

## 《TKDE 2022: A Survey on Accuracy-oriented Neural Recommendation》分享整理

IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING

1

# A Survey on Accuracy-oriented Neural Recommendation: From Collaborative Filtering to Information-rich Recommendation

Le Wu *Member, IEEE*, Xiangnan He *Member, IEEE*, Xiang Wang *Member, IEEE*,  
Kun Zhang *Member, IEEE*, and Meng Wang, *Fellow, IEEE*

**Abstract**—Influenced by the great success of deep learning in computer vision and language understanding, research in recommendation has shifted to inventing new recommender models based on neural networks. In recent years, we have witnessed significant progress in developing neural recommender models, which generalize and surpass traditional recommender models owing to the strong representation power of neural networks. In this survey paper, we conduct a systematic review on neural recommender models from the perspective of recommendation modeling with the accuracy goal, aiming to summarize this field to facilitate researchers and practitioners working on recommender systems. Specifically, based on the data usage during recommendation modeling, we divide the work into collaborative filtering and information-rich recommendation: 1) *collaborative filtering*, which leverages the key source of user-item interaction data; 2) *content enriched recommendation*, which additionally utilizes the side information associated with users and items, like user profile and item knowledge graph; and 3) *temporal/sequential recommendation*, which accounts for the contextual information associated with an interaction, such as time, location, and the past interactions. After reviewing representative work for each type, we finally discuss some promising directions in this field.

**Index Terms**—Recommendation Survey, Deep Learning, Neural Networks, Neural Recommendation Models

Arxiv: <https://arxiv.org/pdf/2104.13030.pdf>

推荐系统的研究主要从上世纪九十年代开始，开发了许多基于内容和协同过滤的方法。在 Netflix 比赛之后，矩阵分解的方法成为了 2008 年到 2016 年间的主流模型。然而受限于因子分解模型天然的线性性质，其在处理大规模以及复杂数据上的性能表现有所欠缺，比如复杂的用户交互、物品侧可能包含复杂的语义信息等。2010 年左右深度学习技术开始在计算机视觉、自然语言处理、语音识别等领域产生了革命性的影响，其也为推荐系统的研究与实践打开了新的思路。这篇综述总结了近年来深度学习相关的推荐系统研究内容，根据使用数据的不同，作者将推荐系统划分的模型进行了划分。

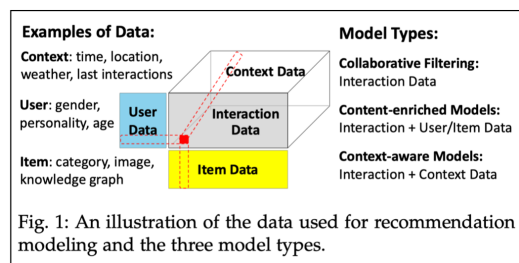
## 任务定义

# Recommender System

### 问题定义:

Given the data  $D_u$ ,  $D_i$ , and  $D_c$  to describe the user  $u$ , item  $i$  and context  $c$ , learning the prediction function  $f$  to estimate the likelihood that a user  $u$  will favor an item  $i$  under the context  $c$ .

$$\hat{y}_{u,i,c} = f(D_u, D_i, D_c)$$



### 模型分类:

- Collaborative filtering
- Content-enriched models
- Context-aware Models

推荐系统的任务可以定义为:

$$\hat{y}_{u,i,c} = f(D_u, D_i, D_c)$$

即给定 user、item 和 context 数据, 学习一个预测函数  $f$ , 拟合 user  $u$  在 context  $c$  的条件下会对 item  $i$  感兴趣的可能性。

根据数据来源不同, 可以将模型分为三类:

- 协同过滤, 主要利用 user 和 item 的 interaction 数据, 是推荐系统中最为主要的研究方向;
- Content-riched 模型, 在 CF 基础上利用了 user 和 item 相关的 side information, 如 user profile, item KG, 但同样的, context 信息也没有利用到。
- 时序/序列推荐, 利用了 interaction 的上下文信息, 如时间、地点、历史 interaction。



虽然进行如此划分, 但是很多情况下, 都可以通过对模型的组件进行简单的调整, 来适应另一种场景。例如 context 信息可以视为 user 或 item 的一种扩充, 应用 content-enriched model 同样可以实现 context-aware。

# Collaborative Filtering Models

## 协同过滤算法：

基于User-Item交互历史，学习一个更优质的User/Item表征，从而可以根据优质的User/Item表征得到用户对每个商品的兴趣。

## 改进方向：

- 如何更好学习user/item表征  **Representation Learning**
- 如何更好为交互历史建模  **Interaction Modeling**

协同过滤算法的思路是利用所有用户的协同行为来预测目标用户的行为，最主要特征是基于 User-Item 交互历史，学习一个更优质的 User/Item 表征，从而可以根据优质的 User/Item 表征得到用户对每个商品的兴趣。

基于协同过滤的改进主要有两个方向，1、如何更好学习 user/item 表征，2、如何更好为交互历史建模。

## User/Item 表征学习

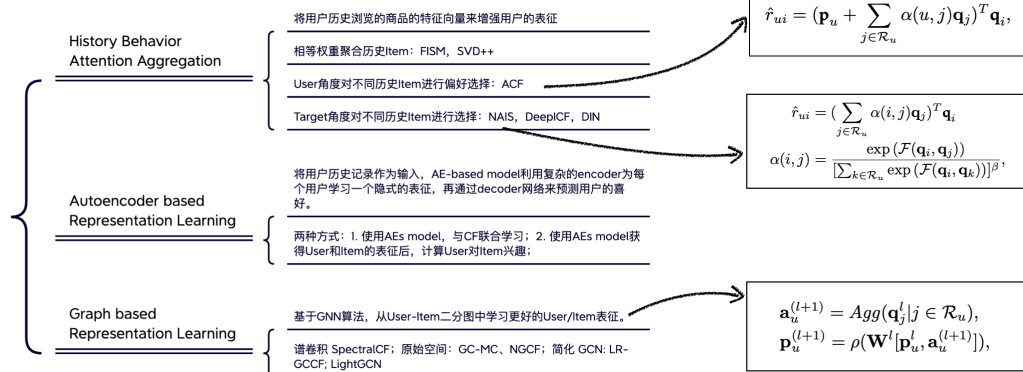
传统的 User/Item 表征学习最经典的就是矩阵分解算法，但是这些传统的表征学习算法能力有限，因此近年来 User/Item 表征学习主要是研究通过深度学习技术对 User/Item 表征进行增强。

表征学习的主要挑战在于，对于庞大的 item 集合，用户行为是有限的，因此在协同过滤算法中，user-item interaction 存在稀疏性，难以学习准确的 user/item embedding。

该论文将相关改进分为三个部分，即 History Attention，Autoencoder Models 和 Graph Learning。

# Collaborative Filtering Models

## Representation Learning



### History Behavior Attention Aggregation Models

这一类方法主要是将用户历史浏览的商品的特征向量来增强用户的表征。FISM 算法通过将用户交互过的 item embedding 池化后，作为 user embedding；SVD++ 则将这样的表征与 user 本身的 embedding 进行相加融合，来作为最终的表征；考虑到不同的历史交互 item 对于 user 建模有不同影响，因此提出了 ACF。ACF 算法采用 Attention 机制，根据商品表征和用户表征，计算不同商品的权重系数，进而将用户浏览过的历史商品特征向量与用户特征向量相加，作为用户侧的最终表征，最终得到预测分数。

$$\hat{\mathbf{r}}_{ui} = \left( \mathbf{p}_u + \sum_{j \in \mathcal{R}_u} \alpha(u, j) \mathbf{q}_j \right)^T \mathbf{q}_i,$$

$$\alpha(u, j) = \frac{\exp(\mathcal{F}(\mathbf{p}_u, \mathbf{q}_j))}{\sum_{j' \in \mathcal{R}_u} \exp(\mathcal{F}(\mathbf{p}_u, \mathbf{q}_{j'}))}$$

NAIS 算法则提出，对于不同的 target item，历史 item 的作用也会不同，因此采用 target-item aware 的注意力系数的计算方式。

$$\hat{\mathbf{r}}_{ui} = \left( \sum_{j \in \mathcal{R}_u} \alpha(i, j) \mathbf{q}_j \right)^T \mathbf{q}_i$$

$$\alpha(i, j) = \frac{\exp(\mathcal{F}(\mathbf{q}_i, \mathbf{q}_j))^j}{[\sum_{k \in \mathcal{R}_u} \exp(\mathcal{F}(\mathbf{q}_i, \mathbf{q}_k))]^\beta}$$

DeepICF、DIN 等经典算法也是针对交互历史进行表示学习。

### *Autoencoder based Representation Learning*

Autoencoder Models 主要是利用 Autoencoder 算法，根据用户的历史行为学习一个用户表征。将用户历史记录作为输入，AE-based model 利用复杂的 encoder 为每个用户学习一个隐式的表征，再通过 decoder 网络来预测用户的喜好。

使用 AE 的方式有两大类，一种是将 AE 直接与 CF 进行结合，进行联合的学习，这可以视为是使用 AE 方法对历史行为进行建模，另一种是分别使用 AE 来对 user 和 item 的表征进行学习，之后在对两者表征使用内积等方法，来计算 user 对 item 的兴趣。

### *Graph based Representation Learning*

如果把 user 和 item 是为节点，那么可以根据 interaction 来构建出一个二分图。在这个 user-item 二分图中，一阶信息也就是直接的交互历史。进一步的，可以从图上去学习高阶的关联性。例如，user 的二阶关联性表示有共同交互历史的相似用户。

使用图学习的方法可以很好挖掘这些信息。Graph Learning 主要是基于 GNN 算法，从 User-Item 二分图中学习更好的 User/Item 表征。

$$\begin{aligned}\mathbf{a}_u^{(l+1)} &= Agg(\mathbf{q}_j^l \mid j \in \mathcal{R}_u) \\ \mathbf{p}_u^{(l+1)} &= \rho\left(\mathbf{W}^l \left[\mathbf{p}_u^l, \mathbf{a}_u^{(l+1)}\right]\right)\end{aligned}$$

近年来，有学者提出，这种基于 GNN 的方式，缺少 user 和 item 的特征。同时，简化的 GNN 方法如 LR-GCCF、LightGCN 等方法被提出。

# Collaborative Filtering Models

## Interaction Modeling

常见方式：内积（Inner product）

存在问题：

- 1. 内积操作并不满足三角不等式
- 2. 线性建模方式可能无法捕捉高阶交互关系

Category	Modeling Summarization	Models
Inner Product	$\hat{r}_{ui} = \mathbf{p}_u^\top \mathbf{q}_i$	Most models
Distance Modeling	Euclidean distance $d_{ui} = \ \mathbf{p}_u - \mathbf{q}_i\ _2^2$	CML [45]
	Nearby translation $\tilde{d}_{ui} = \beta_j - d(\mathbf{q}_j + \mathbf{p}_u, \mathbf{q}_i)$	TransRec [46]
	Memory enhanced Translation $\tilde{d}_{ui} = \ \mathbf{p}_u + \mathbf{E} - \mathbf{q}_i\ _2^2$	LRML [47]
	Distance in Hyperbolic Space	HyperML [48]
Neural Networks	$\hat{r}_{ui} = MLP(\mathbf{p}_u    \mathbf{q}_i)$	NCF [49] et al.
	$\hat{r}_{ui} = CNN(\mathbf{p}_u \otimes \mathbf{q}_i)$	ONCF [50] et al.
	Autoencoder based reconstruction $\ \mathbf{r}_i - dec(enc(\mathbf{r}_i))\ _2^2$	AutoRec [27], CDAE [28] et al.

传统的 User/Item 交互建模即为内积操作，但是内积操作存在两个问题，第一个问题是并不是一个合理的距离测度，因此可能会带来对于相似度的学习不准确的问题，违背所谓的“triangle-inequality”，也就是如果用户同时购买了商品*i*和*j*，该建模方式无法保证 Embedding 向量 $\mathbf{q}_i$ 和 $\mathbf{q}_j$ 相邻。第二个问题是该线性建模方式可能无法捕捉高阶交互关系。

针对第一个问题，一个有效的解决方案是使用基于距离的目标函数。例如，CML 算法将 User 和 Item 之间的 Embedding 距离作为目标函数。因此，如果用户同时购买了两个商品，那么这两个商品的 Embedding 和用户的 Embedding 应该是近似的。

$$d_{ui} = \|\mathbf{p}_u - \mathbf{q}_i\|_2^2$$

TranRec 采用翻译模型 TransE 类似的思想  $\mathbf{q}_i + \mathbf{p}_u \approx \mathbf{q}_j$ 。LRML 则引入了关系向量  $\mathbf{e}$ ，来作为 user 和 item 表征的一个关联，而非直接将产生交互的 user 和 item 学习一个接近的表征  $s_{ui} = \|\mathbf{p}_u + \mathbf{e} - \mathbf{q}_i\|_F^2$ 。其中， $\mathbf{e}$ 是根据 $\mathbf{p}_u$ 和 $\mathbf{q}_i$ 经过网络得到。

针对第二个问题，最简单的解决方式是使用 Deep Learning 模型进行 User-Item 交互建模。一种方法是直接拼接后使用 MLP，如 NCF；一种是首先经过外积，获得 user-item 的高阶相关性，然后使用 CNN-based 模型来捕获高阶的一致性。但这也 会消耗更高的复杂度和时间成本。另一种方式是采用 AEs 模型，和上一小节提到的类似。

## Content-Enriched Recommendation

CF 的方法仅仅针对 Collaborative signal，也就是 User 对 Item 的行为特征进行建模，但忽略了内容语义上的相关性。为了增强表示学习的效果，除了交互行为外，还可以利用多种辅助数据，这些数据分为两类：

**Content-based information**，即 user 和 item 的特征，文本、图片、音视频等描述（item 标签、描述，用户评论），user 社交网络，知识图谱等内容；

**Context-aware data**，即 user 的决策环境，与 item 发生交互的上下文，包括时间、地理位置，或其他相关数据（速度、天气等）。本文只讨论时序数据。

针对 **Content-based information**，下面按照五类内容进行介绍：User 和 Item 通用特征，文本内容，多媒体内容，社交网络，知识图谱

### Content-Enriched Recommendation

#### 分类

- General Feature Interactions, User/Item通用特征
- Modeling Textual Content, 文本输入
- Modeling Multimedia Content, 图片、音视频输入
- Modeling Social Network, 用户社交网络
- Modeling Knowledge Graph, 外部知识图谱信息

## General Feature Interactions

对于 general features，核心的思路是，如何去挖掘出特征之间更为高阶和更为复杂的关系。

因子分解机 FM。因为特征通常十分稀疏，FM 将每种特征嵌入表示为隐式向量  $v_i$ ，并且在特征间建立二阶的交互。FM 构建了二阶的特征交互，并且降低了基于 Embedding 计算特征相似度的参数量。FM 有进一步的改进，如加入场的概念，或建立更高阶的特征交互。FFM 在 FM 基础上，引入了“Filed”概念，按一定规则划分出多个 Filed 后，每个特征在不同 Filed 中都有对应隐向量。



基于 MLP 的高阶建模。通过将特征进行 **embedding** 表示，利用 MLP 来隐式地进行高阶建模。这个过程中，MLP 的结构被认为是一个黑盒，输出结果的可解释性很差，无从得知所学习到的特征交互。这个方向主要研究内容是：1、提高训练效率；2、加入不同模型结构，来更好捕获特征交互，如 DeepCrossing、NFM、PNN 等。

利用 Cross Network 进行 K 阶建模。在 MLP 基础上，通过设计不同的交叉网络，来建模 K 阶特征。如 DCN 通过交叉网络自动显式地对特征交互进行建模，包含了特征在有限阶的所有叉乘组合。然而，交叉网络在进行交叉时是 **bit-wise** 的方式（每层特征都是进行内积，是元素级的交互），最终的输出只是输入的标量倍，这限制了模型的表达能力，xDeepFM 则进一步实现了 **vector-wise** 的特征交互。

利用树模型进行建模。树模型天然能够自动组合多个特征，这种情况下树的深度决定了阶数，代表性算法为 TEM。

### Modeling Textual Content & Multimedia Content

推荐系统中的文本输入可以分为两类：（1）User 或 Item 相关的内容描述，如文章摘要，用户简介。（2）user-item pair 相关内容。如 user 对 item 标注的 tag，user 对 item 的评价。这一类数据一般处理方式是，分别与 user 和 item 进行关联，这会退化为第一类数据

对于文本数据，根据不同的建模方式，可以分为：基于自编码模型，word embedding，注意力模型，用于推荐的文本解释。

对于文本和多媒体内容，利用的方式都是类似，使用 NLP 和 CV 的技术，对 item 或 user 信息进行补充和增强。

### 社交关系建模

社交关系建模主要是根据用户社交关系增强用户的 Embedding 向量。该类建模算法主要分为基于一阶关系和基于高阶关系的社交关系建模算法。基于一阶关系的建模算法只考虑每个用户的直接邻居节点。该算法首先通过社交关系模型得到社交关系特征，随后将用户特征与关系特征进行融合得到最终的用户特征  $p_u$ 。基于高阶关系的建模算法主要利用图神经网络算法聚合用户的高阶特征。这一类算法中，经典的算法是 DiffNet。

### 知识图谱建模

知识图谱建模主要是根据外部的知识图谱信息对推荐系统进行性能增强。目前该类算法主要分为三种算法，即基于元路径的算法，基于正则化的算法和基于图神经网络的算法。

基于元路径的算法按一定规则，提取元路径并从中抽取用户和商品之间的关系，作为商品推荐时的特征。对于一条用户-商品关联路径  $path(u, i)$ ，该类算法使用神经



网络计算路径上每个节点的 Embedding 向量，并使用神经网络聚合路径上的所有 Embedding。

基于正则化的算法主要是利用知识图谱中三元组中的信息。一种方法是将基于知识图谱计算得到的特征向量  $f_{KGE}(i)$  与原始的商品特征向量  $f_{Embed}(i)$  加，得到最终的特征向量  $q_i$ 。

基于图神经网络的算法利用图神经网络算法将知识图谱和用户历史行为信息进行融合，从而增强特征向量表示。例如，KGAT 算法将知识图谱和用户-商品交互二分图合并成一个图，并利用图神经网络得到商品特征向量  $p_u$ 。

## Temporal/Sequential Models

# TEMPORAL/SEQUENTIAL MODELS

### 基于时序/序列的方法：

利用用户对物品的交互矩阵的基础上添加上下文信息并进行建模的方法

- **基于时序的推荐：**对于一个user和一个item，user-item相关的交互行为可以表示为四元组 (user, item, rating, timestamp)，基于时序的推荐侧重于建模用户在时间维度上的动态性。
- **基于会话的推荐：**在一个特定Session中，一个user和一系列item进行交互，产生不同行为（如加入购物车，浏览等），而用户侧信息可能缺失（匿名用户，未登录等情况），基于会话的推荐建模序列间物品与物品间的交互模式，一般不包含用户登录相关的信息。
- **基于时序和会话的推荐：**可视为上两种信息的结合，给出user不同时间的多个session。

常用方法：

- 大多方法集中于user和item的顺序信息，利用序列模型（如RNN、memory network等）来捕获user偏好演化的趋势
- GNN可以处理不同粒度的user和item的交互

基于时序/序列的方法是指在利用用户对物品的交互矩阵的基础上添加上下文信息并进行建模的方法。其中，在上述方法的基础上考虑了交互矩阵的时效性，即用户的偏好是动态变化的。因此基于时序/序列的方法旨在建模用户动态偏好以及随时间变化的序列模式。基于此，主要分为基于时序的推荐方法、基于会话的推荐方法以及基于时序/会话混合的推荐方法。

Temporal-based 推荐，对于一个 user 和一个 item，user-item 相关的交互行为可以表示为四元组 (user, item, rating, timestamp)，基于时序的推荐侧重于建模用户在时间维度上的动态性。

Session-based 推荐，在一个特定 Session 中，一个 user 和一系列 item 进行交互，产生不同行为（如加入购物车，浏览等），而用户侧信息可能缺失（匿名用户

户，未登录等情况），基于会话的推荐建模序列间物品与物品间的交互模式，因此其一般是不会存在用户登录相关的信息。

Temporal-based 类算法和 Session-based 类算法均根据 RNN/CNN/GNN 等算法对用户行为序列进行建模。该类算法中一个典型的代表是 RRN 算法，对于时间点  $t-1$  的用户特征向量  $p_u^{t-1}$  和时间点  $t$  的用户评分行为  $x_u^{t-1}$ ，用户在时间点  $t$  的特征向量  $p_u^t$  可以通过 RNN 计算得到。

大多数现有的工作集中于 user 和 item 的顺序信息，利用序列模型（如 RNN、memory network 等）来捕获 user 偏好演化的趋势。主要挑战在于识别长期和短期的时序兴趣点，以及在没有用户 ID 信息的情况下识别全局和局部的兴趣点。由于 GNN 可以处理不同粒度的 user 和 item 的交互，因此使用的也越来越多。

## Future Directions

### Future Directions

#### Basis: Recommendation Benchmarking

如何设计一个大型基准推荐数据集，跟踪最新的推荐问题并更新SOTA方案以进行比较。

#### Models: Graph Reasoning & Self-supervised Learning

图是表示各种推荐场景的普遍结构，如交互数据可以转化为二部图，以及引入外部KG；自监督学习可以从有限的User-Item交互数据中，获取有效数据，来帮助更好的推荐。

#### Evaluation: Multi-Objective Goals for social good recommendation

考虑解释性长尾项目、利益相关者、社会公平在推荐系统中的影响，

#### Discussion: Reproducibility

#### References:

[1]: Wu, Le, et al. "A Survey on Accuracy-oriented Neural Recommendation: From Collaborative Filtering to Information-rich Recommendation." IEEE Transactions on Knowledge and Data Engineering (2022).

[2]: [A Survey on Neural Recommendation](#)can 思维导图

[3]: [推荐领域，你推荐哪些综述性的文章？ - 震灵的回答 - 知乎](#)