Metadata of the chapter that will be visualized in SpringerLink

Book Title	Advances in Soft Computing	
Series Title		
Chapter Title	Infrequent Item-to-Item Recommendation via Invariant Random Fields	
Copyright Year	2018	
Copyright HolderName	Springer Nature Switzerland AG	
Corresponding Author	Family Name	**Daróczy**
	Particle	
	Given Name	**Bálint**
	Prefix	
	Suffix	
	Role	
	Division	Institute for Computer Science and Control
	Organization	Hungarian Academy of Sciences (MTA SZTAKI)
	Address	Budapest, 1111, Hungary
	Email	daroczyb@ilab.sztaki.hu
Author	Family Name	**Ayala-Gómez**
	Particle	
	Given Name	**Frederick**
	Prefix	
	Suffix	
	Role	
	Division	Faculty of Informatics
	Organization	Eötvös Loránd University
	Address	Pázmány P. sny. 1/C., Budapest, 1117, Hungary
	Email	fayala@caesar.elte.hu
Author	Family Name	**Benczúr**
	Particle	
	Given Name	**András**
	Prefix	
	Suffix	
	Role	
	Division	Institute for Computer Science and Control
	Organization	Hungarian Academy of Sciences (MTA SZTAKI)
	Address	Budapest, 1111, Hungary
	Email	benczur@sztaki.mta.hu

Abstract

Web recommendation services bear great importance in e-commerce and social media, as they aid the user in navigating through the items that are most relevant to her needs. In a typical web site, long history of previous activities or purchases by the user is rarely available. Hence in most cases, recommenders propose items that are similar to the most recent ones viewed in the current user session. The corresponding task is called session based item-to-item recommendation. Generating item-to-item recommendations by “people who viewed this, also viewed” lists works fine for popular items. These recommender systems rely on

item-to-item similarities and item-to-item transitions for building next-item recommendations. However, the performance of these methods deteriorates for rare (i.e., infrequent) items with short transaction history. Another difficulty is the cold-start problem, items that recently appeared and had no time yet to accumulate a sufficient number of transactions. In this paper, we describe a probabilistic similarity model based on Random Fields to approximate item-to-item transition probabilities. We give a generative model for the item interactions based on arbitrary distance measures over the items including explicit, implicit ratings and external metadata. We reach significant gains in particular for recommending items that follow rare items. Our experiments on various publicly available data sets show that our new model outperforms both simple similarity baseline methods and recent item-to-item recommenders, under several different performance metrics.

Keywords (separated by '-') Recommender systems - Fisher information - Markov random fields

Infrequent Item-to-Item Recommendation via Invariant Random Fields

Bálint Daróczy[1(✉)], Frederick Ayala-Gómez[2], and András Benczúr[1]

[1] Institute for Computer Science and Control,
Hungarian Academy of Sciences (MTA SZTAKI), Budapest 1111, Hungary
daroczyb@ilab.sztaki.hu, benczur@sztaki.mta.hu
[2] Faculty of Informatics, Eötvös Loránd University,
Pázmány P. sny. 1/C., Budapest 1117, Hungary
fayala@caesar.elte.hu

Abstract. Web recommendation services bear great importance in e-commerce and social media, as they aid the user in navigating through the items that are most relevant to her needs. In a typical web site, long history of previous activities or purchases by the user is rarely available. Hence in most cases, recommenders propose items that are similar to the most recent ones viewed in the current user session. The corresponding task is called session based item-to-item recommendation. Generating item-to-item recommendations by "people who viewed this, also viewed" lists works fine for popular items. These recommender systems rely on item-to-item similarities and item-to-item transitions for building next-item recommendations. However, the performance of these methods deteriorates for rare (i.e., infrequent) items with short transaction history. Another difficulty is the cold-start problem, items that recently appeared and had no time yet to accumulate a sufficient number of transactions. In this paper, we describe a probabilistic similarity model based on Random Fields to approximate item-to-item transition probabilities. We give a generative model for the item interactions based on arbitrary distance measures over the items including explicit, implicit ratings and external metadata. We reach significant gains in particular for recommending items that follow rare items. Our experiments on various publicly available data sets show that our new model outperforms both simple similarity baseline methods and recent item-to-item recommenders, under several different performance metrics.

AQ1

Keywords: Recommender systems · Fisher information
Markov random fields

1 Introduction

Recommender systems [26] have become common in a variety of areas including movies, music, videos, news, books, and products in general. They produce a list of recommended items by either collaborative or content based filtering.

I. Batyrshin et al. (Eds.): MICAI 2018, LNAI 11288, pp. 1–19, 2018.
https://doi.org/10.1007/978-3-030-04491-6_20

Collaborative filtering methods [19,27,31] build models of the past user-item interactions, while content based filtering [20] typically generates lists of similar items based on item properties. To assess the attitude towards the items viewed by the user, recommender systems rely on users explicit feedback (e.g., ratings, like/dislike) or implicit feedback (e.g., clicks, plays, views).

The Netflix Prize Challenge [3,17] revolutionized our knowledge of recommender systems but biased research towards the case where user profiles and item ratings (1–5 stars) are known. However, for most Web applications, users are reluctant to create logins and prefer to browse anonymously. Or, we purchase certain types of goods (e.g., expensive electronics) so rarely that our previous purchases will be insufficient to create a meaningful user profile. Several practitioners [16] argue that most of the recommendation tasks they face are implicit feedback and without sufficient user history. In [23] the authors claim that 99% of the recommendations systems they built for industrial application tasks are implicit, and most of them are item-to-item. For these cases, recommender systems rely on the recent items viewed by the user in the actual shopping session.

In this paper, we consider the task of recommending relevant items to a user based on items seen during the current session [19,27] rather than on user profiles. Best known example of this task is the Amazon list of books related to the last visited one [19]. An intuitive approach to building the list of relevant items to recommend is to consider item pair frequencies. However, for rare items, it is necessary to use global similarity data to avoid recommendations based on low support. In addition, we have to devise techniques that handle new items well. In the so-called cold start case [28], the new items have yet insufficient number of interactions to reliably model their relation to the users.

Our key idea is to utilize the known, recent or popular items for item-to-item recommendation via multiple representations. The starting point of our method is the idea of [16], to utilize the entire training data and not just the item-item conditional probabilities. Our item-to-item model is able to use single or combined similarity measures such as Jaccard or cosine based on collaborative, content, multimedia and metadata information.

We evaluate the top-n recommendation [6] performance of our models by Recall and DCG. We present our proposed approach in Sects. 3, 4, and 6. The experimental results are presented in Sect. 7.

2 Related Work

Recommender systems are surveyed in [26]. Several recommender systems consider a setting similar to the Netflix Prize Competition [3], where users and their explicit feedback (1–5 stars) are given, and the task is to predict unseen ratings. In this paper, we consider cases where users do not give explicit ratings, and we have to infer their preferences from their implicit feedback [16]. And, we assume that a rich user history is not available, so we rely on the present items of the user's session.

The first item-to-item recommender methods [19,27] used similarity information to find nearest neighbor transactions [7]. Another solution is to extract

association rules [5]. Both classes of these methods deteriorate if the last item of the session has a low item transition support (e.g., rare or recent items). Nearest neighbor methods were criticized for two reasons. First, the similarity metrics typically have no mathematical justification. Second, the confidence of the similarity values is often not involved when finding the nearest neighbor, which leads to overfitting in sparse cases. In [18], a method is given that learns similarity weights for users, however the method gives global and not session based user recommendation.

Rendle et al. [24] proposed a session-based recommender system that models the users by factorizing personal Markov chains. Their method is orthogonal to ours in that they provide more accurate user based models if more data is available, while we concentrate on extracting actionable knowledge from the entire data for the sparse transactions in a session.

The item-to-item recommendation can be considered a particular context-aware recommendation problem. In [10] sequentiality as a context is handled by using pairwise associations as features in an alternating least squares model. They mention that they face the sparsity problem in setting minimum support, confidence, and lift of the associations, and they used the category of last purchased item as a fallback. In a follow-up result [11], they use the same context-aware ALS algorithm. However, they only consider seasonality as a context in that paper.

Closest to our work is the *Euclidean Item Recommender* (`EIR`) [16] by Koenigstein and Koren. They model item-to-item transitions using item latent factors where the Euclidean distance between two vectors approximates the known transition probabilities in the training dataset. Our model differs in that we do not need to optimize a vector space to learn the transition probabilities in a lower dimensional space. Instead, we start from an arbitrary similarity definition, and we may extend similarity for all items, by using all training data, in a mathematically justified way. We use Fisher information, that is applied for DNA splice site classification [13] and computer vision [22], but we are the first to apply it in recommender systems. We applied – to the most extent reproducible – the experimental settings of `EIR`.

3 Similarity Graph

The starting point of our item-to-item recommender model is a set of arbitrary item pair similarity measures, which may be based on implicit or explicit user feedback, user independent metadata such as text description, linkage or even multimedia content. By the pairwise similarity values and potentially other model parameters θ, we model item i as a random variable $p(i|\theta)$. From $p(i|\theta)$, we will infer the distance and the conditional probability of pairs of items i and j by using all information in θ.

Formally, let us consider a certain sample of items $S = \{i_1, i_2, \ldots, i_N\}$ (e.g., most popular or recent items), and assume that we can compute the distance of any item i from each of $i_n \in S$. We will consider our current item i along with its

distance from each $i_n \in S$ as a random variable generated by a Markov Random Field (MRF). Random fields are a set of (dependent) random variables. In case of MRF the connection between the elements is described by an undirected graph satisfying the Markov property [4]. For example, the simplest Markov Random Field can be obtained by using a graph with edges between item i and items $i_n \in S$, as shown in Fig. 1.

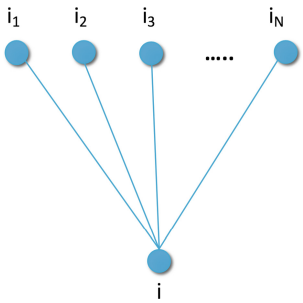

Fig. 1. Similarity graph of item i with sample items $S = \{i_1, i_2, ..., i_N\}$ of distances $\text{dist}(i, i_n)$ from i.

Let us assume that we are given a Markov Random Field generative model for $p(i|\theta)$. By the Hammersley-Clifford theorem [9], the distribution of $p(i|\theta)$ is a Gibbs distribution, which can be factorized over the maximal cliques and expressed by a potential function U over the maximal cliques as follows:

$$p(i \mid \theta) = \mathrm{e}^{-U(i|\theta)}/Z(\theta), \tag{1}$$

where $U(i \mid \theta)$ is the energy function and

$$Z(\theta) = \sum_i \mathrm{e}^{-U(i|\theta)}$$

is the sum of the exponent of the energy function over our generative model, a normalization term called the partition function. If the model parameters are previously determined, then $Z(\theta)$ is a constant.

Given a Markov Random Field defined by a certain graph such as the one in Fig. 1 (or some more complex graph defined later), a wide variety of proper energy functions can be used to define a Gibbs distribution. The weak but necessary restrictions are that the energy function has to be positive real valued, additive over the maximal cliques of the graph, and more probable parameter configurations have to have lower energy.

Given a finite sample set $S = \{i_1, .., i_N\}$, we define the simplest similarity graph as seen in Fig. 1 by describing the energy function for (1) as

$$U(i \mid \theta = \{\alpha_1, .., \alpha_N\}) := \sum_{n=1}^{N} \alpha_n \text{dist}(i, i_n), \tag{2}$$

where dist is an arbitrary distance or divergence function of item pairs and the hyperparameter set θ is the weight of the elements in the sample set.

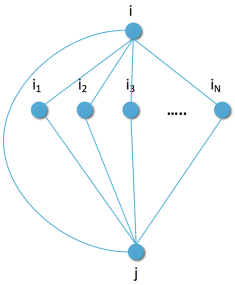

Fig. 2. Pairwise similarity graph with sample set $S = \{i_1, i_2, ..., i_N\}$ for a pair of items i and j.

In a more complex model, we capture the connection between pairs of items by extending the generative graph model with an additional node for the previous item as shown in Fig. 2. In the pairwise similarity graph, the maximal clique size increases to three. To capture the joint energy with parameters $\theta = \{\beta_n\}$, we can use a heuristic approximation similar to the pseudo-likelihood method [4]: we approximate the joint distribution of each size three clique as the sum of the individual edges by

$$U(i, j \mid \theta) := \sum_{n=1}^{N} \beta_n (\mathrm{dist}(i, i_n) + \mathrm{dist}(j, i_n) + \mathrm{dist}(i, j)), \tag{3}$$

At first glance, the additive approximation seems to oversimplify the clique potential and falls back to the form of Eq. (2). However, the effect of the clique is apparently captured by the common clique hyperparameter β_n, as also confirmed by our experiments.

4 Fisher Information

Items with low support usually cannot be captured by traditional similarity models. To handle the similarity of rare items, in this section we introduce the Fisher information to estimate distinguishing properties by using the similarity graphs.

Let us consider a general parametric class of probability models $p(i|\theta)$, where $\theta \in \Theta \subseteq \mathbb{R}^\ell$. The collection of models with parameters from a general hyperparameter space Θ can then be viewed as a (statistical) manifold M_Θ, provided that the dependence of the potential on Θ is sufficiently smooth. By [15], M_Θ can be turned into a Riemann manifold by giving an inner product (kernel) at

the tangent space of each point $p(i|\theta) \in M_\Theta$, where the inner product varies smoothly with p.

The notion of the inner product over $p(i|\theta)$ allows us to define the so-called Fisher metric on M. The fundamental result of Čencov [29] states that the Fisher metric exhibits a unique invariance property under some maps which are quite natural in the context of probability. Thus, one can view the use of Fisher kernel as an attempt to introduce a natural comparison of the items on the basis of the generative model [13].

We start defining the Fisher kernel over the manifold M_Θ of probabilities $p(i|\theta)$ as in Eq. (1) by considering the tangent space. The tangent vector

$$G_i = \nabla_\theta \log p(i|\theta) = \left(\frac{\partial}{\partial \theta_1} \log p(i|\theta), \ldots, \frac{\partial}{\partial \theta_l} \log p(i|\theta) \right) \tag{4}$$

is called the *Fisher score* of item i. The *Fisher information matrix* is a positive semidefinite matrix defined as

$$F(\theta) := \mathbf{E}_\theta (\nabla_\theta \log p(i|\theta) \nabla_\theta \log p(i|\theta)^T), \tag{5}$$

where the expectation is taken over $p(i|\theta)$. In particular, the nm-th entry of $F(\theta)$ is

$$F_{nm} = \sum_i p(i|\theta) \left(\frac{\partial}{\partial \theta_n} \log p(i|\theta) \right) \left(\frac{\partial}{\partial \theta_m} \log p(i|\theta) \right).$$

Thus, to capture the generative process, the gradient space of M_Θ is used to derive the Fisher vector, a mathematically grounded feature representation of item i. The corresponding kernel function

$$K(i,j) := G_i^T F^{-1} G_j \tag{6}$$

is called the *Fisher kernel*. An intuitive interpretation is that G_i gives the direction where the parameter vector θ should be changed to fit item i the best [22]. In addition, we prove a theorem for our kernels on a crucial reparametrization invariance property that typically holds for Fisher kernels [25].

Theorem 1. *For all $\theta = \rho(\mu)$ for a continuously differentiable function ρ, K_θ is identical.*

Proof. The Fisher score is

$$G_i(\mu) = G_i(\rho(\mu)) \left(\frac{\partial \rho}{\partial \mu} \right)$$

and therefore

$$\begin{aligned} K_\mu(i,j) &= G_i(\mu) F_\mu^{-1} G_j(\mu) \\ &= G_i(\rho(\mu)) \left(\frac{\partial \rho}{\partial \mu} \right) \left(F_{\rho(\mu)} \left(\frac{\partial \rho}{\partial \mu} \right)^2 \right)^{-1} G_j(\rho(\mu)) \left(\frac{\partial \rho}{\partial \mu} \right) \\ &= G_i(\rho(\mu)) F_{\rho(\mu)}^{-1} G_j(\rho(\mu)) = K_\rho(i,j). \end{aligned}$$

□

Essentially, the theorem states that the kernel will not depend on the hyperparameters θ.

5 Item Similarity by Fisher Information

Based on the similarity graphs introduced in Sect. 3 and by taking advantage of the invariance properties of the Fisher metric, we propose two ranking methods for item-item transitions.

5.1 Item-Item Fisher Conditional Score (FC)

Our first item-to-item recommender method will involve similarity information in the item-item transition conditional probability computation by using Fisher scores as in Eq. (4). By the Bayes theorem,

$$\begin{aligned} G_{j|i} &= \nabla_\theta \log p(j \mid i;\theta) = \nabla_\theta \log \frac{p(i,j \mid \theta)}{p(i \mid \theta)} \\ &= \nabla_\theta \log p(i,j \mid \theta) - \nabla_\theta \log p(i \mid \theta), \end{aligned} \tag{7}$$

thus we need to determine the joint and the marginal distributions for a particular item pair.

First, let us calculate the Fisher score of (4) with $p(i|\theta)$ of the single item generative model defined by (2),

$$\begin{aligned} G_i^k(\theta) &= \nabla_{\theta_k} \log p(i|\theta) \\ &= \frac{1}{Z(\theta)} \sum_i \mathrm{e}^{-U(i|\theta)} \frac{\partial U(i \mid \theta)}{\partial \theta_k} - \frac{\partial U(i \mid \theta)}{\partial \theta_k} \\ &= \sum_i \frac{\mathrm{e}^{-U(i|\theta)}}{Z(\theta)} \frac{\partial U(i \mid \theta)}{\partial \theta_k} - \frac{\partial U(i \mid \theta)}{\partial \theta_k}. \end{aligned}$$

By (1), our formula can be simplified as

$$\begin{aligned} G_i^k(\theta) &= \sum_i p(i \mid \theta) \frac{\partial U(i \mid \theta)}{\partial \theta_k} - \frac{\partial U(i \mid \theta)}{\partial \theta_k} \\ &= \mathbf{E}_\theta[\tfrac{\partial (U(i|\theta)}{\partial \theta_k}] - \tfrac{\partial U(i|\theta)}{\partial \theta_k}. \end{aligned} \tag{8}$$

For an energy function as in Eq. (2), the Fisher score of i has a simple form,

$$G_i^k(\theta) = \mathbf{E}_\theta[\mathrm{dist}(i,i_k)] - \mathrm{dist}(i,i_k), \tag{9}$$

and similarly for Eq. (3),

$$\begin{aligned} G_{ij}^k(\theta) = \mathbf{E}_\theta[&\mathrm{dist}(i,i_k) + \mathrm{dist}(j,i_k) + \mathrm{dist}(i,j)] \\ &-(\mathrm{dist}(i,i_k) + \mathrm{dist}(j,i_k) + \mathrm{dist}(i,j)). \end{aligned} \tag{10}$$

Now, if we put (9) and (10) into (7), several terms cancel out and the Fisher score becomes

$$G_{j|i}^k = \mathbf{E}_\theta[\mathrm{dist}(j,i_k) + \mathrm{dist}(i,j)] - (\mathrm{dist}(j,i_k) + \mathrm{dist}(i,j)).$$

The above formula involves the distance values on the right side, which are readily available, and the expected values on the left side, which may be estimated by using the training data. We note that here we make a heuristic approximation: instead of computing the expected values (e.g., by simulation), we substitute the mean of the distances from the training data.

As we discussed previously, the Fisher score resembles how well the model can fit the data, thus we can recommend the best fitting next item j^* based on the norm of the Fisher score,

$$j^* = \arg\min_{j \neq i} ||G_{j|i}(\theta)||,$$

where we will use ℓ_2 for norm in our experiments.

5.2 Item-Item Fisher Distance (FD)

In our second model, we rank the next item by its distance from the last one, based on the Fisher metric. With the Fisher kernel $K(i, j)$, the *Fisher distance* can be formulated as

$$\mathrm{dist}_F(i, j) = \sqrt{K(i, i) - 2K(i, j) + K(j, j)}, \tag{11}$$

thus we need to compute the Fisher kernel over our generative model as in (6). The computational complexity of the Fisher information matrix estimated on the training set is $\mathcal{O}(T|\theta|^2)$, where T is the size of the training set. To reduce the complexity to $\mathcal{O}(T|\theta|)$, we can approximate the Fisher information matrix with the diagonal as suggested in [13,22]. Hence we will only use the diagonal of the Fisher information matrix,

$$\begin{aligned} F_{k,k} &= \mathbf{E}_\theta[\nabla_{\theta_k} \log p(i|\theta)^T \nabla_{\theta_k} \log p(i|\theta)] \\ &= \mathbf{E}_\theta[(\mathbf{E}_\theta[\frac{\partial U(i \mid \theta)}{\partial \theta_k}] - \frac{\partial (U(i \mid \theta)}{\partial \theta_k})^2]. \end{aligned}$$

For the energy functions of Eqs. (2) and (3), the diagonal of the Fisher kernel is the standard deviation of the distances from the samples. We give the Fisher vector of i for (2):

$$\begin{aligned} \mathcal{G}_i^k &= F^{-\frac{1}{2}} G_i^k \approx F_{kk}^{-\frac{1}{2}} G_i^k \\ &= \frac{\mathbf{E}_\theta[\mathrm{dist}(i, i_k)] - \mathrm{dist}(i, i_k)}{\mathbf{E}_\theta^{\frac{1}{2}}[(\mathbf{E}_\theta[\mathrm{dist}(i, i_k)] - \mathrm{dist}(i, i_k))^2]}. \end{aligned}$$

The final kernel function is

$$\begin{aligned} K(i, j) &= G_i^T F^{-1} G_j \approx G_i^T F_{diag}^{-1} G_j \\ &= G_i^T F_{diag}^{-\frac{1}{2}} F_{diag}^{-\frac{1}{2}} G_j = \sum_k \mathcal{G}_i^k \mathcal{G}_j^k. \end{aligned}$$

By substituting into (11), the recommended next item after item i will be

$$j^* = \arg\min_{j \neq i} \mathrm{dist}_F(i, j).$$

5.3 Multimodal Fisher Score and Distance

So far we considered only a single distance or divergence measure over the items. We may expand the model with additional distances with a simple modification to the graph of Fig. 1. We expand the points of the original graph into new points $R_i = \{r_{i,1}, .., r_{i,|R|}\}$ corresponding to R representatives for each item i_n in Fig. 3. There will be an edge between two item representations $r_{i,\ell}$ and $r_{j,k}$ if they are the same type of representation ($\ell = k$) and the two item was connected in the original graph. This transformation does not affect the maximal clique size and therefore the energy function is a simple addition, as

$$U(i \mid \theta) = \sum_{n=1}^{N} \sum_{r=1}^{|R|} \alpha_{nr} \text{dist}_r(i_r, i_{nr}), \tag{12}$$

and if we expand the joint similarity graph to a multimodal graph, the energy function will be

$$\begin{aligned} U(i, j \mid \theta) = \sum_{n=1}^{N} \sum_{r=1}^{|R|} \beta_{nr} (\text{dist}_r(i_r, i_{nr}) \\ + \text{dist}_r(j_r, i_{nr}) + \text{dist}_r(i_r, j_r)). \end{aligned} \tag{13}$$

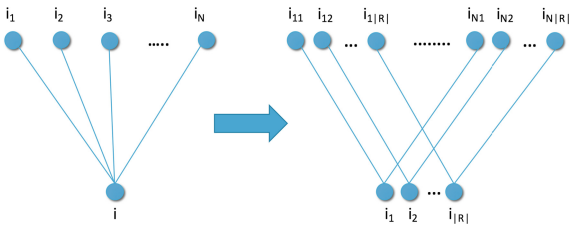

Fig. 3. The single and multimodal similarity graph with sample set $S = \{i_1, i_2, ..., i_N\}$ and $|R|$ modalities.

Now, let the Fisher score for any distance measure $r \in R$ be G_{ir}, than the Fisher score for the multimodal graph is concatenation of the unimodal Fisher scores as

$$G_i^{multi} = \{G_{i1}, .., G_{i|R|}\},$$

and therefore the norm of the multimodal Fisher score is a simple sum over the norms:

$$||G_i^{multi}|| = \sum_{r=1}^{|R|} ||G_{ir}||. \tag{14}$$

The calculation is similar for the Fisher kernel of Eq. (12), thus the multimodal kernel can be expressed as

$$K_{multi}(i,j) = \sum_{r=1}^{|R|} K_r(i,j). \tag{15}$$

6 Similarity Measures

Next we enumerate distance and divergence measures that can be used in the energy functions (2) and (3). Without using the Fisher information machinery, these measures yield the natural baseline methods for item-to-item recommendation. We list both implicit feedback collaborative filtering and content based measures.

6.1 Feedback Similarity

For user implicit feedback on item pairs, various joint and conditional distribution measures can be defined based on the frequency f_i and f_{ij} of items i and item pairs i, j, as follows.

1. Cosine similarity (Cos):
$$cos(i,j) = \frac{f_{ij}}{\sqrt{f_i f_j}}.$$
2. Jaccard similarity (JC):
$$JC(i,j) = \frac{f_{ij}}{f_i + f_j - f_{ij}}.$$
3. Empirical Conditional Probability (ECP): estimates the item transition probability:
$$ECP(j|i) = \frac{f_{ij}}{f_i + 1},$$
where the value 1 is a smoothing constant.

Additionally, in [16] the authors suggested a model, the Euclidean Item Recommender (EIR) to approximate the transition probabilities with the following conditional probability

$$p(j|i) = \frac{\exp^{-||x_i - x_j||^2 + b_j}}{\sum \exp^{-||x_i - x_k||^2 + b_k}},$$

where they learn the item latent vector x_i and bias b_i.

All of the above measures can be used in the energy function as the distance measure after small modifications.

Now, let us assume that our similarity graph (Fig. 1) has only one sample element i and the conditional item is also i. The Fisher kernel will be,

$$\begin{aligned} K(i,j) &= \frac{1}{\sigma_i^2}(\mu_i - \text{dist}(i,i))(\mu_i - \text{dist}(i,j)) \\ &= \frac{\mu_i^2}{\sigma_i^2} - \frac{\mu_i}{\sigma_i^2}\text{dist}(i,j)) \\ &= C_1 - C_2 * \text{dist}(i,j), \end{aligned}$$

where μ_i and σ_i are the expected value and variance of distance from item i. Therefore if we fix θ, C_1 and C_2 are positive constants and the minimum of the Fisher distance will be

$$\begin{aligned} \min_{j \neq i} \text{dist}_F(i,j) &= \min_{j \neq i} \sqrt{K(i,i) - 2K(i,j) + K(j,j)} \\ &= \min_{j \neq i} \sqrt{2C_2 * \text{dist}(i,j)} = \min_{j \neq i} \text{dist}(i,j). \end{aligned}$$

Hence if we measure the distance over the latent factors of EIR, the recommended items will be the same as defined by EIR, see Eq. (10) in [16].

6.2 Content Similarity

Besides item transitions, one can measure the similarity of the items based on their content (e.g., metadata, text, title). The content similarity between two items is usually measured by the cosine, Jaccard, tf-idf, or the Jensen-Shannon divergence of the "bag of words".

7 Experiments

We performed experiments on four publicly available data sets. As baseline methods, we computed four item-item similarity measures: Empirical Conditional Probability (ECP), Cosine (Cos), Jaccard (JC) as defined in Sect. 6, and we also implemented the Euclidean Item Recommender of [16]. As content similarity, we mapped the movies in the MovieLens dataset to DBpedia [1] [2]. DBpedia represents Wikipedia as a graph. For instance, a movie in DBPedia is represented as a node connected by labeled edges to other nodes such as directors, actors or genre. We compute the Jaccard similarity between two items using the nodes connected to the movies as a "bag of words". For evaluation, we use Recall, and Discounted Cumulative Gain (DCG) [14].

We conducted experiments by adding 200 sampled items to the testing item to evaluate recommendations. That is, given the current item in a session i and a known co-occurrence j we add randomly 200 items and rank them based on the score of the models. The best models should preserve j on the top of the sorted list.

[1] http://wiki.dbpedia.org.

Table 1. Co-occurrence quartiles

Dataset	25%	50%	75%	Max
Books	1	1	2	1,931
Yahoo! Music	4	9	23	160,514
MovieLens	29	107	300	2,941
Netflix	56	217	1,241	144,817

7.1 Data Sets and Experimental Settings

We carried out experiments over four data sets: Netflix [3], MovieLens[2], Ziegler's Books [30] and Yahoo! Music [8].

To generate item transitions by creating pairs from the items consumed by the users in the data set. For example, if a user consumed items a, b and c we create three co-occurrence pairs. That is, $[(a, b), (b, c), (c, a)]$. We do this for all the users and then we calculate the frequency of each pair. Figure 4 and Table 1 shows that most of the co-occurrence in the datasets are infrequent. 75% of the pairs have low item support. Since our research is focused on infrequent items, we filtered out the items with high support. The maximum co-occurrence frequency that we considered for the data sets in our experiments are 2 for Books, 23 for Yahoo! Music, 300 for MovieLens and 1241 for Netflix.

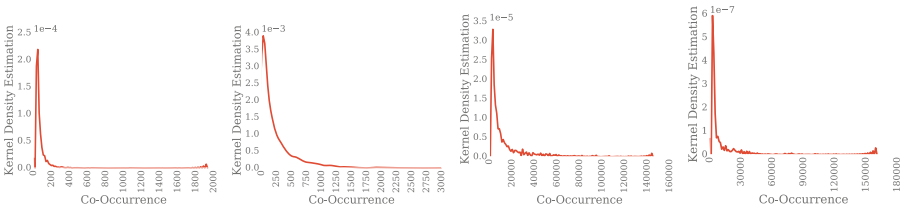

Fig. 4. The Kernel Density Estimation of the item co-occurrence saturates in items that are infrequent. From left to right and top to bottom: Books, MovieLens, Netflix, Yahoo! Music.

To generate the training and testing set we place most of the users in a training set and the rest of the users in the testing set. Then, we generate the pairs as described before. The number of training and testing pairs and the properties of the data sets can be seen in Table 2. During testing the evaluation was performed over a sampled set of 200 items as in [16] for all three metrics and solved ties arbitrary.

In our experiments, all algorithms use the item frequencies of the training period as input parameters. However, it could be possible to keep the cur-

[2] http://grouplens.org/datasets/movielens/.

Table 2. Data sets used in the experiments.

Data set	Items	Users	Training pairs	Testing pairs
Netflix	17,749	478,488	7,082,109	127,756
MovieLens	3,683	6,040	670,220	15,425
Yahoo! Music	433,903	497,881	27,629,731	351,344
Books	340,536	103,723	1,017,118	37,403

rent frequencies up to date and recalculate the prediction of each algorithm on the fly.

7.2 Experimental Results

This section presents different experiments related to the size of the sample set, the modalities used (e.g., implicit, content), the performance on infrequent items and finally the overall performance. As acronyms FC stands for Fisher conditional score from Sect. 5.1 followed by similarity, FD for Fisher distance from Sect. 5.2 followed by similarity. In case of multimodal the model use both content and collaborative similarity values.

Sample Set. The similarity graphs are defined via the set of items used as samples (Figs. 1, 2 and 3). To smooth the Fisher vector representation of sparse items we choose the most popular items in the training set as elements for the sample set. As we can see in Figs. 5 and 6, recommendation quality saturates at a certain

Table 3. Experiments with combination of collaborative filtering for the least frequent (25%) conditional items of the MovieLens data.

	Recall@20	DCG@20
Cosine	0.0988	0.0553
Jaccard	0.0988	0.0547
ECP	0.0940	0.0601
EIR	0.1291	0.0344
FC Cosine	0.1020	0.0505
FD Cosine	0.1578	0.0860
FC Jaccard	0.1770	**0.1031**
FD Jaccard	**0.1866**	0.1010
FC ECP	0.0940	0.0444
FD ECP	0.1626	0.0856
FC EIR	0.0861	0.0434
FD EIR	0.1068	0.0560

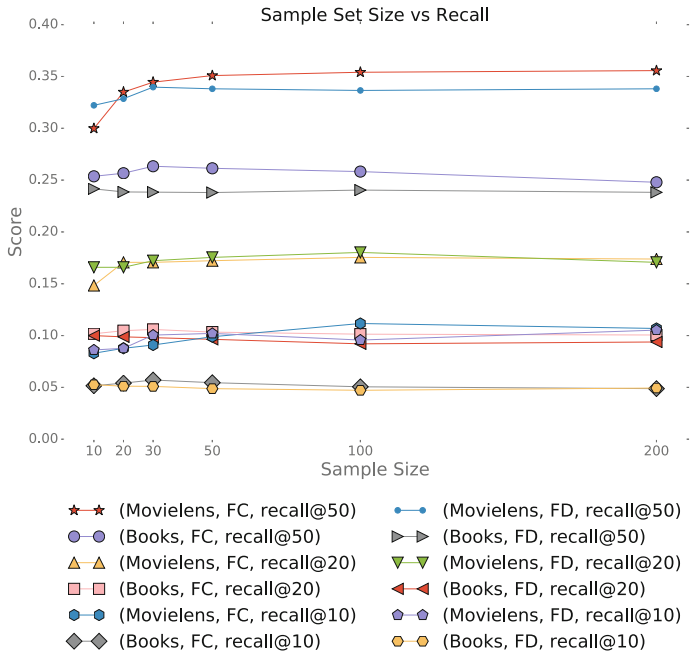

Fig. 5. The sample set size improves Recall until it saturates at 50. In this example we use Jaccard similarity.

Table 4. Experiments on MovieLens with DBPedia content, all methods using Jaccard similarity.

	Recall@20	DCG@20
Collaborative baseline	0.139	0.057
Content baseline	0.131	0.056
FC content	0.239	0.108
FD content	0.214	0.093
FC multimodal	0.275	0.123

sample set size. Therefore we set the size of the sample set to 20 for the remaining experiments.

Performance of Similarity Functions. Another relevant parameter of the similarity graphs is the choice of the similarity functions. Table 3 presents the performance of the different similarity functions. Overall, Jaccard similarity is the best performing and we used it for the rest of our experiments.

Performance on Infrequent Items. One of the main challenges in the field of recommendation systems is the "cold start" problem, therefore we examine the per-

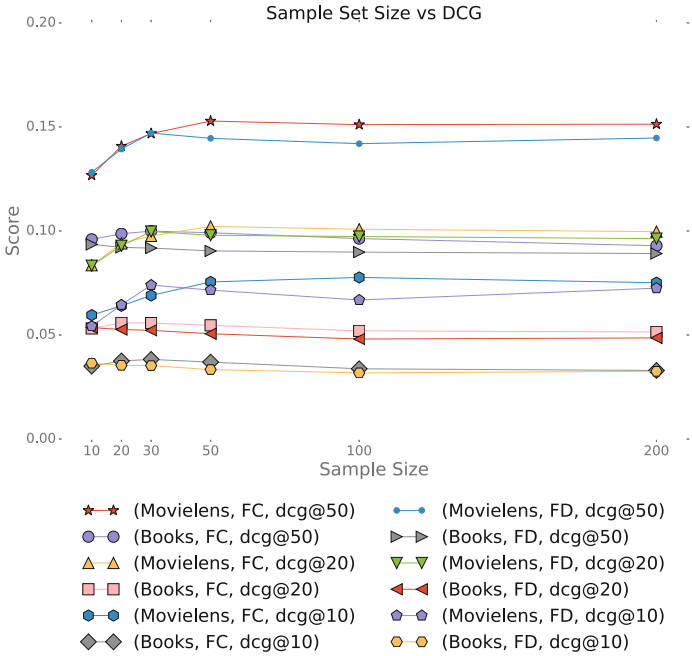

Fig. 6. The sample set size improves DCG until it saturates at 50. In this example we use Jaccard similarity.

formance in case of low item support. Figure 7 shows the advantage of the Fisher methods for infrequent items. As support increases, best results are reached by blending based on item support. If the current session ends with an item of high support, we may take a robust baseline recommender. And if the support is lower, less than around 100, Fisher models can be used to compile the recommendation.

Modalities: Implicit Feedback and Content. In Table 4 we show our experiments with DBPedia content as a modality on MovieLens. For simplicity, we set the size of the sample set for both Fisher models to 10. The overall best performing model is the multimodal Fisher with Jaccard similarity, while every unimodal Fisher method outperform the baselines. By using Eq. (15), we could blend different modalities such as content and feedback without the need of setting external parameters or applying learning for blending.

Summary of Performance vs Baselines. Tables 3, 4 and 5 present our implicit feedback results. The choice of the distance function strongly affects the performance of the Fisher models. As seen in Table 3, the overall best performing distance measure is Jaccard for both types of Fisher models. The results in Table 5 show that the linear combination of the standard normalized scores of the Fisher

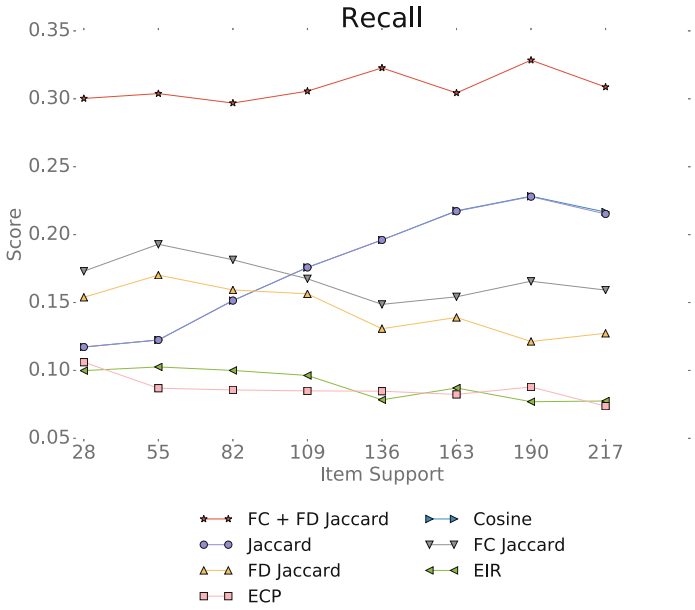

Fig. 7. Recall@20 as the function of item support for the Netflix data set.

methods outperforms the best unimodal methods (Fisher with Jaccard) for Netflix and Books, while for MovieLens and Yahoo! Music, Fisher distance with Jaccard performs best.

8 Discussion and Future Work

Recommending infrequent item-to-item transitions without personalized user history is a challenging problem. We consider our results for simple, non-personalized item-to-item recommendation as the first step towards demonstrating the power of the method. As a key feature, the model can fuse different modalities including collaborative filtering, content, and side information, without the need for learning weight parameters or using wrapper methods. In the near future, we plan to extend our methods to personalized recommendation settings and refine the underlying similarity measures with complex models (e.g., neural networks [31]). The publicly available datasets we used limited our experiments. Datasets containing a session id, item, and timestamp are scarce. Because of this, future work could be to experiment with real sessions, especially within a short period (e.g., news recommendation). Also, we constrained our similarity graphs for simple item-to-item transitions, defining the next item in the "random walk" depending only on the last seen item. To find out the limitation of this hypothesis we intend to expand the generative model to utilize the previous items in a session.

Table 5. Summary of experiments results for the four datasets. The *Max freq* are defined in the frequency quartile (Table 1). For most methods, there are (up to rounding errors) two best baseline and two best Fisher models, except for Recall where a third method Cosine appears in the cell marked by a star (∗). The best methods are usually FD Jaccard and FC + FD.

	Best baseline & new method	Max freq	MovieLens	Books	Yahoo! Music	Netflix
Recall@20	Jaccard	25%				0.13
		50%				0.18
		75%	0.12*			0.20
	EIR	25%	0.12	0.10	0.13	
		50%	0.11	0.10	0.11	
		75%		0.10	0.12	
	FD Jaccard	25%	**0.18**		**0.23**	
		50%	**0.19**		**0.23**	
		75%	**0.14**		**0.20**	
	FC + FD	25%		**0.14**		**0.30**
		50%		**0.14**		**0.30**
		75%		**0.13**		**0.31**
DCG@20	ECP	25%	0.05			
		50%	0.05			
		75%	0.05			
	EIR	25%		0.06	0.05	0.12
		50%		0.06	0.05	0.12
		75%		0.06	0.05	0.12
	FD Jaccard	25%	**0.10**		**0.11**	
		50%	**0.11**		**0.11**	
		75%	**0.08**		**0.10**	
	FC + FD	25%		**0.08**		**0.17**
		50%		**0.08**		**0.17**
		75%		**0.08**		**0.17**

9 Conclusions

In this paper, we considered the session based item-to-item recommendation task, in which the recommender system has no personalized knowledge of the user beyond the last items visited in the current user session. We proposed Fisher information based global item-item similarity models for this task. We reached significant improvement over existing methods in case of infrequent item-to-item transitions by experimenting with a variety of data sets as well as evaluation metrics.

Acknowledgments. The publication was supported by the Hungarian Government project 2018-1.2.1-NKP-00008: Exploring the Mathematical Foundations of Artificial Intelligence and by the Momentum Grant of the Hungarian Academy of Sciences. F.A. was supported by the Mexican Postgraduate Scholarship of the Mexican National Council for Science and Technology (CONACYT). B.D. was supported by 2018-1.2.1-NKP-00008: Exploring the Mathematical Foundations of Artificial Intelligence.

References

1. Adhikari, V.K., et al.: Unreeling netflix: understanding and improving multi-CDN movie delivery. In: INFOCOM, pp. 1620–1628. IEEE (2012)
2. Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., Ives, Z.: DBpedia: a nucleus for a web of open data. In: Aberer, K., et al. (eds.) ASWC/ISWC -2007. LNCS, vol. 4825, pp. 722–735. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-76298-0_52
3. Bennett, J., Lanning, S.: The netflix prize. In: Proceedings of KDD Cup and Workshop (2007)
4. Besag, J.: Statistical analysis of non-lattice data. Statistician **24**(3), 179–195 (1975)
5. Davidson, J., et al.: The YouTube video recommendation system. In: Proceedings of the Fourth ACM RecSys, pp. 293–296 (2010)
6. Deshpande, M., Karypis, G.: Item-based top-n recommendation algorithms. ACM Trans. Inf. Syst. (TOIS) **22**(1), 143–177 (2004)
7. Desrosiers, C., Karypis, G.: A comprehensive survey of neighborhood-based recommendation methods. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.) Recommender Systems Handbook, pp. 107–144. Springer, Boston, MA (2011). https://doi.org/10.1007/978-0-387-85820-3_4
8. Dror, G., Koenigstein, N., Koren, Y., Weimer, M.: The Yahoo! music dataset and KDD-Cup'11. In: KDD Cup, pp. 8–18 (2012)
9. Hammersley, J.M., Clifford, P.: Markov fields on finite graphs and lattices. Seminar (1971, unpublished)
10. Hidasi, B., Tikk, D.: Fast ALS-based tensor factorization for context-aware recommendation from implicit feedback. In: Flach, P.A., De Bie, T., Cristianini, N. (eds.) ECML PKDD 2012. LNCS (LNAI), vol. 7524, pp. 67–82. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-33486-3_5
11. Hidasi, B., Tikk, D.: Context-aware item-to-item recommendation within the factorization framework. In: Proceedings of the 3rd Workshop on Context-awareness in Retrieval and Recommendation, pp. 19–25. ACM (2013)
12. Hu, Y., Koren, Y., Volinsky, C.: Collaborative filtering for implicit feedback datasets. In: IEEE ICDM 2008, pp 263–272 (2008)
13. Jaakkola, T.S., Haussler, D.: Exploiting generative models in discriminative classifiers. In: Advances in Neural Information Processing Systems, pp. 487–493 (1999)
14. Järvelin, K., Kekäläinen, J.: Cumulated gain-based evaluation of IR techniques. ACM Trans. Inf. Syst. (TOIS) **20**(4), 422–446 (2002)
15. Jost, J.: Riemannian Geometry and Geometric Analysis. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-21298-7
16. Koenigstein, N., Koren, Y.: Towards scalable and accurate item-oriented recommendations. In: Proceedings of the 7th ACM RecSys, pp. 419–422. ACM (2013)
17. Koren, Y.: The bellkor solution to the netflix grand prize. Netflix Prize Documentation **81**, 1–10 (2009)

18. Koren, Y.: Factor in the neighbors: scalable and accurate collaborative filtering. ACM Trans. Knowl. Disc. Data (TKDD) **4**(1), 1 (2010)
19. Linden, G., Smith, B., York, J.: Amazon.com recommendations: item-to-item collaborative filtering. IEEE Internet Comput. **7**(1), 76–80 (2003)
20. Lops, P., de Gemmis, M., Semeraro, G.: Content-based recommender systems: state of the art and trends. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.) Recommender Systems Handbook, pp. 73–105. Springer, Boston, MA (2011). https://doi.org/10.1007/978-0-387-85820-3_3
21. Lösch, U., Bloehdorn, S., Rettinger, A.: Graph kernels for RDF data. In: Simperl, E., Cimiano, P., Polleres, A., Corcho, O., Presutti, V. (eds.) ESWC 2012. LNCS, vol. 7295, pp. 134–148. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-30284-8_16
22. Perronnin, F., Dance, C.: Fisher kernels on visual vocabularies for image categorization. In: IEEE CVPR 2007 (2007)
23. Pilászy, I., Serény, A., Dózsa, G., Hidasi, B., Sári, A., Gub, J.: Neighbor methods vs. matrix factorization - case studies of real-life recommendations. In: ACM RecSys 2015 LSRS (2015)
24. Rendle, S., Freudenthaler, C., Schmidt-Thieme, L.: Factorizing personalized Markov chains for next-basket recommendation. In: Proceedings of the 19th International Conference on WWW, pp. 811–820. ACM (2010)
25. Janke, W., Johnston, D., Kenna, R.: Information geometry and phase transitions. Physica A: Stat. Mech. Appl. **336**(1), 181–186 (2004)
26. Ricci, F., Rokach, L., Shapira, B.: Introduction to Recommender Systems Handbook. Springer, Boston (2011). https://doi.org/10.1007/978-0-387-85820-3_1
27. Sarwar, B., Karypis, G., Konstan, J., Reidl, J.: Item-based collaborative filtering recommendation algorithms. In: Proceedings of the 10th International Conference on WWW, pp. 285–295 (2001)
28. Schein, A.I., Popescul, A. Ungar, L.H. Pennock, D.M.: Methods and metrics for cold-start recommendations. In: Proceedings of the 25th ACM SIGIR, pp. 253–260. ACM (2002)
29. Čencov, N.N.: Statistical Decision Rules and Optimal Inference, vol. 53. American Mathematical Society (1982)
30. Ziegler, C.-N., McNee, S.M., Konstan, J.A., Lausen, G.: Improving recommendation lists through topic diversification. In: Proceedings of the 14th International Conference on WWW, pp. 22–32. ACM (2005)
31. Wang, H., Yeung, D.-Y.: Towards bayesian deep learning: a framework and some existing methods. IEEE Trans. Knowl. Data Eng. **28**(12), 3395–3408 (2016)

Author Queries

Chapter 20

Query Refs.	Details Required	Author's response
AQ1	This is to inform you that corresponding author has been identified as per the information available in the Copyright form.	
AQ2	References [1, 12, 21] are given in list but not cited in text. Please cite in text or delete from list.	