Collaborative Metric Learning with Memory Network for Multi-Relational Recommender Systems

Xiao Zhou[1] , **Danyang Liu**[2] , **Jianxun Lian**[3] and **Xing Xie**[3]

[1]Department of Computer Science and Technology, University of Cambridge, UK

[2]University of Science and Technology of China, Hefei, China

[3]Microsoft Research, Beijing, China

xz331@cam.ac.uk, ldy591@mail.ustc.edu.cn, {Jianxun.Lian, Xing.Xie}@microsoft.com

Abstract

The success of recommender systems in modern online platforms is inseparable from the accurate capture of users' personal tastes. In everyday life, large amounts of user feedback data are created along with user-item online interactions in a variety of ways, such as browsing, purchasing, and sharing. These multiple types of user feedback provide us with tremendous opportunities to detect individuals' fine-grained preferences. Different from most existing recommender systems that rely on a single type of feedback, we advocate incorporating multiple types of user-item interactions for better recommendations. Based on the observation that the underlying spectrum of user preferences is reflected in various types of interactions with items and can be uncovered by latent relational learning in metric space, we propose a unified neural learning framework, named Multi-Relational Memory Network (MRMN). It can not only model fine-grained user-item relations but also enable us to discriminate between feedback types in terms of the strength and diversity of user preferences. Extensive experiments show that the proposed MRMN model outperforms competitive state-of-the-art algorithms in a wide range of scenarios, including e-commerce, local services, and job recommendations.

1 Introduction

With the ability to select the most relevant content from the flood of information resources, recommender systems have played an increasingly vital role in modern society by optimising user experiences for individuals and promoting business objectives for online platforms [Tang *et al.*, 2016]. Keeping pace with the growing requirements of customisation and personalisation, recommender systems that are capable of learning fine-grained individual preferences with a concise, flexible, and efficient structure are eagerly expected.

Given that mining valuable information hidden in users' historical interactions with items in multiple ways might hold the key to better characterise users, we devise a recommendation scheme that takes full advantage of various user feedback types in this paper. Before diving into the details, it is
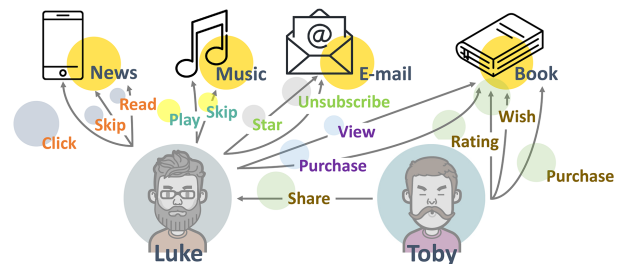

Figure 1: An example of multiple types of user feedback.

necessary to clarify the meaning of '*multiple feedback types*', which can be varied in different scenarios. As exemplified in Figure 1, possible interaction types existing between users and recommended items include clicking, skipping, and reading for news recommendation; playing and skipping songs in music recommendation scenario; and purchasing, rating, and sharing for book recommendations.

Most existing item-to-user recommender systems typically only employ one primary type of user-item interaction [Hu *et al.*, 2008], like clicking for online news recommendation and purchasing for the e-commerce scenario. Apart from primary feedback, online systems also allow users to leave additional types of feedback as listed above. Unfortunately, the power of using such extra feedback information to enhance recommendation performance has been largely neglected. In this paper, we leverage binary preference data in the implicit form and build a unified recommendation model that makes full use of multiple types of user-item interactions in various scenarios. Generally, it offers the following key advantages: 1) *Alleviating data sparsity problem.* Compared to explicit data like numerical ratings, implicit feedback data are relatively cheap and widely available since they are usually gathered by systems automatically. Through utilising rich implicit feedback data, the sparsity issue can be effectively alleviated; 2) *More precise and comprehensive user preference profile.* Incorporating multiple feedback types is essential for fine-grained detection of individual **preference diversity** and **strength**. Take the e-commerce scenario as an example, Toby likes Tesla's vehicles and always pays close attention to its latest models. However, he has never bought a Tesla in real life as the price is beyond his reach. In this case, if only purchasing feedback is taken into account, Toby's interest in Tesla can hardly be captured. In another instance, when Luke considered buying

a mobile phone, he browsed both iPhone 8 and iPhone XS before eventually buying the latter. By considering both browsing and purchasing feedback types, we can infer that Luke had a stronger preference for iPhone XS, relative to iPhone 8.

In this paper, we argue that jointly modeling multiple feedback types can help us reveal the underlying spectrum of user preferences in different dimensions and thus lead to better recommendation performance. More specifically, our ideas are materialised in the form of a neural learning framework that leverages the recent advancements of attention mechanism, augmented memory networks as well as metric learning. Our main contributions can be summarised as follows:

- We propose an end-to-end neural network architecture, named MRMN to address implicit collaborative filtering with multiple types of user feedback data.

- External memory and attention mechanism are augmented in MRMN, making it capable of learning adaptive relational vectors in metric place for each specific user, item, and feedback type, and detecting fine-grained user-item relations and multidimensional preferences.

- MRMN uncovers the underlying relationships between feedback types and shows multi-task ability to predict various types of user actions using one unified model.

- Comprehensive experiments on real-world datasets demonstrate the effectiveness of MRMN against competitive baselines in various recommendation scenarios.

- Qualitative analyses of the attention weights provide insights into the learning process of relations and illustrate the interpretability of MRMN in capturing higher order complex interactions between users and items.

2 Related Work

The past two decades have witnessed tremendous advances in the recommendation techniques from content-based filtering [Pazzani and Billsus, 2007] to collaborative filtering [Ekstrand et al., 2011], from explicit feedback [Koren, 2008] to implicit feedback [He and McAuley, 2016], and from shallow models to deep models [Lian et al., 2018]. While most recommender systems typically take one type of user-item interaction into account, we emphasise the critical importance of incorporating multiple types of user feedback into the recommendation. A review of existing literature on recommender systems with multiple feedback types is provided below.

Based on LinkedIn recommendation products, [Tang et al., 2016] presented a general investigation of possible ways to incorporate multiple types of user feedback from an empirical standpoint. To make a more specific discussion, other studies considering multiple feedback types in recommender systems can be mainly divided into three categories: sampling-based, model-based, and loss-based approaches. For the first group of methods focusing on sampling refinements, [Loni et al., 2016] extended the sampling method of vanilla Bayesian Personalized Ranking (BPR) [Rendle et al., 2009] to distinguish different strengths of user preferences reflected by various feedback types. Similarly, [Ding et al., 2018a] leveraged view data in e-commerce recommendation and developed a view-enhanced sampler for classical BPR. Another line of

works that sought to utilise multi-feedback interactions focused on model modification. For instance, [Liu et al., 2017] developed the MFPR model that employed one type of explicit feedback (e.g., ratings) and several types of implicit user feedback (e.g., viewing, clicking logs) as input. Based on LSTM networks, [Li et al., 2018] designed an architecture that enables us to learn the short-term motivation and long-term preferences of users for the next purchase recommendation. By devising tailored loss function, [Ding et al., 2018b] emphasised the importance of integrating view feedback in e-commerce recommendation and added pairwise ranking relations between purchased, viewed, and non-interacted actions instead of applying pointwise matrix factorization methods.

Through the literature review above, it can be found that the existing approaches proposed to handle multi-feedback recommendation tasks are relatively simple. There is a lack of in-depth investigation of relationships between user feedback types. In most cases, the importance weights of multiple feedback types were set manually. Moreover, current studies generally built the model in one particular recommendation scenario that a generic and adaptive architecture for multi-relational recommendation is still missing.

3 Background

To fill the research gap discovered in the current literature, we build a novel neural network framework that can learn multi-relational vectors for each specific user, item, and feedback type adaptively and flexibly. Our work is highly inspired by recent advances in metric learning and memory networks, which are introduced in the following two subsections.

3.1 Metric Learning based Recommendation

Over the last decade, matrix factorization (MF), as one of the most outstanding representative techniques of collaborative filtering (CF), has gained rapid acceptance in the field of recommender systems. Essentially, the main purpose of MF-based methods is to extract features of users and items by mapping them to a latent vector space. By doing this, MF can capture the existing user-item interactions approximately and infer the missing values by inner product for further recommendation [Tay et al., 2018]. Even though it is able to detect the most prominent features, we can hardly expect MF to uncover more fine-grained user preferences and to generate interpretable recommendations. From a fire-new perspective, [Hsieh et al., 2017] applied metric learning techniques to CF, and proposed collaborative metric learning (CML) algorithm. The essential distinction between CML and MF is that CML maps users and items in a metric space to minimise the distance between them if positive interactions exist. However, for many-to-many collaborative ranking task, fitting all positive user-item pairs into the same point in vector space would inevitably cause geometrical congestion and instability. To overcome this limitation, [Tay et al., 2018] proposed Latent Relational Metric Learning (LRML) to learn a latent relation vector for each given user-item pair. The architecture of our proposed model takes advantage of this metric-based learning scheme and enables multi-relational modeling in metric space for the collaborative ranking task, which obeys the crucial triangle inequality to capture fine-grained user preferences.

3.2 Memory Networks for Recommendation

The memory network was initially proposed in the context of question answering [Weston *et al.*, 2015]. Until recently, it has been applied to recommender systems. [Huang *et al.*, 2017] extended end-to-end memory networks to recommend hashtags for microblogs. [Chen *et al.*, 2018] employed memory networks for e-commerce recommendations using historical purchase records of users. [Ebesu *et al.*, 2018] proposed a collaborative memory network, where memory slots were used to store user preferences and encode items' attributes. [Zhou *et al.*, 2019] proposed a topic-enhanced memory network to optimise point-of-interest recommender systems.

The success of these recent applications highlight that the memory network architecture is efficient and flexible enough to perform joint task learning [Ebesu *et al.*, 2018], making it a suitable approach to our multi-relational learning task naturally. To the best of our knowledge, no prior work has employed the memory network for collaborative metric learning using multiple types of user feedback in recommender systems. Then, how to apply the memory mechanism properly in our case becomes a key focus. In question answering, the most classical application of memory networks, a short passage is provided along with a relevant question, the answer to which can be generated automatically via memory blocks. If we analogise our multi-relational recommendation task to a question answering problem, the question we are asking now becomes how likely a user would enjoy an item she has never interacted with, based on the user-item relations learned from the memory network module.

4 Proposed Model

We introduce a unified hybrid model called Multi-Relational Memory Network (MRMN) for personalised recommendation with multiple user-item interactions in this section. The overall architecture of MRMN model is presented in Figure 2. Generally speaking, we fuse a memory component and neural attention mechanism to learn multi-relational vectors for each specific user-item pair in a non-linear fashion and produce the ranking score of candidate items to the target user for recommendation. In the following subsections, we will first give the formulation of our prediction task before introducing the structure of the proposed MRMN model in detail.

4.1 Problem Formulation

Our research targets at recommender systems that operate multiple types of user feedback data in implicit form. Assume that we are given a set of users U and a set of items I, associating with each other through t types of implicit feedback ($t \geqslant 2$). For each feedback type τ in set T, the binary relations between U and I are represented by an interaction matrix Y^τ, where an entry is set to 1 if the interaction exists between the pair of user and item. Otherwise, it is set to 0, meaning that the user has not interacted with the item yet. The main task of the model is to derive a prediction score for a given user-item pair, signalling the likelihood that the user would be interested in the item based on her historical interactions with items in multiple ways.
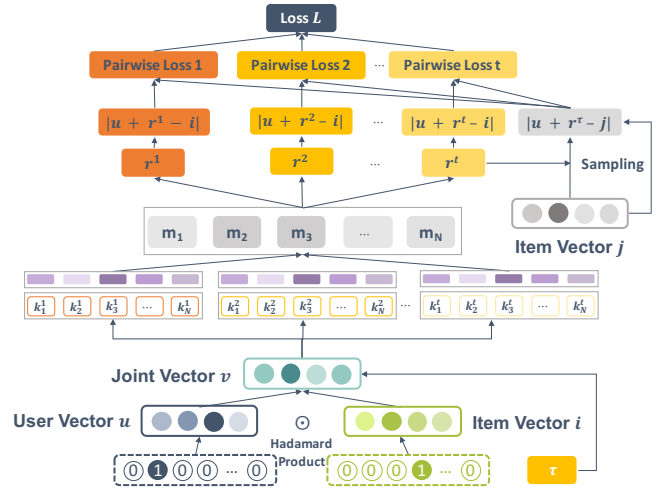

Figure 2: The architecture of MRMN.

4.2 User and Item Embedding

For a training set consisting of triplets (user, item, τ), the identities of a user and an item, represented as two binary sparse vectors via one-hot encoding, are employed as input features initially. Then the two vectors are projected to low-dimensional dense vectors to generate a pair of user and item embeddings, denoted as (u, i). After that, Hadamard product is applied to learn a joint user-item embedding v through:

$$v = u \odot i \tag{1}$$

The dimension of the generated vector $v \in \mathbb{R}^d$ is the same as u and i, which is d. Apart from the pair of user and item, the input feedback type τ plays as a controller. Given that the MRMN is devised as a partially shared structure for various user feedback types, the observed interaction type τ for a certain user-item pair determines which part of the model would be activated during the training scheme. This particular design makes the learned relation vector specific to each triplet of a user, item, and interaction type, even though the embeddings of u and i are shared across various feedback types.

4.3 Key Addressing and Attention Layers

To discriminate the diverse interests of a user reflected via various interaction types, neural attention mechanism is used to read a memory matrix and generate multi-relational vectors for different feedback types adaptively. The main idea of the attention mechanism is to rescue the model from encoding all information into each memory slot. We assume that the multiple relations learnt from different feedback types for the same user-item pair can not only be distinguished from each other but also have inherent connections, characterised by the attention module over augmented memory blocks.

Formally, a key matrix $K^\tau \in \mathbb{R}^{d \times N}$ is built for each feedback type τ, from which the attention vector of a particular feedback type can be obtained. Let N represent the number of key slots in K^τ, which is a user-specified hyperparameter. For a given joint embedding vector v and an observed interaction type τ, the similarity between v and each key slot k_i^τ

in the matrix K^τ is calculated by dot product to obtain each element of the attention vector ω^τ for feedback τ by:

$$\omega_i^\tau = v^T k_i^\tau \tag{2}$$

Then the generated attention weights in vector ω^τ are converted into a probability distribution over memory blocks using the softmax function to infer the importance of each memory slot's contribution for a given feedback type τ.

4.4 Multi-relational Vectors Generation

Next the generated attention vector ω^τ is utilised to learn a weighted sum of a sequence of memory slices in the memory matrix $M \in \mathbb{R}^{N \times d}$. Here the size of each memory slice is d, and the number of slices is N. The memory component is the core of the system where complex user-item relationships reflected in multiple feedback types are built. It is worth noting that different from the key addressing layer, where feedback types learn their key vectors, respectively; while here, they share the memory matrix. Each memory slice $m_i \in \mathbb{R}^d$ in M can be seen as a building block selected to form the relation vector for each feedback type according to its attention vector. In other word, the latent relation vector for feedback type τ denoted as r^τ is a weighted representation of memory matrix M, which can be generated as:

$$r^\tau = \sum_{i=1}^{N} \omega_i^\tau m_i \tag{3}$$

Through the architecture introduced above, a relation set R consisting of multiple types of relation vectors for a specific user-item pair forms.

4.5 Optimisation

Inspired by the multi-relational modeling technique of *TransE* [Bordes *et al.*, 2013] in knowledge base embedding, we define the scoring function of MRMN as:

$$s(u, i, \tau) = \|u + r^\tau - i\|_2^2 \tag{4}$$

where $\|\cdot\|_2$ denotes the L_2-norm. The r^τ employed to model the user-item relationship by a translation operating ensures the flexibility and superiority of MRMN in many-to-many recommendation (a user may enjoy many items, and many users can like an item) compared with the vanilla metric learning approach of CML, an ill-posed algebraic system mathematically which minimises the Euclidean distance between positive user-item pair via $\|u - i\|_2^2$.

We adopt the pairwise ranking loss for optimisation. Different from the pointwise loss [Hu *et al.*, 2008] treating unobserved entries as negative samples, the basic idea of pairwise learning [Rendle *et al.*, 2009] is ranking observed entries higher than those unobserved. We also tried to combine point-wise loss and pair-wise loss in a unified framework, like that in [Ding *et al.*, 2018b]. However, it turned out to be about 30% worse than a margin-based approach. In this case, we do not integrate the point-wise loss in this paper. For each positive triplet (u, i, τ) representing u has an interaction of type τ with i, a corrupted triplet (u, j, τ) is sampled according to the rule that u has never interacted with item j. Mathematically, we define the objective function as:

$$\mathcal{L} = \sum_{\tau} \sum_{(u,i,\tau) \in \eta} \sum_{(u,j,\tau) \notin \eta} \phi(s(u,i,\tau) + \lambda^\tau - s(u,j,\tau)) \tag{5}$$

where η is the set of all positive triplets; λ^τ is the margin separating the positive triplets and corrupted ones in terms of feedback τ. The non-linear activation function $\phi(\cdot)$ applied here is the rectified linear unit (*ReLU*) function as we found it performed best empirically. Additionally, since MRMN is end-to-end differentiable, stochastic gradient descent (SGD) is employed to minimise the objective function.

It is worth mentioning that by injecting discriminative λ^τ, we can encode the *strength* deviation corresponding to different feedback types. Here a larger setting of margin value will push the two points of positive sample and negative sample farther away from each other, suggesting that this feedback type is more reliable in depicting user preferences. We will discuss more with experimental results in Subsection 5.5.

5 Experiments

5.1 Datasets

The real-world datasets with multiple user feedback types used for experiments are Tmall[1], Xing[2], and Dianping[3] corresponding to e-commerce, job, and local services recommendation scenarios, respectively. Interactions including *bookmark*, *reply*, and *response* from a recruiter in Xing are regarded equally as positive feedback type to reduce sparsity. For Dianping, ratings on aspects of *overall*, *taste*, *environment*, and *service* are converted into implicit form. The feedback termed as *visited* denotes a user posted a review on an item without rating it. Users with less than 12 and items with less than 16 interactions in Tmall and Dianping are filtered. While for Xing, the threshold is set to 5 for both user and item. Table 1 summarises the statistics of the filtered datasets.

Dataset	Type	#User	#Item	#Inter	Density
Tmall	Purchase*	84k	36k	1.1m	0.04%
	Cart	3.7k	5.4k	7.6k	0.01%
	Collect	53k	34k	582k	0.02%
	Click	84k	36k	9.2m	0.30%
Xing	Positive*	19k	9.6k	201k	0.11%
	Click	18k	9.2k	272k	0.15%
	Hide	2.0k	5.2k	33k	0.02%
Dianping	Overall*	129k	27k	2.2m	0.06%
	Taste	60k	12k	214k	0.01%
	Environment	53k	14k	169k	0.01%
	Service	57k	15k	175k	0.01%
	Visited	129k	27k	4.6m	0.13%

Table 1: Statistics of the datasets. k indicates thousand; m indicates million; and *Inter* indicates interaction. '*' means primary feedback.

[1] https://www.tmall.com

[2] http://www.recsyschallenge.com/2017/

[3] http://www.dianping.com/

Methods		Tmall		Xing		Dianping	
Type	Name	HR@10	NDCG@10	HR@10	NDCG@10	HR@10	NDCG@10
Primary	MF-BPR	0.1404	0.0693	0.6203	0.5653	0.5784	0.3321
	CML	0.3213	0.1763	0.6611	0.6012	0.5965	0.3556
	CML(ASP)	0.3472	0.2014	0.8556	0.7407	0.6067	0.3548
	LRML	0.3248	0.1821	0.5858	0.5283	0.5910	**0.3635**
	LRML(ASP)	0.4112	0.2430	0.8532	0.7364	0.5831	0.3585
Multiple	TCF	0.3065	0.1776	0.7017	0.6263	0.5273	0.3211
	MFPR	0.4534	0.2783	0.7840	0.6946	0.5836	0.3579
	MR-BPR	0.4416	0.2566	0.8548	0.7425	0.5975	0.3477
	MC-BPR	0.3457	0.1931	0.7204	0.6405	0.5348	0.3302
	VALS	0.3948	0.2486	0.7550	0.6842	0.5233	0.3329
	MRMN	**0.5063**	**0.3042**	**0.8604**	**0.7443**	**0.6132**	0.3614

Table 2: Experimental results on the datasets. The best performance is in boldface. *ASP* means we treat all types of (positive) feedbacks as primary feedback to dispel doubts on the data size inequality caused by involving more feedback types for a fair comparison.
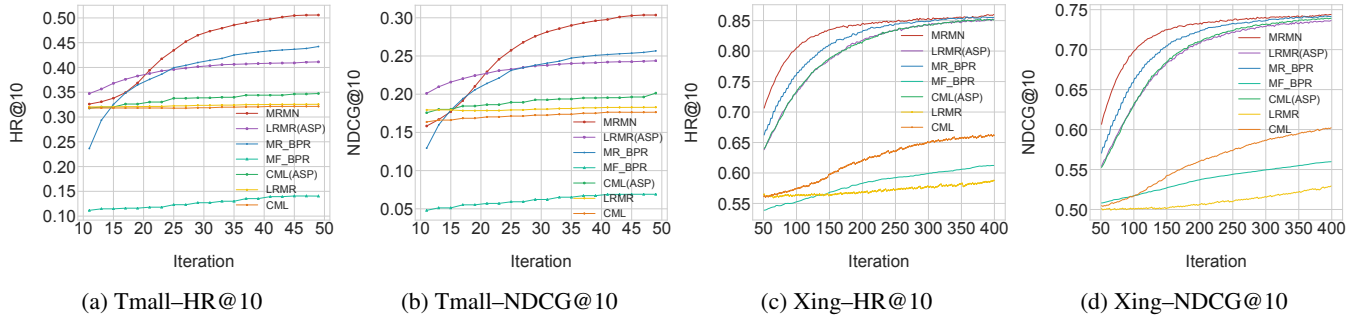

(a) Tmall–HR@10 (b) Tmall–NDCG@10 (c) Xing–HR@10 (d) Xing–NDCG@10

Figure 3: Performance of MRMN and selected baselines w.r.t. the number of iterations on Tmall and Xing.

5.2 Evaluation Methodology

We adopt the *leave-one-out* protocol for model evaluation. Each user's interactions with items are firstly sorted by the timestamps ascendingly. Then the last two records are held out as validation and test set, while the remaining serve as training data. We randomly sample 100 items that the user has not interacted with as negative samples. There is an additional constraint for Dianping that negative samples are selected randomly from places within 3 miles from the positive one. For the two evaluation metrics, *Hit Ratio at K (HR@K)* only considers whether the ground truth is ranked among the top K items; while *Normalised Discounted Cumulative Gain at K (NDCG@K)* is a position-aware ranking metric.

5.3 Baselines

According to whether considering additional types of user feedback, baselines are split into two groups. *Primary* group that only employ primary feedback include:

- **MF-BPR** [Rendle *et al.*, 2009] It is the MF model using a pairwise ranking loss for optimisation. This algorithm is tailored to implicit feedback recommendations.
- **CML** [Hsieh *et al.*, 2017] It is a metric learning approach minimising the Euclidean distance between the user vector and item vector.
- **LRML** [Tay *et al.*, 2018] This is a metric learning technique employing a memory network to learn the latent relationship between the user and item.

Multiple group dealing with multiple feedback types include:

- **TCF** [Pan *et al.*, 2010] This is a transfer learning approach that is capable of transferring knowledge from auxiliary feedback to the primary feedback.
- **MFPR** [Liu *et al.*, 2017] It is a method that integrates multiple implicit feedback types in the SVD++ manner.
- **MR-BPR** [Singh and Gordon, 2008] BPR framework is applied to collective matrix factorization to make it capable of handling multiple types of implicit data.
- **MC-BPR** [Loni *et al.*, 2016] This is a method that samples positive and negative items from multiple relations.

5.4 Experimental Results

The experimental results of MRMN along with the baselines for *HR@10* and *NDCG@10* are shown in Table 2. As can be observed, MRMN obtains the best performance across all the datasets and metrics in general. In Figure 3, we also present the recommendation performance of MRMN and six selected baselines of each iteration on *Tmall* and *Xing*. Results on *Dianping* show the same trend and are omitted due to space limitations. In the figure, with the increase of iteration times, the performance of MRMN is gradually improved until convergence is reached. The most effective updates occur in the first 30 iterations and 150 iterations for *Tmall* and *Xing*, respectively. As the architecture of MRMN fuses memory module and multi-relational metric learning, the following will provide a detailed breakdown of our experimental results.

Single Feedback against Multiple Feedback Types

As it shows in Table 2, multi-feedback methods as a whole provide clearly better performance than single-feedback approaches, especially on datasets of *Tmall* and *Xing*. This is probably due to the fact that *Tmall* and *Xing* consist of binary implicit feedbacks in multiple types, while *Dianping* reflects more about whether a user likes an item in various aspects. Among multi-feedback algorithms, our proposed MRMN performs best that it outperforms the best baseline by about 5% on *Tmall*, and around 1% on the other two datasets in terms of the *HR* metric. This phenomenon reveals the effectiveness of integrating multiple types of user feedback in recommender systems.

Contribution of Relational Metric Learning

The models applying latent relational learning are MRMN and LRML. Compared with CML, LRML learns a single transition vector for each pair of user and item vectors, while MRMN enables multi-relational learning to identify much more complex interactions existing between them. As we can see from Table 2 that LRML offers slightly better performance than CML, and MRML further boosts the performance significantly. This observation is particularly evident on *Tmall*, where the improvement brought by multi-relational metric learning is around 10% on both *HR* and *NDCG*. These findings reveal that the application of multi-relational learning does enhance the recommendation performance since MRML outperforms both of its single-relational counterpart LRML and non-relational learning version CML.

Contribution of Memory Mechanism

The high performances provided by memory augmented networks of MRML and LRML also illustrate the effectiveness of the memory mechanism in recommendation tasks. Moreover, we can detect a hint of memory mechanism's superiority in multi-relational recommendations as the MRMN model performs strikingly better than LRML on almost all the metrics and datasets.

In a nutshell, the impressive performance of MRMN portrays the successful integration of the memory component and attention mechanism over existing metric learning-based algorithms and multi-feedback models. Additionally, it can be expected to show better performance for a larger dataset with more complex user-item interactions.

5.5 Hyper-parameter Investigation

The number of memory slots and key slots N in MRMN is tuned amongst $\{5, 10, 20, 50\}$, and 10 works best in most cases. The dimension of user and item embeddings d is tuned amongst $\{20, 50, 75, 100\}$ and set to 20 eventually. To explore which combination of λ^{τ} on different datasets can provide best performance, and to further investigate the relationships between various types of feedback in a deep sense, we run experiments with different λ^{τ} settings on our datasets and list the results of *Tmall* in Table 3 as an example. In the table, the first set of margin values is determined intuitively according to feedback types' importance levels reflected on how likely a user enjoys an item. In the second experiment, the same value is allocated to all the feedback types. In the last row, we reverse the order in the first set, allocating a smaller
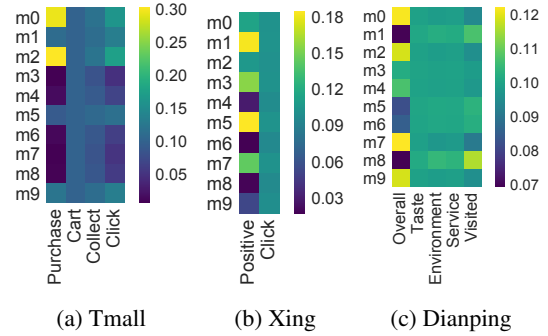

(a) Tmall   (b) Xing   (c) Dianping

Figure 4: Attention weights over memory slices for feedback types.

margin value to a more important feedback type. As the performances of the three settings suggest, assigning higher margin value to more 'important' feedback gives better recommendations on *Tmall*. Similar patterns are also found on *Xing* and *Dianping*. This discussion is not to affirm that setting margin values by experience is always the best. Instead, we show how the MRMN model can be applied to detect the underlying relationships between multiple feedback types.

$[\lambda^{purchase}, \lambda^{cart}, \lambda^{collect}, \lambda^{click}]$	HR@10	NDCG@10
[0.2,0.15,0.1,0.05]	0.5063	0.3012
[0.1,0.1,0.1,0.1]	0.4648	0.2748
[0.05,0.1,0.15.0.2]	0.3885	0.2240

Table 3: Impacts of margin values on MRMN's performance.

5.6 Relation Visualisation

The attention mechanism in MRMN enables us to visualise the weighted importance of memory slices for multiple feedback types. As can be seen in Figure 4, three datasets present different patterns, indicating that feedback types have different selection rules across memory slices in multi-relational vector learning. Furthermore, the attention weights of popular interaction types are more diverse compared with other minority types. For instance, in Figure 4a, *Purchase* and *Click* have more diverse attention weights compared with *Cart* and *Collect*. Similar pattern also appears in Figures 4b and 4c. This finding further proves the necessity of multi-relational detection between each user-item pair that learning an overall relation vector is not fine-grained enough.

6 Conclusion

We introduce a novel end-to-end architecture named MRMN, for recommendation with multiple types of user feedback. MRMN is augmented with external memory and neural attention mechanism to capture fine-grained user preference across various interaction space. Comprehensive experiments under multiple configurations demonstrate the proposed architecture's significant improvements over competitive baselines. Qualitative analyses of the attention weights bring insights into the multi-relational learning process and suggest the existence of complex relationships between a pair of user and item, effectively captured by our MRMN model.

References

[Bordes *et al.*, 2013] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. In *Advances in neural information processing systems*, pages 2787–2795, 2013.

[Chen *et al.*, 2018] Xu Chen, Hongteng Xu, Yongfeng Zhang, Jiaxi Tang, Yixin Cao, Zheng Qin, and Hongyuan Zha. Sequential recommendation with user memory networks. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, pages 108–116. ACM, 2018.

[Ding *et al.*, 2018a] Jingtao Ding, Fuli Feng, Xiangnan He, Guanghui Yu, Yong Li, and Depeng Jin. An improved sampler for bayesian personalized ranking by leveraging view data. In *Companion of the The Web Conference 2018 on The Web Conference 2018*, pages 13–14. International World Wide Web Conferences Steering Committee, 2018.

[Ding *et al.*, 2018b] Jingtao Ding, Guanghui Yu, Xiangnan He, Yuhan Quan, Yong Li, Tat-Seng Chua, Depeng Jin, and Jiajie Yu. Improving implicit recommender systems with view data. In *IJCAI*, pages 3343–3349, 2018.

[Ebesu *et al.*, 2018] Travis Ebesu, Bin Shen, and Yi Fang. Collaborative memory network for recommendation systems. *arXiv preprint arXiv:1804.10862*, 2018.

[Ekstrand *et al.*, 2011] Michael D Ekstrand, John T Riedl, Joseph A Konstan, et al. Collaborative filtering recommender systems. *Foundations and Trends® in Human–Computer Interaction*, 4(2):81–173, 2011.

[He and McAuley, 2016] Ruining He and Julian McAuley. Vbpr: Visual bayesian personalized ranking from implicit feedback. In *AAAI*, pages 144–150, 2016.

[Hsieh *et al.*, 2017] Cheng-Kang Hsieh, Longqi Yang, Yin Cui, Tsung-Yi Lin, Serge Belongie, and Deborah Estrin. Collaborative metric learning. In *Proceedings of the 26th International Conference on World Wide Web*, pages 193–201. International World Wide Web Conferences Steering Committee, 2017.

[Hu *et al.*, 2008] Yifan Hu, Yehuda Koren, and Chris Volinsky. Collaborative filtering for implicit feedback datasets. In *Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on*, pages 263–272. Ieee, 2008.

[Huang *et al.*, 2017] Haoran Huang, Qi Zhang, Xuanjing Huang, et al. Mention recommendation for twitter with end-to-end memory network. In *Proc. IJCAI*, volume 17, pages 1872–1878, 2017.

[Koren, 2008] Yehuda 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.

[Li *et al.*, 2018] Zhi Li, Hongke Zhao, Qi Liu, Zhenya Huang, Tao Mei, and Enhong Chen. Learning from history and present: next-item recommendation via discriminatively exploiting user behaviors. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1734–1743. ACM, 2018.

[Lian *et al.*, 2018] Jianxun Lian, Xiaohuan Zhou, Fuzheng Zhang, Zhongxia Chen, Xing Xie, and Guangzhong Sun. xdeepfm: Combining explicit and implicit feature interactions for recommender systems. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1754–1763. ACM, 2018.

[Liu *et al.*, 2017] Jian Liu, Chuan Shi, Binbin Hu, Shenghua Liu, and S Yu Philip. Personalized ranking recommendation via integrating multiple feedbacks. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 131–143. Springer, 2017.

[Loni *et al.*, 2016] Babak Loni, Roberto Pagano, Martha Larson, and Alan Hanjalic. Bayesian personalized ranking with multi-channel user feedback. In *Proceedings of the 10th ACM Conference on Recommender Systems*, pages 361–364. ACM, 2016.

[Pan *et al.*, 2010] Weike Pan, Evan Wei Xiang, Nathan Nan Liu, and Qiang Yang. Transfer learning in collaborative filtering for sparsity reduction. In *AAAI*, volume 10, pages 230–235, 2010.

[Pazzani and Billsus, 2007] Michael J Pazzani and Daniel Billsus. Content-based recommendation systems. In *The adaptive web*, pages 325–341. Springer, 2007.

[Rendle *et al.*, 2009] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. In *Proceedings of the twenty-fifth conference on uncertainty in artificial intelligence*. AUAI Press, 2009.

[Singh and Gordon, 2008] Ajit P. Singh and Geoffrey J. Gordon. Relational learning via collective matrix factorization. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 650–658. ACM, 2008.

[Tang *et al.*, 2016] Liang Tang, Bo Long, Bee-Chung Chen, and Deepak Agarwal. An empirical study on recommendation with multiple types of feedback. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 283–292. ACM, 2016.

[Tay *et al.*, 2018] Yi Tay, Luu Anh Tuan, and Siu Cheung Hui. Latent relational metric learning via memory-based attention for collaborative ranking. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web*, pages 729–739. International World Wide Web Conferences Steering Committee, 2018.

[Weston *et al.*, 2015] Jason Weston, Sumit Chopra, and Antoine Bordesm. Memory networks. In *ICLR*, 2015.

[Zhou *et al.*, 2019] Xiao Zhou, Cecilia Mascolo, and Zhongxiang Zhao. Topic-enhanced memory networks for personalised point-of-interest recommendation. *arXiv preprint arXiv:1905.13127*, 2019.