- **Article Title**
  Recommender systems with heterogeneous information network for cold start items

- **Authors**
  Zhang, Di; Zhang, Qian; Zhang, Guanquan; Lu, Jie

- **Name of conference series**
  14th International FLINS Conference (FLINS 2020)

- **Published proceedings**
  Developments of Artificial Intelligence Technologies in Computation and Robotics

- **Publication date:** Oct 2020
-

DIO: 10.1142/9789811223334_0060

# Recommender systems with heterogeneous information network for cold start items

Di Zhang, Qian Zhang, Guanquan Zhang, Jie Lu

*Decision Systems and e-Service Intelligence Laboratory,*
*Centre for Artificial Intelligence*
*University of Technology Sydney, Australia*
*Di.Zhang-7@student.uts.edu.au, Qian.Zhang-1@uts.edu.au*
*Guangquan.Zhang@uts.edu.au, Jie.Lu@uts.edu.au*

Recommender System has been widely adopted in real-world applications. Collaborative Filtering (CF) and matrix based approach has been the forefront for the past decade in both implicit and explicit recommendation tasks. One prominent challenge most recommendation approach facing is dealing with different data quality conditions. i.e. cold start and data sparsity. Some model based CF methods use condensed latent space to overcome sparsity problem. However, when dealing with constant cold start problem, CF approach can be ineffective and costly. In this paper, we propose MERec, a novel approach that adopts graph meta-path embedding to learn item/user features independently besides learning from user-item interactions. Such approach allows unseen data to be incorporated as part of user/item learning process, hence effectively reduce the impact in cold start problem for new or sparse dataset.

*Keywords*: recommender system; cold-start; heterogeneous information network.

## 1. Introduction

Recommender system is an indispensable technology in this big data era [1]. It helps us to find personalized products or services from the ever-increasing information and make our life more efficient and focused. Collaborative Filtering (CF) and matrix factorization methods are the dominant recommendation approach for the past decades. These methods predict user's preference base off user-item interactions. Consequently, a warming up period is commonly required for CF based approach to over come the data sparsity problems, by accumulating interactions records.

In general, matrix factorization approach projects user/item features into a latent space, then calculate user-item similarity based on their condensed latent vector values. Such methods are not suited in dealing brand new items, which have a wide range of use case in real-world scenarios. A known workaround is running a routine end to end re-train process to pick up new interactions, which can be inefficient and costly. Content-based or hybrid recommendation approaches are one way to overcome the problem. It is proved to be challenging when dealing large categorical features, such as exponential computation complexity increase, as well as introducing noise.

Recently, many embedding methods are developed via Heterogeneous Information Network (HIN) based approaches[2,3]. Graph data structure enables continuously evolving information. The embedding process learns item/users' representations based on nodes' local and global structure, via propagation process such as random walk, such as Node2vec[4] and DeepWalk[5]. However, random walks based presentation learning are known to be more biased to highly connected nodes[6]. This problem is alleviated by meta-path based approaches[7] especially for recommendation with explicit feedback. However, not many research emphasize on the data sparsity problem and they cannot deal with brand new items. Moreover, many real-world applications are focusing on implicit feedback, such as optimizing click through rate. This remains to be a challenging problem to be solved.

In this paper we propose a Meta-path Embedding based Recommendation (MERec), a novel approach that utilizing meta-path based presentation learning on HIN. This approach is primarily focusing on solving the cold start problems when data is continuously evolving. It is also designed to be adaptive to newly emerged data, so that recommendation could be made even there is zero user-items interaction available. By translating domain experts relevancy rules into meta-path set, our approach allow a controlled random walk to reduce computation intensity and improve on end result.

We Identify the paper contributions as following:

- A novel meta-path based embedding method to effectively handle cold-start problem in recommendation with implicit feedback;
- Our approach uses meta-path based item feature embedding, then learns user presentation in subsequent step using interactions data, allowing model to predict recommendations on brand new items.
- Experiments are conducted on 3 different data conditions: pure cold start, sparse data, non-sparse data. the end result shows consistent performs across all 3 scenarios;

The rest of this paper is constructed as follows. In Section 2, we discuss the related research and relevant definition that helped our research. In Section 3, we explain our framework and related algorithms. In Section 4, we show our experiment design and result compared with other baselines in cold-start scenario and sufficient data scenario. Lastly, we give conclusion and future directions in Section 5.

## 2. Preliminaries

Following definitions are preliminaries of our approach.

**Definition 2.1 (Heterogeneous Information Graph).** *An information graph is $G = (V, E)$, where $V$ is the set of nodes (or entities) of the graph. $E$ is the set of edges connecting the nodes in $V$, $E \subset V \times V$. Entity and link type mapping $\phi$: $V \to A$, $\varphi$: $E \to R$, where $A$ and $R$ denote the sets of predefined entity and link types, and $|A| > 1$ or $|R| > 1$ indicating that there are more than one type of nodes and inter nodes relationships.*

**Definition 2.2 (Meta-Path).** *A Meta-Path $\mathcal{P}$ is a path defined on the graph schema $T_G = (A, R)$.*
*Meta-Path $\mathcal{P}$ is denoted as $A_1 \xrightarrow{r1} A_2 \xrightarrow{r2} \ldots \xrightarrow{rn} A_n$.*

*Relationship $R$ is denoted as $r1 \bullet r2 \bullet ...rn$ for different types of relationship between different types of entity nodes, where $\bullet$ denotes composition operator or relations.*

Meta-Path $\mathcal{P}_1 : A_1 \xrightarrow{r_1} A_2$ is the Meta-Path connects source entity type $A_1$ and target $A_2$. Similarly, $\mathcal{P}_2 : A_2 \xrightarrow{r_2} A_3$, $\mathcal{P}_2$ is the Meta-Path between source entity type $A_2$ and target entity type $A_3$. Here we call $\mathcal{P}_1$ and $\mathcal{P}_2$ are contactable. Then $\mathcal{P}_1$ and $\mathcal{P}_2$ can be combined as $\mathcal{P}_{1,2} : A_1 \xrightarrow{r_1} A_2 \xrightarrow{r_2} A_3$. For example, $Movie \to Director$ and $Director \to Movie$ can be combined to $Movie \to Director \leftarrow Movie$.

## 3. Meta-Path Embedding based Recommendation

In this Section, we propose MERec that meta-path based embedding and Bayesian Probability Ranking to overcome cold start problem.

The key idea of our approaches is inspired by recent research on graph-based embedding[7]. Unlike traditional matrix based approach, learning user/item latent representation jointly based on user-item interactions matrix. Our approach breaks down item and user latent matrix learning through 2 separate steps. as illustrated in Fig 1. Item embedding is first learned based on item and its features nodes within HIN with a user defined

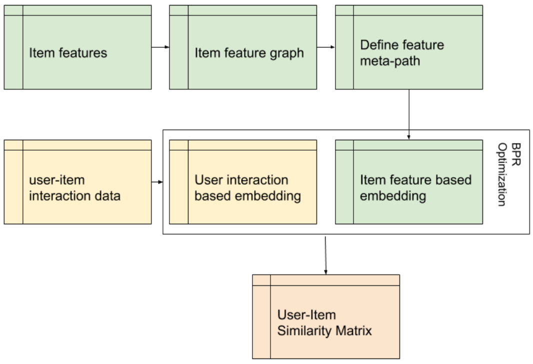


Fig. 1. workflow of MERec

meta-path sets for the moderating its random walk process. Then user latent feature is trained though Factorization Machine, which is optimized by using Bayesian Probability Ranking.

### 3.1. *Step 1: Item Embedding via Meta-path Based Random Walk*

For items such as movies, books, music, there are a number of factors impacting users' decision. Instead of taking the one-hot encoding approach for category value. We first propose to treat different categorical information as separate node types. By leveraging domain knowledge we can form a heterogeneous information graph and sets of user defined meta-path for item-item similarities learning.

For example, movie choice is closely linked to directors and its casts. Thus $Movie \rightarrow Director \leftarrow Movie$, $Movie \rightarrow Actor/Actress \leftarrow Movie$ can be very important meta-path in deciding how similar 2 movies are. We use those insights as a guideline to define meta-path from a heterogeneous information graph for item features embedding. As shown in Fig 2. Each meta-path can derive a item-item similarity matrix $\mathbb{R}_i(\mathcal{P}_i)$, each different meta-path can be regarded as bias toward different feature aspects, so items co-occurrence can be learned separately under different meta-path.

as a result, item-item similarity score can be calculated by normalized meta-path similarity times meta-path weight.

$$\mathcal{S}(v_i, v_j) = \begin{cases} \sum_{n=1}^{n} \mathcal{R}_i j(\mathcal{P}_n, W_n), & \text{if } (v_i, .., v_j) \in \mathcal{P} \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

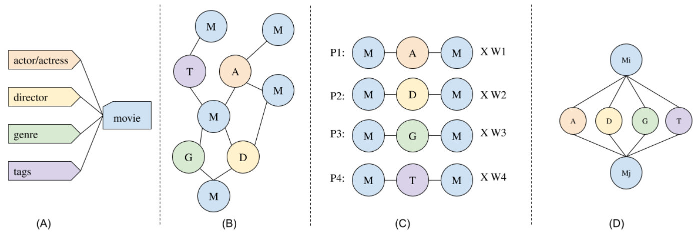


Fig. 2. heterogeneous information graph based on item features

$\mathcal{S}(v_i, v_j)$ stands for similarity between items $v_i$ and $v_j$ which shares the same node type. $R_ij()$ is a similarity function, where $\mathcal{P}_n, W_n$ stands for individual meta-path and its weights respectively. This end result provides guidance for the random walkers on our heterogeneous graph of item features. $(v_i, .., v_j)$ is denoted as a meta-path instance, where $v_i$ is the starting node and $v_j$ the end node.

Given a heterogeneous graph $G = (V, E)$, and a meta-path set $[\mathcal{P}_1, \mathcal{P}_2, ...\mathcal{P}_n]$, the probability of transition is defined as following:

$$P(v_{i+1}, \mathcal{P}, w) = \begin{cases} p(N^{t+1}(v_i^t)), & \text{if } (v_{i+1}, .., v_i^t) \in \mathcal{P} \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

where $t$ is denoted as $t^th$ steps, as the walker traversing through the graph and $p(N^{t+1}(v_i^t))$ is a $softmax$ function on top of the neighbors of node $v_i^t$ as $p(N^{t+1}(v_i^t)) = \frac{Exp(\mathcal{S}(v_i,v_j))}{\sum\limits_{n=1}^{n} Exp(\mathcal{S}(v_i,v_j))}$.

We enable skip-gram to learn the presentation of given node $v$:

$$argmax \sum_{v \in V} \sum_{c \in N(v)} \log p(c|v;\theta) \tag{3}$$

where$\log p(c|v;\theta))$ is defined as above. $c$ is denoted as $context$, in graph structure setting, $c$ is the neighboring nodes of given node $v$, i.e. $N(v)$.

During the item embedding learning, we introduce 3 hyper parameters to learn the item vector representation. $d$ for dimension size, $x$ for number of walks, and $l$ for depth of each random walk.

By the end of this step, what we have is the item representation being projected in to a latent embedding space with reduced the complexity and sparsity issue comparing to content based approach. In Section 4, we would show more detailed comparison results with traditional categorical one-hot encoding approach

### 3.2. *Step 2: User Latent Feature Learning*

After having feature embedding learned. Instead of learning $U, V$ jointly, we replace $V$ with the meta-path based item embedding vectors $Vec(v)$. This approach provides several benefits:

(1) Reducing learning complexities.
(2) Taking both item features and user-item interactions into account.
(3) $Vec(v)$ is not impacted even though user-item interactions are sparse.

We use Bayesian Probability Ranking as our optimization objective:

$$argmax \sum_{v \in V} (u_i \cdot Vec(v) - u_j \cdot Vec(v)) - \frac{\lambda}{2} tr(U^T U) \tag{4}$$

Where $u_i \cdot Vec(v)$ is a positive interaction between item $v$ and user $i$, while $u_j$ is a randomly selected negative sample. A fixed item embedding, also means during the training process user matrix is trained to fit the user-item interaction based on the item embedding. This means users representation is optimized independently based on item-feature embedding. As a result, the end user-item similarly score are equally effective for both observed and user-item pairs and brand new items. This characteristic ensures the final result to be effective on data sparsity problem as well as pure cold start problem.

Finally, we rank user candidate based on a dot product between trained user vector and item embedding vector. $score = u \cdot Vec(v)$

## 4. Experiments and Results

We have run a number of experiments based on several common real world scenarios, and we compared our approach with other popular methods through the experiments.

### 4.1. *Dataset and experiment setup*

Our experiments is based on HetRec 2011[8] data set. To construct the HIN, We use movies which have at least 10 distinctive user views. We also limit top 3 actors/actresses to be associated. We use tags information which is shared by more than 1 movies. In the end, we concluded 6 different types of *node* which can be used in HI: $Users$, $Movies$, $Tags$, $Actors$, $Directors$, $Genres$. Additionally, 5 relationships are defined as edges, which are

$Movie - Users$, $Movie - Tags$, $Movie - Actors$, $Movie - Directors$, and $Movie - Genres$.

For the simplicity of the experiment, we weigh all of our meta-path equally, as it is sufficient in illustrating the effectiveness of MERec. On the other hand, we do acknowledge that, those hyper-parameters can be further tuned to improve the result to tailor different problem domains.

For pure cold-start problem, we split our data set based on time. We use movies prior 2008 as training data set (6724 movies), and 2008 2011 as testing data set (193 movies). The choice of meta-path in this experiment are: $Movie-Tags-Movie$, $Movie-Actors-Movie$, $Movie-Directors-Movie$, and $Movie-Genres-Movie$. In this problem setting, we compare our result with common One-Hot encoding approach for categorical data in content based recommendation approach.

For data sparsity problem, we split data in two sparse ratios to evaluate the effectiveness of MERec performance. we also compares it with CF+BPR, which is one of the most popular and effective algorithms. For sparse data scenarios, the data density set to be 1.1%, while in non-sparse experiment settings, the density is set at 2.3%.

### 4.2. *Experiment results*

In pure cold start case, as shown in Table 1, the MERec is far superior comparing to one hot encoding when handling complex categorical.

Table 1. Pure cold start comparison results

| Models | Precision@5 | Recall@5 | Precision@10 | Recall@10 |
|---|---|---|---|---|
| One-Hot | 0.1229 | 0.0146 | 0.1481 | 0.0334 |
| MERec | **0.3037** | **0.0402** | **0.2711** | **0.0665** |

As shown in the result, when dealing with complex categorical data. EMRec had shown a significant edge comparing with one-hot encoding approach, with near doubling both precision and recall in pure cold start problem settings.

We run experiment on both sparse data split as well as non-sparse data split. we see the comparison as below in table 2: In sparse data split, we can see in average a near 20% performance boost comparing to Traditional Matrix Factorization based approach. While in non-sparse case, we can still see some minor enhancement across precision and recall.

Table 2. Sparse data comparison results

| Data sparsity | Models | Precision@5 | Recall@5 | Precision@10 | Recall@10 |
|---|---|---|---|---|---|
| 1.1% | MF | 0.5498 | 0.0126 | 0.5548 | 0.0265 |
| | MERec | **0.6194** | **0.0163** | **0.6035** | **0.0308** |
| 2.3% | MF | 0.5566 | 0.0175 | 0.5419 | 0.0336 |
| | MERec | **0.5858** | **0.0195** | **0.5575** | **0.0355** |

Through above 3 experiments, EMRec exhibits a stable superior performance in various data quality scenarios.

## 5. Conclusion and Future Work

In this paper we propose MERec, a novel approach that can effectively predict newly emerged item/user when there is little or no interaction data available. It has a wide application in real-world recommendation tasks, such as job, real-estate, where cold-start is a common problem. While MERec allows user to define meta-path settings in terms of controlling random walk process, it also opens up possible enhancement on learning different weights automatically across multiple meta-path with in the HIN. Making the embedding inductive instead of transitive is another interesting topic worthy exploring.

## References

1. J. Lu, D. Wu, M. Mao, W. Wang and G. Zhang, Recommender system application developments: a survey, *Decision Support Systems* **74**, 12 (2015).
2. M. Mao, J. Lu, G. Zhang and J. Zhang, Multirelational social recommendations via multigraph ranking, *IEEE Transactions on Cybernetics* **47**, 4049 (2016).
3. W. Wang, G. Zhang and J. Lu, Member contribution-based group recommender system, *Decision Support Systems* **87**, 80 (2016).
4. A. Grover and J. Leskovec, node2vec: Scalable feature learning for networks, in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016.
5. B. Perozzi, R. Al-Rfou and S. Skiena, Deepwalk: Online learning of social representations, in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2014.
6. Y. Sun, J. Han, X. Yan, P. S. Yu and T. Wu, Pathsim: meta path-

based top-k similarity search in heterogeneous information networks, *Proceedings of the VLDB Endowment* **4**, 992 (2011).

7. Y. Dong, N. V. Chawla and A. Swami, Metapath2vec: Scalable representation learning for heterogeneous networks, in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2017.
8. I. Cantador, P. Brusilovsky and T. Kuflik, 2nd workshop on information heterogeneity and fusion in recommender systems, in *Proceedings of the 5th ACM conference on Recommender systems*, RecSys 2011 (ACM, New York, NY, USA, 2011).