JMLR: Workshop and Conference Proceedings 39:283–298, 2014 ACML 2014

Ordinal Random Fields for Recommender Systems

Shaowu Liu S.LIU@DEAKIN.EDU.AU
School of Information Technology
Deakin University
221 Burwood Highway, Vic 3125, Australia

Truyen Tran TRUYEN.TRAN@DEAKIN.EDU.AU
Pattern Recognition and Data Analytics
Deakin University
Waurn Ponds, Vic 3216, Australia

Gang Li GANG.LI@DEAKIN.EDU.AU
School of Information Technology
Deakin University
221 Burwood Highway, Vic 3125, Australia

Yuan Jiang JIANGYUAN@NJU.EDU.CN
National Key Laboratory for Novel Software Technology
Nanjing University
Nanjing, 210023, China

Editors: List of editors' names

Abstract

Recommender Systems heavily rely on *numerical preferences*, whereas the importance of *ordinal preferences* has only been recognised in recent works of *Ordinal Matrix Factorisation* (OMF). Although the *OMF* can effectively exploit ordinal properties, it captures only the higher-order interactions among users and items, without considering the localised interactions properly. This paper employs *Markov Random Fields* (MRF) to investigate the localised interactions, and proposes a unified model called *Ordinal Random Fields* (ORF) to take advantages of both the representational power of the *MRF* and the ease of modelling ordinal preferences by the *OMF*. Experimental result on public datasets demonstrates that the proposed *ORF* model can capture both types of interactions, resulting in improved recommendation accuracy.

Keywords: Ordinal Random Fields, Ordinal Matrix Factorisation, Markov Random Fields, Collaborative Filtering

1. Introduction

Recommender Systems (RecSys) aim to suggest items that are potentially of interest to users, where the items can be virtually anything such as *movies* and *attractions for travel*. To identify the appropriate items, RecSys use various sources of information including *item content* (Balabanović and Shoham, 1997) and *user preferences* (Koren et al., 2009). By far, *Collaborative Filtering* (Sarwar et al., 2001; Koren et al., 2009) is one of the most popular RecSys techniques, which exploits user preferences especially the *numerical preferences*.

However, numerical preferences are often difficult to collect as users may find it easier to tell which item is preferable to others, rather than expressing the precise degree of liking. Furthermore, researchers argued that numerical preferences may not be completely trustworthy (Koren and Sill, 2011; Brun et al., 2010). For example, the internal scales of users can be different, where the rating 4 out of 5 generally indicates high quality, but it is possible to be just fine for critical users. While users are not good at making consistent quantitative judgement, *ordinal preferences* are considered to be more consistent across like-minded users (Desarkar et al., 2010).

Ordinal preferences is an alternative view of user preferences, in which the relative orders between items are measured. To adopt ordinal preferences, substantial research efforts have been made over the past five years (Koren and Sill, 2011; Tran et al., 2012; Paquet et al., 2012; Sharma and Yan, 2013). While most data collections are still dominated by numerical preferences, the shift from numerical to ordinal is a slow process. Instead of going solely ordinal preferences in a sudden, most existing ordinal approaches begin with exploiting the ordinal properties possessed by numerical preferences. Among them, *Ordinal Matrix Factorisation* (OMF) has been suggested as an effective method in recent developments (Koren and Sill, 2011; Tran et al., 2012; Paquet et al., 2012; Houlsby et al., 2014). In contrast to the numerical approaches, *OMF* makes weaker assumptions as the user preferences are no longer required to be interpreted as numbers, instead, only the ordering of items matters.

Despite of its effectiveness in modelling ordinal properties, *OMF* is incapable of exploiting the *local structure* described as follows. Typical collaborative filtering methods discover two types of information: the *neighbourhoods* and the *latent factors*, which we refer to as the *local* and the *global structures* of the preferences:

Local Structure The *local structure* (LS) refers to the second-order interactions between similar users or items. This type of information is often used by *neighbourhood-based* collaborative filtering, in which the predictions are made by looking at the neighbourhood of users (Resnick et al., 1994) or items (Sarwar et al., 2001). Though the majority of preferences will be ignored in making predictions, *LS*-based approaches are effective when the users/items correlations are highly localised.

Global Structure The *global structure* (GS) refers to the weaker but higher-order interactions among all users and items. This type of information is often used by *latent factor models* such as *SVD* (Koren et al., 2009) and *LDA* (Marlin, 2003), which aim at discovering the *latent factor spaces* in the preferences. *GS*-based approaches are often competitive in terms of accuracy as well as computational efficiency.

Exiting literature has suggested that the *LS* and the *GS* are complementary since they address different aspects of the preferences (Tran et al., 2009; Koren, 2008). In 2008, a unified framework has been proposed by Koren (Koren, 2008) to capture both structures, but only for numerical preferences. To the best of our knowledge, there is yet no method for the *OMF* to capture both the *LS* and the *GS*.

Recent advances in *Probabilistic Graphical Models*, especially the *Markov Random Fields* (MRF), have provided methods of building *RecSys* capable of exploiting both the *LS* and the *GS* (Tran et al., 2009). However, there has been little attempt to address the ordinal preferences issue due to the complication of modelling ordinal preferences with the *MRF*.

This paper aims to develop a unified model in which the *OMF* and the *MRF* are seamlessly combined to take advantages of both the representational power of the *MRF* and the ease of modelling ordinal preferences by the *OMF*. The proposed *Ordinal Random Fields* (ORF) model is not designed for a particular *OMF* but can incorporate any *OMF* model that produces ordinal distributions such as those in (Koren and Sill, 2011; Tran et al., 2012; Paquet et al., 2012; Houlsby et al., 2014). While this paper primarily focuses on exploiting the *LS*, the representational power of the *ORF* is by no mean limited to this. For example, the *MRF* employed in *ORF* can be extended to *Conditional Random Fields* (CRF) (Tran et al., 2007; Lafferty et al., 2001) to fuse auxiliary information such as the *item content* (Balabanović and Shoham, 1997) and *social relations* (Ma et al., 2011). These information has been shown helpful in making better recommendations (Basilico and Hofmann, 2004; Ma et al., 2011), and becomes even more valuable when the preferences data are highly sparse. Besides the extensibility, the *ORF* inherits other advantages of the probabilistic graphical models as well, such as supporting missing data by its nature, and disciplined learning and inferences techniques.

The remaining part of this paper is organised as follows. Section 2 reviews the basic concepts of the *Matrix Factorisation* and the *OMF* which form the basis of this work. Section 3 is devoted to the proposed *ORF* model. In Section 4, experimental results of the proposed *ORF* model are presented. Finally, Section 5 concludes this paper by summarising the main contributions and future works.

2. Preliminaries

Recommender Systems (RecSys) usually predict *users*' future interest in *items*. Let $\mathcal{U}$ and $\mathcal{I}$, denote the set of all *users* and the set of all *items*, respectively. The interest of the user $u \in \mathcal{U}$ in the item $i \in \mathcal{I}$ is encoded as the preference $r_{ui} \in R$, where the rating matrix R contains all known preferences.

Definition 1 (Recommender System) *RecSys aims to identify the item* $\hat{i} \in \mathcal{I}$ *that maximises the interest of the* target user $u \in \mathcal{U}$ *(Adomavicius and Tuzhilin, 2005)*

$$\hat{i} = \arg\max_{i \in \mathcal{I}} (r_{ui}) \tag{1}$$

In the rest of this section, we briefly review two *RecSys* approaches: *Matrix Factorisation* and *Ordinal Matrix Factorisation* that form a basis of this work For ease of reference, notations used throughout this paper are summarised in Table 1, and the term *preference* and *rating* will be used interchangeably.

2.1. Matrix Factorisation

Matrix Factorisation (MF) (Koren et al., 2009) is a popular and accurate approach to *RecSys*. This approach discovers the latent factor spaces shared between users and items, where the latent factors can be used to describe both the *taste* of users and the *characteristics* of items. The attractiveness of an item to a user is then measured by the inner product of their latent feature vectors.

Formally, each user u is associated with a latent feature vector $\mathbf{p}_u \in \mathbb{R}^k$ and each item i is associated with a latent feature vector $\mathbf{q}_i \in \mathbb{R}^k$, where k is the number of factors.

Table 1: Summary of Major Notations

Notations	Mathematical Meanings
$\mathcal{U}$	the set of all users
$\mathcal{I}$	the set of all items
R	the set of known preferences
$\mathcal{G}$	an undirected graph which encodes relations of preferences
$\mathcal{V}$	the set of vertices each represents a preference
$\mathcal{E}$	the set of edges each connects two vertices
$\mathbf{r}_u$	the set of all preferences by user u
f_{ij}	the correlation feature between items i and j
w_{ij}	the weight associated to the correlation feature f_{ij}
L	the number of rating levels, and the ratings are integers from 1 to L

The aim of *MF* is then to estimate $\hat{r}_{ui} = b_{ui} + \mathbf{p}_u^T \mathbf{q}_i$ such that $\hat{r}_{ui} \simeq r_{ui}$. The bias term $b_{ui} = \mu + b_u + b_i$ takes the biases into consideration, where μ is the overall average rating, b_u is the user bias, and b_i is the item bias. The latent feature vectors are learned by minimising regularised squared error with respect to all known preferences

$$\min_{\mathbf{p}_u, \mathbf{q}_i \in \mathbb{R}^k} \sum_{(u,i) \in R} (r_{ui} - b_{ui} - \mathbf{p}_u^T \mathbf{q}_i)^2 + \lambda(\|\mathbf{p}_u\|^2 + \|\mathbf{q}_i\|^2) \tag{2}$$

where λ is the regularisation coefficient. The optimisation can be done with *Stochastic Gradient Descent* for the favour of speed on sparse data, or with *Alternating Least Squares* for the favour of parallelization on dense data.

Comparing to *neighbour-based* approaches (Sarwar et al., 2001), *MF-based* approaches (Koren, 2008, 2010) have shown advantages in terms of accuracy and computational efficiency. Nevertheless, all of these approaches treat the preferences as numerical and are incapable of exploiting ordinal preferences.

2.2. Ordinal Matrix Factorisation

The ordinal nature of preferences has been overlooked in *RecSys* literature, until recently *Ordinal Matrix Factorisation* (OMF) (Koren and Sill, 2011; Tran et al., 2012; Paquet et al., 2012; Houlsby et al., 2014) has emerged to explore the ordinal properties of ratings.

In general, *OMF* aims to generate an ordinal distribution $Q(r_{ui}|u,i)$ over all possible rating values for each user/item pair. Predicting the rating for user u on item i is then equivalent to identifying the rating with the greatest mass in the ordinal distribution $Q(r_{ui}|u,i)$. While traditional *RecSys* approaches make only a point estimate, the *OMF* produces a full distribution and each prediction is associated with a probability as a *confidence* measure.

Typical *OMF* approaches assume the existence of a *latent utility* x_{ui} that captures how much the user u is interested in the item i. The latent utility x_{ui} can be defined in different ways (Koren and Sill, 2011; Tran et al., 2012; Paquet et al., 2012; Houlsby et al., 2014), but under the same framework of *Random Utility Models* (McFadden, 1980)

$$x_{ui} = \mu_{ui} + \epsilon_{ui} \tag{3}$$

where μ_{ui} is an internal score represents the interaction between the user u and the item i. The ϵ_{ui} is the random noise normally assumed to follow the logistic distribution in practice (Koren and Sill, 2011). The latent utility x_{ui} is then generated from a logistic distribution centred at μ_{ui} with the scale parameter s_{ui} proportional to the standard deviation

$$x_{ui} \sim Logi(\mu_{ui}, s_{ui}) \tag{4}$$

In collaborative filtering, the user-item interaction is often captured by *MF* techniques, thereby the internal score μ_{ui} can be substituted with the *MF* term $b_{ui} + \mathbf{p}_u^T \mathbf{q}_i$

$$x_{ui} = b_{ui} + \mathbf{p}_u^T \mathbf{q}_i + \epsilon_{ui} \tag{5}$$

where $\mathbf{p}_u$ and $\mathbf{q}_i$ are, respectively, the latent feature vectors of the user u and the item i. Modelling the latent utility with *MF* reflects the name *OMF*.

Despite how the latent utility is modelled, an *ordinal assumption* is required to convert the numerical utility into ordinal values. A common approach is the *ordinal logistic regression* originally described by McCullagh (McCullagh, 1980), which assumes that the rating is chosen based on the interval to which the utility belongs

$$r_{ui} = l \text{ if } x_{ui} \in (\theta_{l-1}, \theta_l] \text{ for } l < L \text{ and } r_{ui} = L \text{ if } x_{ui} > \theta_{L-1} \tag{6}$$

where L is the number of ordinal levels and θ_l are the threshold values of interest. Other assumptions (Mare, 1980) are also possible but McCullagh's model is by far the most popular. The probability of receiving a rating l is therefore

$$Q(r_{ui} = l|u, i) = \int_{\theta_{l-1}}^{\theta_l} P(x_{ui}|\theta) = F(\theta_l) - F(\theta_{l-1}) \tag{7}$$

where $F(\theta_l)$ is the cumulative logistic distribution evaluated at θ_l

$$F(x_{ui} \leq l|\theta_l) = \frac{1}{1 + \exp(-\frac{\theta_{uil} - \mu_{ui}}{s_{ui}})} \tag{8}$$

where the thresholds θ_l can be parameterised to depend on user or item. This paper employs the user-specific thresholds parameterisation described in Koren and Sill (2011). Therefore a set of thresholds $\{\theta_{ul}\}_{l=1}^{L}$ is defined for each user u to replace the thresholds θ_{uil} in Eq. 8.

Given the learned ordinal distribution $Q(r_{ui}|u, i)$, not only the ratings can be predicted but also the *confidence* for each prediction.

2.3. Summary

Matrix Factorisation has been one of the most popular *RecSys* approaches, which primarily focuses on numerical preferences such as ratings. Nevertheless, the nature of user preferences is often ordinal, and the importance of modelling ordinal properties has been recognised in recent works on *OMF* (Koren and Sill, 2011; Tran et al., 2012; Paquet et al., 2012; Houlsby et al., 2014). Although the *OMF* enables the modelling of ordinal properties, the employment of *MF* makes it only focuses on the higher-order interactions (the *GS*) regardless of the localised interactions (the *LS*), whereas both information are valuable (Koren, 2008;

Tran et al., 2009). Furthermore, the *OMF* by its nature cannot model auxiliary information such as *content* (Balabanović and Shoham, 1997) directly.

The powerful representation of *Markov Random Fields* (MRF) offers an opportunity to take advantages from all of these information, and have been developed in recent works (Tran et al., 2007, 2009). Nevertheless, exploiting the ordinal properties is not an easy task for *MRF* (Tran et al., 2009), therefore the strengths of the *OMF* and the *MRF* are nicely complementary. This observation leads to a naturally extension of unifying these two approaches, and motivates the present work.

3. Ordinal Random Fields

In this section, we propose the *Ordinal Random Fields* (ORF) to model the ordinal properties and capture both the *LS* and the *GS*. Here we exploit the *LS* of the item-item correlations only, while the user-user correlations can be modelled in a similar manner. The rest of this section introduces the concept of the *Markov Random Fields* followed a detailed discussion of the *ORF* including its feature design, parameter estimation, and predictions.

3.1. Markov Random Fields

Markov Random Fields (MRF) (Tran et al., 2007; Defazio and Caetano, 2012) models a set of random variables having Markov property with respect to an undirected graph $\mathcal{G}$. The undirected graph $\mathcal{G}$ consists a set of vertices $\mathcal{V}$ connected by a set of edges $\mathcal{E}$ without orientation, where two vertices are neighbourhood of each other when connected. Each vertex in $\mathcal{V}$ encodes a random variable, and the Markov property implies that a variable is conditionally independent of other variables given its neighbourhoods.

In this work, we use *MRF* to model user preferences and their relations respect to a set of undirected graphs. Specifically for each user u, there is a graph $\mathcal{G}_u$ with a set of vertices $\mathcal{V}_u$ and a set of edges $\mathcal{E}_u$. Each vertex in $\mathcal{V}_u$ represents a preference r_{ui} of user u on item i, and each edge in $\mathcal{E}_u$ captures a relation between two preferences by the same user.

As we consider only the item-item correlations in this work, two preferences are connected by an edge if and only if they are given by the same user. Fig. 1 shows an example of two graphs for users u and v. Note that vertices of different graphs are not connected directly, however, the edges between the same pair of items are associated to the same item-item correlation. For example, the edge between r_{ui} and r_{uj} and the edge between r_{vi} and r_{vj} are associated to the same item-item correlation between items i and j (see the green dashed line in Fig. 1).

Formally, let $\mathcal{I}(u)$ be the set of all items rated by user u and $\mathbf{r}_u = \{r_{ui} | i \in \mathcal{I}(u)\}$ be the joint set of all preferences (the variables) related to user u, then the *MRF* defines a distribution $P(\mathbf{r}_u)$ over the graph $\mathcal{G}_u$:

$$P(\mathbf{r}_u) = \frac{1}{Z_u} \Psi(\mathbf{r}_u) \tag{9}$$

$$\Psi(\mathbf{r}_u) = \prod_{(ui,uj) \in \mathcal{E}_u} \psi_{ij}(r_{ui}, r_{uj}) \tag{10}$$

where Z_u is the normalisation term that ensures $\sum_{\mathbf{r}_u} P(\mathbf{r}_u) = 1$, and $\psi(\cdot)$ is a positive function known as *potential*.

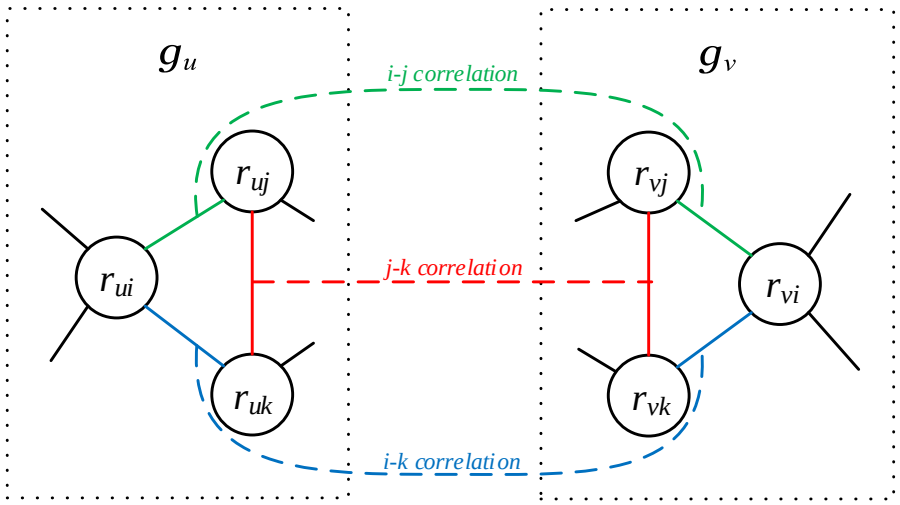


Figure 1: Example of undirected graphs for users u and v

The potential $\psi_{ij}(r_{ui}, r_{uj})$ captures the correlation between items i and j

$$\psi_{ij}(r_{ui}, r_{uj}) = \exp\{w_{ij} f_{ij}(r_{ui}, r_{uj})\} \tag{11}$$

where $f_{ij}(\cdot)$ is the feature function and w_{ij} is the corresponding weight. The correlation features capture the *LS*, while the weights realise the importance of each correlation feature. In *ORF*, the weights also control the relative importance between the *LS* and the *GS*. With the weights estimated from data, the unknown preference r_{ui} can be predicted as

$$\hat{r}_{ui} = \arg\max_{r_{ui}} P(r_{ui}|\mathbf{r}_u) \tag{12}$$

where $P(r_{ui}|\mathbf{r}_u)$ serves as the confidence measure of the prediction.

3.2. ORF: Unifying MRF and OMF

The standard *MRF* approach captures the *LS* by modelling item-item correlations under the framework of probabilistic graphical models. However, it employs the log-linear modelling as shown in Eq. 11, and therefore does not enable a simple treatment of ordinal preferences. *OMF*, on the other hand, can nicely model the ordinal preferences in a probabilistic way but is weak in capturing the *LS*. The complementary between these two techniques calls for the unified *ORF* model to take all of the advantages.

Essentially, the proposed *ORF* model promotes the agreement between the *GS* discovered by the *OMF* and the *LS* discovered by the *MRF*. More specifically, the *ORF* model combines the item-item correlations (Eq. 11) and the point-wise ordinal distribution $Q(r_{ui}|u, i)$ obtained from the *OMF* (Eq. 7)

$$P(\mathbf{r}_u) \propto \Psi_u(\mathbf{r}_u) \prod_{r_{ui} \in \mathbf{r}_u} Q(r_{ui}|u, i) \tag{13}$$

where $\Psi_u(\mathbf{r}_u)$ is the potential function capturing the interaction among items, and $\mathbf{r}_u$ is the set of preferences from user u.

The potential function $\Psi_u(\mathbf{r}_u)$ can be further factorised into pairwise potentials based on Eq. 11 and Eq. 10:

$$\Psi_u(\mathbf{r}_u) = \exp\left(\sum_{r_{ui}, r_{uj} \in \mathbf{r}_u} w_{ij} f_{ij}(r_{ui}, r_{uj})\right) \tag{14}$$

where $f_{ij}(\cdot)$ is the correlation feature between items i and j to be defined shortly in Section 3.3, and w_{ij} is the corresponding weight controls the relative importance of each correlation feature (LS) to the ordinal distribution (GS). Put all together, the joint distribution $P(\mathbf{r}_u)$ for each user u can be modelled as

$$P(\mathbf{r}_u) \propto \exp\left(\sum_{r_{ui}, r_{uj} \in \mathbf{r}_u} w_{ij} f_{ij}(r_{ui}, r_{uj})\right) \prod_{r_{ui} \in \mathbf{r}_u} Q(r_{ui}|u, i) \tag{15}$$

where there is a graph for each user but the weights are optimised by all users.

In fact, the user-user correlations can also be captured as

$$P(R) \propto \prod_i \Psi_i(\mathbf{r}_i) \prod_u \Psi_u(\mathbf{r}_u) \prod_{u,i} Q(r_{ui}|u, i) \tag{16}$$

but we limit our discussion to item-item correlations in this paper.

3.3. Feature Design

A feature is essentially a function f of $n > 1$ arguments that maps the (n-dimensional) input onto the unit interval $f : \mathbb{R}^n \to [0, 1]$, where the input can be ratings or auxiliary information such as *content* (Tran et al., 2007).

The item-item correlation is captured by the following feature

$$f(r_{ui}, r_{uj}) = g(|(r_{ui} - \bar{r}_i) - (r_{uj} - \bar{r}_j)|) \tag{17}$$

where $g(t) = 1/(1 + \mathrm{e}^{-t})$ does normalisation, and $\bar{r}_i$ and $\bar{r}_j$ are the average ratings for items i and j, respectively. This correlation feature captures the intuition that correlated items should receive similar ratings by the same user after offsetting the goodness of each item.

Though this work focuses on the item-item correlations, the feature for user-user correlations can be designed in a similar manner:

$$f(r_{ui}, r_{vi}) = g(|(r_{ui} - \bar{r}_u) - (r_{vi} - \bar{r}_v)|) \tag{18}$$

where $\bar{r}_u$ and $\bar{r}_v$ are the global average ratings for users u and v respectively.

Although the user and item bias have been modelled by the underlying *OMF*, the *ORF* itself can also model the bias with *identity features* for item i and for user u

$$f_i(r_{ui}, i) = g(|r_{ui} - \bar{r}_i|),\ f_u(r_{ui}, u) = g(|r_{ui} - \bar{r}_u|) \tag{19}$$

Indeed, auxiliary information such as *content* (Balabanović and Shoham, 1997) and *social relations* (Ma et al., 2011) can also be modelled by designing corresponding features. That being said, the *ORF* is a generic framework with great extensibility to integrate multiple sub-components such as neighbourhood, content, and ordinal ratings.

Nevertheless, this work focuses on the item-item correlation features only. Since one correlation feature exists for each possible pair of co-rated items, the number of correlation features can be large, and this makes the estimation slow to converge and less robust. Therefore we only keep the correlation features if strong correlation exists between two items i and j. Specifically, the *strong correlation features* are extracted based on the Pearson correlation and a user-specified *minimum correlation threshold*.

3.4. Parameter Estimation

In general, *MRF* models cannot be determined by standard maximum likelihood approaches, instead, approximation techniques such as *Markov Chain Monte Carlo* (MCMC) (Green, 1995) and *Pseudo-likelihood* (Besag, 1974) are often used in practice. The *pseudo-likelihood* leads to exact computation of the loss function and its gradient with respect to parameters, and thus faster. The MCMC-based methods may, on the other hand, lead to better estimation given enough time. As the experiments involve different settings and large number of features, this study employs the *pseudo-likelihood* technique to perform efficient parameter estimation by maximising the regularised sum of log local likelihoods

$$\mathcal{L}(\mathbf{w}) = \sum_{r_{ui} \in R} \log P(r_{ui} | \mathbf{r}_u \backslash r_{ui}) - \frac{1}{2\sigma^2} \sum_{u \in \mathcal{U}} \mathbf{w}_u^T \mathbf{w}_u \tag{20}$$

where σ is the regularisation coefficient, and $\mathbf{w}_u$ is the subset of weights related to user u.

The local likelihood is defined as

$$P(r_{ui} | \mathbf{r}_u \backslash r_{ui}) = \frac{1}{Z_{ui}} Q(r_{ui} | u, i) \exp \left(\sum_{r_{uj} \in \mathbf{r}_u \backslash r_{ui}} w_{ij} f_{ij}(r_{ui}, r_{uj}) \right) \tag{21}$$

where Z_{ui} is the normalisation term.

$$Z_{ui} = \sum_{r_{ui}=1}^{L} Q(r_{ui} | u, i) \exp \left(\sum_{r_{uj} \in \mathbf{r}_u \backslash r_{ui}} w_{ij} f_{ij}(r_{ui}, r_{uj}) \right) \tag{22}$$

To optimise the parameters, we use the stochastic gradient ascent procedure that updates the parameters by passing through the set of ratings of each user:

$$\mathbf{w}_u \leftarrow \mathbf{w}_u + \eta \nabla \mathcal{L}(\mathbf{w}_u) \tag{23}$$

where η is the learning rate. More specifically, for each r_{ui} and its neighbour r_{uj} in the set of ratings $\mathbf{r}_u$ by user u, update the weight w_{ij} using the gradient of the log pseudo-likelihood

$$\frac{\partial log \mathcal{L}}{\partial w_{ij}} = f_{ij}(r_{ui}, r_{uj}) - \sum_{r_{ui}=1}^{L} P(r_{ui} | \mathbf{r}_u \backslash r_{ui}) f_{ij}(r_{ui}, r_{uj}) \tag{24}$$

Algorithm 1 *Ordinal Random Fields* Algorithm

Require: the user preferences R; the ordinal distribution Q from Eq. 7.
Ensure: $\mathbf{w}$: the learned weights for correlation features.

Generate strong correlation features: $\mathbf{f}_{strong} \leftarrow \{f_{ij} | Pearson(i,j) \geq minCorr\}$
Initialise the weights: $\forall w_{ij} \in \mathbf{w}$, $w_{ij} \leftarrow \mathcal{N}(0, 0.01)$;
repeat
  for each $u \in \mathcal{U}$ **do**
    for each $r_{ui}, r_{uj} \in \mathbf{r}_u$, $i \neq j$ **do**
      if $f_{ij} \in \mathbf{f}_{strong}$ **then**
        Compute correlation feature f_{ij} according to Eq. 17
        Compute normalisation term Z_{ui} according to Eq. 22
        Compute local likelihood according to Eq. 21
        Compute the gradient for weight w_{ij} according to Eq. 24
        Update w_{ij} with the gradient $w_{ij} \leftarrow w_{ij} + \eta \nabla \mathcal{L}(w_{ij})$
      end if
    end for
  end for
until convergence
return $\mathbf{w}$;
Predictions:
(1) Predict most likely rating with confidence measure using Eq. 26
(2) Predict expectation using Eq. 25

3.5. Preference Prediction

The prediction of rating r_{ui} is straightforward, which can be done by identifying the rating with the greatest mass in local likelihood:

$$\hat{r}_{ui} = \arg\max_{r_{ui}} P(r_{ui} | \mathbf{r}_u) \tag{25}$$

where the local likelihood is given by Eq. 21. Prediction made in this approach identifies the most likely rating from discrete values 1 to L, and the local likelihood serves as a *confidence* measure. For predictions of scalar values, the expectation can be used instead:

$$\hat{r}_{ui} = \sum_{r_{ui}=1}^{L} r_{ui} P(r_{ui} | \mathbf{r}_u) \tag{26}$$

Finally, Alg. 1 summarises the learning and prediction procedures for the *ORF*.

4. Experiment and Analysis

To study the performance of the proposed *ORF* model, comparisons were made with the following representative algorithms: *a*) *K-Nearest Neighbours* (K-NN) (Resnick et al., 1994; Sarwar et al., 2001), which represents the methods exploiting the *LS*; *b*) *OMF* (Koren and Sill, 2011), which exploits the *GS* and ordinal properties; *c*) and finally the *ORF* model,

which takes ordinal properties into account and exploits both the *LS* and the *GS*. Details of the experimental settings and results are presented in this section.

4.1. Experimental Settings

Datasets Experiments were conducted on two public movie rating datasets: the MovieLens-100K and the MovieLens-1M [1] datasets. The MovieLens-1M dataset contains roughly 1 million ratings by 6040 users on 3900 movies. The MovieLens-100K dataset contains 100K ratings by 943 users on 1682 movies. Ratings are on the $1-5$ scale.

To perform a reliable evaluation, we keep only users who rated at least 30 movies, and each dataset is shuffled and split into the disjoint training set, validation set and test set. For each user, 5 ratings are kept in the validation set for tuning the hyper-parameters, 10 ratings are reserved for testing, and the rest for training.

Evaluation Metric The *Mean Absolute Error* (MAE) and the *Root Mean Square Error* (RMSE) are used as the evaluation metric

$$MAE = \frac{1}{|R_{test}|} \sum_{(u,i)\in R_{test}} |\hat{r}_{ui} - r_{ui}|,\ RMSE = \sqrt{\frac{\sum_{(u,i)\in R_{test}} (\hat{r}_{ui} - r_{ui})^2}{|R_{test}|}} \tag{27}$$

where R_{test} is the test set kept aside until all parameters have been tuned. A smaller *MAE* or *RMSE* value indicates better performance. Although both metrics are used, we consider the *MAE* metric to be more suitable for ordinal preferences. The reason is that it makes more scenes to consider being off by 4 is just twice as bad as being off by 2 when the preferences are ordinal. The *RMSE* metric, on the other hand, can be skewed to methods that are optimised for numerical preferences.

Parameters To perform a fair comparison, we fix the number of latent factors to the typical value of 50 for all algorithms, and all weights are randomly initialised from $\mathcal{N}(0, 0.01)$. The number of neighbours for *K-NN* algorithms is set to 10 and 30. The minimum correlation threshold for the *ORF* is set to reasonable values considering both the prediction performance and computational efficiency. We will also report the effect of varying the minimum correlation threshold.

4.2. Results

We first compare the performance of the proposed *ORF* model with related algorithms: user-based *K-NN*, item-based *K-NN* and *OMF*, where the *OMF* is the targeted baseline. Then the impact of parameters is investigated for the *ORF* model, in particular the regularisation coefficient and the minimum correlation threshold.

4.2.1. Comparison with Other Methods

The comparison results in terms of prediction accuracy are shown in Table 2. The *global average* is used only as a benchmark, which uses the average rating as the predictions. The following observations can be made based on the results.

1. http://grouplens.org/datasets/movielens

Table 2: For both the *OMF* and the *ORF*, the expectation values (Eq. 26) are used for *RMSE* and the most likely values (Eq. 25) are used for *MAE*.

	MovieLens-100K		MovieLens-1M	
Method	RMSE	MAE	RMSE	MAE
Global Ave.	1.1186	0.9430	1.1123	0.9401
UserKNN, K=10	0.9687	0.7584	0.9350	0.7328
ItemKNN, K=10	0.9372	0.7305	0.9032	0.7041
UserKNN, K=30	0.9463	0.7413	0.9149	0.7173
ItemKNN, K=30	0.9315	0.7295	**0.8987**	0.70478
OMF	0.9525	0.7226	0.9144	0.6918
ORF, minCorr=0.4	0.9475	0.7185	0.9117	0.6887
ORF, minCorr=0.3	0.9448	**0.7148**	**0.9093**	**0.6870**

Firstly, the *K-NN* methods, especially the item-based *K-NN*, perform quite well. As the *K-NN* methods exploit only the *LS*, this result indicates the effectiveness of the *LS*. However, the ignorance of the *GS* makes the *K-NN* methods not less generalised and thus highly susceptible to noisy data.

Secondly, the *OMF* fits the data quite well when predicting the most likely ratings for the *MAE* metric. However, it exploits only the *GS* and therefore further improvements are possible by incorporating the *LS* information.

Finally, the *ORF* has made further improvements upon the *OMF* by unifying the modelling of both the *LS* and the *GS*, as well as ordinal properties. Note that the performance of the *ORF* relies on the ordinal distributions generated by the underlying *OMF*, which can be implemented in different ways (Koren and Sill, 2011; Tran et al., 2012; Paquet et al., 2012; Houlsby et al., 2014). In present work, the improvements over the *OMF* are soley based on incorporating the *LS* information.

Table 3: Paired t-test for the *ORF* and the *OMF*.

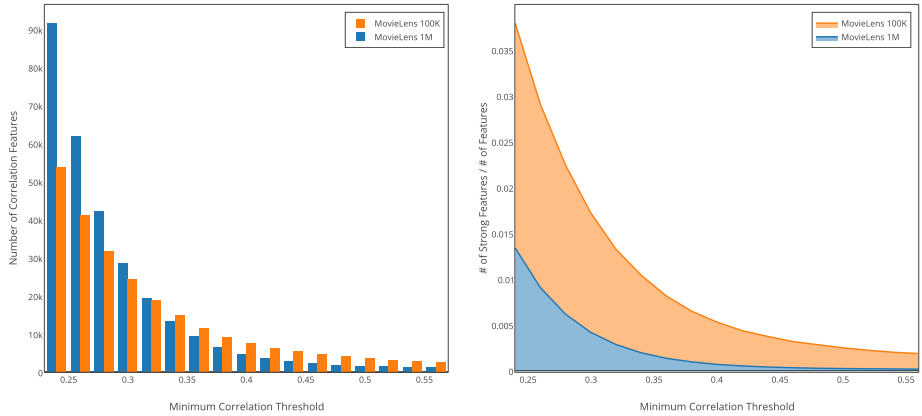
	t-test statistics		
Methods	df	t	p-value
ORF vs. OMF on MAE	9	6.0163	0.0002
ORF vs. OMF on RMSE	9	4.8586	0.0009

To confirm the improvements, a paired t-test (two-tailed) with a confidence level of 95% has been applied to the *ORF* and the *OMF*. Results shown in Table 3 confirm that the performance of methods with and without capturing the *LS* is statistically significant.

4.2.2. Impact of Minimum Correlation Threshold

As mentioned in Section 3.3, the *ORF* model requires a minimum correlation threshold to control the number of correlation features. The reason is that the number of correlation

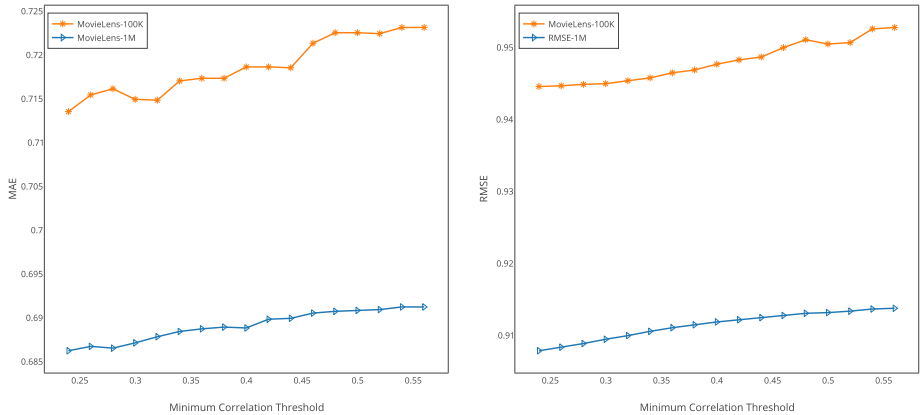
features can be very large, which makes the model less robust and slow to converge. Specifically, when this threshold goes to minimum (e.g. -1 for Pearson correlation), the potential number of correlation features can be as large as $n^2/2$ where n is the number of items. On the other hand, the number of correlation features goes to zero when the threshold goes to maximum, and the *ORF* reduces to the *OMF*.



(a) Number of Correlation Features (b) Coverage of Correlation Features

Figure 2: Impact of Minimum Correlation Threshold on Number of Correlation Features

Fig. 2(a) shows the number of correlation features for different minimum correlation thresholds. Given that the MovieLens-100K dataset contains less items comparing the MovieLens-1M dataset, there are even more correlation features remained in the MovieLens-100K dataset when the threshold becomes larger. This observation implies that the items in the MovieLens-100K dataset are more correlated with each other. We also show the coverage of correlation features for both datasets, and the MovieLens-100K has consistently higher coverage of correlation features.



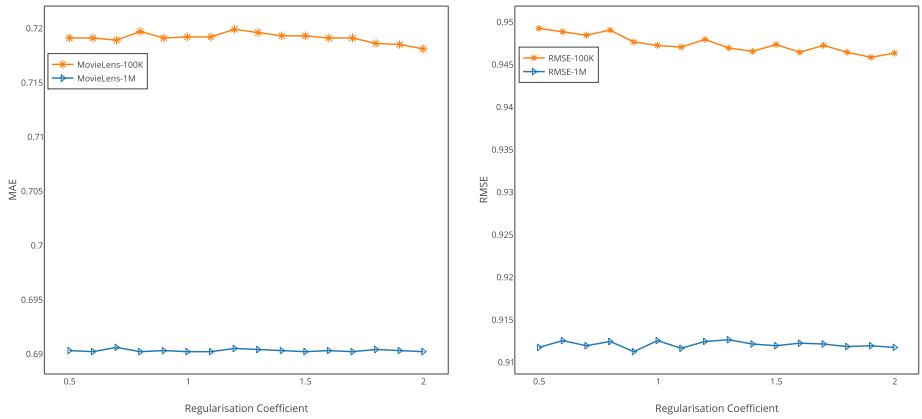
(a) MAE (b) RMSE

Figure 3: Impact of Minimum Correlation Threshold

Having these statistics result, we further examine the impact of the minimum correlation threshold on prediction accuracy, as plotted in Fig. 3. It can be observed that the prediction accuracy improves as the minimum correlation threshold becomes smaller. However, we notice that the performance on the smaller MovieLens-100K dataset is not as stable as that on the MovieLens-1M dataset, where the curve of the MovieLens-1M dataset is smoother and shows better monotonicity. One explanation is that the MovieLens-100K dataset may not have enough data to make robust estimation for large number of weights. However, given adequate data and time, the best prediction performance can be achieved by including all correlation features, i.e., the minimum correclation threshold is set to minimum.

4.2.3. Impact of Regularisation Coefficient

While the number of correlation features can be large, the model might be prone to over-fitting. Therefore we investigate the impact of varying the the regularisation coefficient. Fig. 4 shows the performance of the *ORF* under different regularisation settings. We observe



(*a*) MAE (*b*) RMSE

Figure 4: Impact of Regularisation Coefficient

that by varying the regularisation coefficient the prediction performance was not affected too much. One possible explanation is that the ordinal distribution employed in the *ORF* is generated by the underlying *OMF* with its own regularisation mechanism, whereas the regularisation term in the *ORF* controls only the weights for the second-order item-item correlation features. In other words, the *ORF* model by itself is unlikely to over-fit the data given that the underlying *OMF* model has been properly regularised.

5. Conclusions and Future Works

In this paper we presented the *ORF* model that takes advantages of both the representational power of the *MRF* and the ease of modelling ordinal properties by the *OMF*. While the standard *OMF* approaches exploit only the *GS*, the *ORF* captures the *LS* as well. In addition, the *ORF* model defines a uniformed interface for different *OMF* methods with various internal implementations. Last but not least, the *ORF* model is a generic framework that can be extended to incorporate additional information by designing more features.

A future extension could take the user-user correlations into account as we modelled only the item-item correlations in this work. Incorporating the user-user correlations may further improve the prediction performance. Another future work is to take *auxiliary information* into consideration by replacing the *MRF* with the *Conditional Random Fields* (Lafferty et al., 2001). Fusing *auxiliary information* such as the *item content* and *social relations* could improve the prediction performance especially when the data is highly sparse.

Acknowledgement

This work was partially supported by the National Science Foundation of China (61273301) and the Collaborative Innovation Center of Novel Software Technology and Industrialization.

References

G. Adomavicius and A. Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17(6):734–749, 2005.

M. Balabanović and Y. Shoham. Fab: content-based, collaborative recommendation. *Communications of the ACM*, 40(3):66–72, 1997.

J. Basilico and T. Hofmann. Unifying collaborative and content-based filtering. In *Proceedings of the ICML'04*, pages 65–72. ACM, 2004.

J. Besag. Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 192–236, 1974.

A. Brun, A. Hamad, O. Buffet, and A. Boyer. Towards preference relations in recommender systems. In *Preference Learning (PL 2010) ECML/PKDD 2010 Workshop*, 2010.

A. Defazio and T. Caetano. A graphical model formulation of collaborative filtering neighbourhood methods with fast maximum entropy training. In *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*, pages 265–272, 2012.

M. S. Desarkar, S. Sarkar, and P. Mitra. Aggregating preference graphs for collaborative rating prediction. In *Proceedings of the RecSys'10*, pages 21–28. ACM, 2010.

P. J. Green. Reversible jump markov chain monte carlo computation and bayesian model determination. *Biometrika*, 82(4):711–732, 1995.

N. Houlsby, J. M. Hernández-Lobato, and Z. Ghahramani. Cold-start active learning with robust ordinal matrix factorization. pages 766–774, 2014.

Y. Koren. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 426–434. ACM, 2008.

Y. Koren. Collaborative filtering with temporal dynamics. *Communications of the ACM*, 53(4):89–97, 2010.

Y. Koren and J. Sill. Ordrec: an ordinal model for predicting personalized item rating distributions. In *Proceedings of the RecSys'11*, pages 117–124. ACM, 2011.

Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *IEEE Computer*, 42(8):30–37, 2009.

J. D. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, pages 282–289, 2001.

H. Ma, D. Zhou, C. Liu, M. R. Lyu, and I. King. Recommender systems with social regularization. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 287–296. ACM, 2011.

R. D. Mare. Social background and school continuation decisions. *Journal of the American Statistical Association*, 75(370):295–305, 1980.

B. M. Marlin. Modeling user rating profiles for collaborative filtering. In *Advances in neural information processing systems*. MIT Press, 2003.

P. McCullagh. Regression models for ordinal data. *Journal of the Royal Statistical Society, Series B*, 42(2):109–142, 1980.

D. McFadden. Econometric models for probabilistic choice among products. *Journal of Business*, 53(3):S13–S29, 1980.

U. Paquet, B. Thomson, and O. Winther. A hierarchical model for ordinal matrix factorization. *Statistics and Computing*, 22(4):945–957, 2012.

P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl. Grouplens: an open architecture for collaborative filtering of netnews. In *Proceedings of the 1994 ACM conference on Computer supported cooperative work*, pages 175–186. ACM, 1994.

B. Sarwar, G. Karypis, J. Konstan, and J. Riedl. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the WWW'10*, pages 285–295. ACM, 2001.

A. Sharma and B. Yan. Pairwise learning in recommendation: experiments with community recommendation on linkedin. In *Proceedings of RecSys'13*, pages 193–200. ACM, 2013.

T. Tran, D. Q. Phung, and S. Venkatesh. Preference networks: Probabilistic models for recommendation systems. In *Proceedings of the 6th Australasian Data Mining Conference (AusDM'07)*, pages 195–202. Australian Computer Society, 2007.

T. Tran, D. Q. Phung, and S. Venkatesh. Ordinal boltzmann machines for collaborative filtering. In *Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence*, pages 548–556. AUAI Press, 2009.

T. Tran, D. Q. Phung, and S. Venkatesh. A sequential decision approach to ordinal preferences in recommender systems. In *Proceedings of the AAAI'12*, 2012.