precision@10, Recall@5, Recall@10, AUC, MAP, NDCG, and MRR, are selected to evaluate our algorithms and the baseline approaches. Since the ultimate goal of recommender systems on an online retail platform is to prompt users to make a purchase decision, a purchased item in the test set is treated as a positive instance.

6.4 Results and analysis

6.4.1 Results of confidence estimation

We compare the prediction accuracy of confidence estimation using two feature sets (small feature set and full feature set, as described in the experiment settings), and using different classification algorithms. We find that using the full feature set and the GBDT model can achieve the best results in Table 3. We also analyze the distribution of the confidence value learned by our method and by ABPR. We can see that our method obtains a smoother confidence distribution than ABPR, and it is more in line with the distribution of user preference in reality in Fig. 3.

Table 3 AUC for confidence estimation with the REC-TMALL and IJCAI-15 datasets using two feature sets, small and full

Dataset	AUC			
	REC-TMALL		IJCAI-15	
	Small	Full	Small	Full
Logistic regression	0.6327	0.6457	0.6509	0.6625
Random forest	0.6841	0.6954	0.8983	0.89999
GBDT	0.7039	0.7111	0.9011	0.9035

6.4.2 Results of collaborative filtering

First, we analyze the point-wise confidence-based methods, including the proposed CL-ImplicitALS and the baseline approaches, ImplicitALS(T) and ImplicitALS($T\cup E$), in Tables 4 and 5, from which we have the following observations:

1. ImplicitALS($T\cup E$) does not outperform ImplicitALS(T), and sometimes it is even worse than ImplicitALS(T). We think that this is because the original ImplicitALS treats all types of implicit feedback equally, which is not reasonable, so incorporating more data does not help improve the recommendation performance.

2. For both datasets, CL-ImplicitALS achieves the best results. It seems that our approaches can better leverage various types of implicit feedback and characterize the confidence levels well from heterogeneous implicit feedback.

Table 4 Recommendation performance of point-wise methods with the REC-TMALL dataset

Algorithm	P@5	P@10	R@5	R@10	AUC	MAP	NDCG	MRR
ImplicitALS(T)	0.0158	0.0112	0.0664	0.0857	0.5511	0.0336	0.0483	0.0416
ImplicitALS($T\cup E$)	0.0158	0.0125	0.0664	0.0903	0.5572	0.0339	0.0499	0.0428
CL-ImplicitALS	0.0211	0.0158	0.0725	0.1091	0.5763	0.0472	0.0676	0.0681

Table 5 Recommendation performance of point-wise methods with the IJCAI-15 dataset

Algorithm	P@5	P@10	R@5	R@10	AUC	MAP	NDCG	MRR
ImplicitALS(T)	0.0131	0.0094	0.0097	0.0140	0.5364	0.0063	0.0139	0.0317
ImplicitALS($T\cup E$)	0.0103	0.0082	0.0076	0.0122	0.5295	0.0054	0.0118	0.0256
CL-ImplicitALS	0.0212	0.0156	0.0155	0.0231	0.5576	0.0110	0.0233	0.0522

Then we analyze the pair-wise confidence-based methods, including the proposed CL-BPR and baseline approaches, BPR(T), BPR($T\cup E$), and ABPR, in Tables 6 and 7, from which we have the following observations:

In general, BPR($T\cup E$) outperforms BPR(T), and we think this is because BPR($T\cup E$) has more effective comparable item pairs than BPR(T) by incorporating uncertain feedback. Although we are not sure whether user u prefers item i or not if only uncertain feedback exists, compared to the extremely small probability that a user prefers an unseen item j, treating user u as preferring item i over item j can benefit more from relief of the sparsity problem.

3. ABPR does not outperform BPR($T\cup E$), which is unexpected. As previously mentioned, we analyze the automatically learned confidence in ABPR, and find these values vary very little, showing that ABPR cannot characterize the confidence contained in the heterogeneous implicit feedback of our datasets well.

4. For both datasets, our proposed CL-BPR achieves the best results, which validates the effectiveness of the proposed strategies for choosing training pairs and their corresponding confidence.

6.4.3 Other findings

We study the relationship between the accuracy of the classification model in predicting the

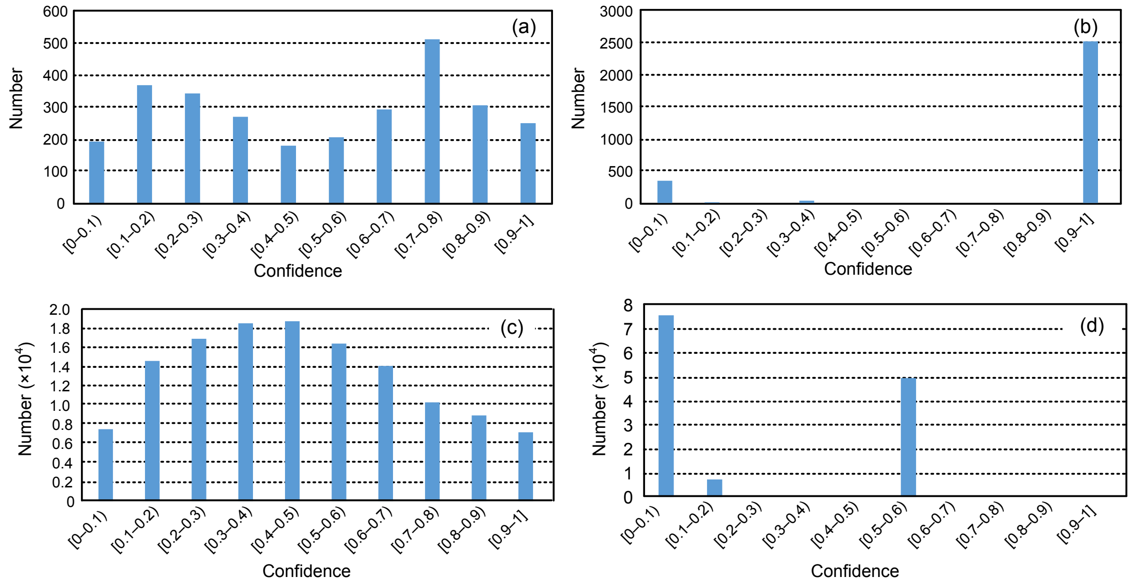

Fig. 3 Distributions of confidence: (a) confidence learned by our method on REC-TMALL; (b) confidence learned by ABPR on REC-TMALL; (c) confidence learned by our method on IJCAI-15; (d) confidence learned by ABPR on IJCAI-15

Table 6 Recommendation performance of pair-wise methods with the REC-TMALL dataset

Algorithm	P@5	P@10	R@5	R@10	AUC	MAP	NDCG	MRR
BPR(T)	0.0172	0.0152	0.0634	0.1098	0.5603	0.0350	0.0554	0.0439
BPR($T\cup E$)	0.0198	0.0165	0.0634	0.0867	0.5721	0.0352	0.0538	0.0541
ABPR	0.0211	0.0165	0.0662	0.1041	0.5759	0.0372	0.0592	0.0612
CL-BPR	0.0238	0.0185	0.0725	0.1135	0.5821	0.0459	0.0683	0.0688

Table 7 Recommendation performance of pair-wise methods with the IJCAI-15 dataset

Algorithm	P@5	P@10	R@5	R@10	AUC	MAP	NDCG	MRR
BPR(T)	0.0156	0.0129	0.0116	0.0191	0.5401	0.0090	0.0184	0.0376
BPR($T\cup E$)	0.0177	0.0133	0.0138	0.0208	0.5498	0.0091	0.0196	0.0408
ABPR	0.0162	0.0094	0.0115	0.0182	0.5429	0.0056	0.0139	0.0350
CL-BPR	0.0203	0.0157	0.0149	0.0233	0.5535	0.0116	0.0240	0.0529

probability of $(u, i) \in T'$, and the accuracy of collaborative filtering for the REC-TMALL and IJCAI-15 datasets, and then evaluate this through AUC (Fig. 4). We can see the coherence of their changing trends: better classification performance can generally result in better recommendation performance. This indicates that $(u, i) \in T'$ can characterize well the confidence in a user preference.

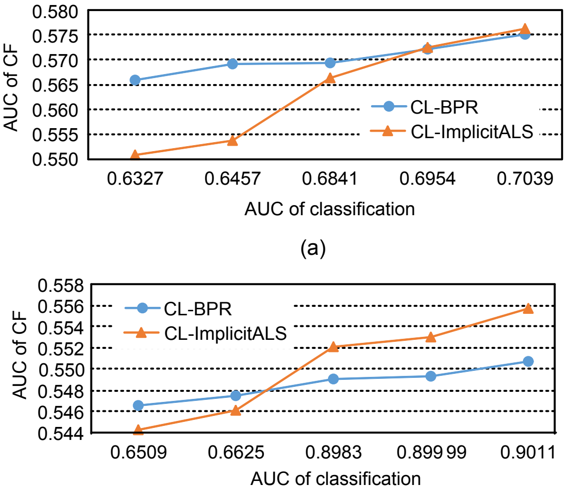

Fig. 4 AUC of the classification and of confidence-based collaborative filtering (CF): (a) REC-TMALL; (b) IJCAJ-15

7 Conclusions and future work

In this paper, we propose a novel confidence-estimation method to quantify the confidence of user preference based on heterogeneous implicit feedback, by studying the internal relations of two categories of implicit feedback: certain feedback and uncertain feedback. Our method can deal with certain feedback and multiple types of uncertain feedback in a unified way, while existing methods either deal with only one type of implicit feedback or treat all kinds of uncertain feedback equally. We also propose CL-BPR, which generalizes BPR and can construct more effective training samples from the heterogeneous implicit feedback data. Experiments on two real-world e-commerce datasets, one for brands and the other for products, show that our methods outperform the baseline approaches.

In future work, more features can be added, and more advanced classification algorithms can be used to improve the performance of confidence estimation. We can also apply the inferred confidence to more comprehensive collaborative filtering methods, such as context-aware approaches.

References

Bobadilla, J., Ortega, F., Hernando, A., *et al.*, 2013. Recommender systems survey. *Knowl.-Based Syst.*, **46**:109-132. https://doi.org/10.1016/j.knosys.2013.03.012

Freedman, D.A., 2009. Statistical Models: Theory and Practice (2nd Ed.). Cambridge University Press, Cambridge.

Friedman, J.H., 2001. Greedy function approximation: a gradient boosting machine. *Ann. Stat.*, **29**(5):1189-1232.

Friedman, J.H., 2002. Stochastic gradient boosting. *Comput. Stat. Data Anal.*, **38**(4):367-378. https://doi.org/10.1016/S0167-9473(01)00065-2

Hu, Y.F., Koren, Y., Volinsky, C., 2008. Collaborative filtering for implicit feedback datasets. Proc. 8th IEEE Int. Conf. on Data Mining, p.263-272. https://doi.org/10.1109/ICDM.2008.22

Koren, Y., 2008. Factorization meets the neighborhood: a multifaceted collaborative filtering model. Proc. 14th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining, p.426-434. https://doi.org/10.1145/1401890.1401944

Koren, Y., 2010. Factor in the neighbors: scalable and accurate collaborative filtering. *ACM Trans. Knowl. Discov. Data*, **4**(1), Article 1. https://doi.org/10.1145/1644873.1644874

Lee, T.Q., Park, Y., Park, Y.T., 2008. A time-based approach to effective recommender systems using implicit feedback. *Expert Syst. Appl.*, **34**(4):3055-3062. https://doi.org/10.1016/j.eswa.2007.06.031

Liaw, A., Wiener, M., 2002. Classification and regression by randomForest. *R News*, **2**(3):18-22.

Pan, R., Zhou, Y.H., Cao, B., *et al.*, 2008. One-class collaborative filtering. Proc. 8th IEEE Int. Conf. on Data Mining, p.502-511. https://doi.org/10.1109/ICDM.2008.16

Pan, W.K., Zhong, H., Xu, C.F., *et al.*, 2015. Adaptive Bayesian personalized ranking for heterogeneous implicit feedbacks. *Knowl.-Based Syst.*, **73**:173-180. https://doi.org/10.1016/j.knosys.2014.09.013

Pan, W.K., Liu, M.S., Ming, Z., 2016. Transfer learning for heterogeneous one-class collaborative filtering. *IEEE Intell. Syst.*, **31**(4):43-49. https://doi.org/10.1109/MIS.2016.19

Park, D.H., Kim, H.K., Choi, I.Y., *et al.*, 2012. A literature review and classification of recommender systems research. *Expert Syst. Appl.*, **39**(11):10059-10072. https://doi.org/10.1016/j.eswa.2012.02.038

Rendle, S., Freudenthaler, C., Gantner, Z., *et al.*, 2009. BPR: Bayesian personalized ranking from implicit feedback. Proc. 25th Conf. on Uncertainty in Artificial Intelligence, p.452-461.

Ricci, F., Rokach, L., Shapira, B., *et al*., 2011. Recommender Systems Handbook. Springer, Boston, MA, US. https://doi.org/10.1007/978-0-387-85820-3

Shi, Y., Larson, M., Hanjalic, A., 2014. Collaborative filtering beyond the user-item matrix: a survey of the state of the art and future challenges. *ACM Comput*. *Surv*., **47**(1):1-45. https://doi.org/10.1145/2556270

Tuzhilin, A., Adomavicius, G., 2005. Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. *IEEE Trans*. *Knowl*. *Data Eng*., **17**(6):734-749. https://doi.org/10.1109/TKDE.2005.99

Volkovs, M., Yu, G.W., 2015. Effective latent models for binary feedback in recommender systems. Proc. 38th Int. ACM SIGIR Conf. on Research and Development in Information Retrieval, p.313-322. https://doi.org/10.1145/2766462.2767716

Wang, J., Lin, L.F., Zhang, H., *et al*., 2016. Confidence-learning based collaborative filtering with heterogeneous implicit feedbacks. Proc. 18th Asia-Pacific Web Conf., p.444-455. https://doi.org/10.1007/978-3-319-45814-4_36

Wang, S., Zhou, X.B., Wang, Z.Q., *et al*., 2012. Please spread: recommending tweets for retweeting with implicit feedback. Proc. Workshop on Data-Driven User Behavioral Modelling and Mining from Social Media, p.19-22. https://doi.org/10.1145/2390131.2390140