Learning from the Data Heterogeneity for Data Imputation

by

Xu Liu

A Dissertation Presented in Partial Fulfillment
of the Requirement for the Degree
Doctor of Philosophy

Approved June 2021 by the
Graduate Supervisory Committee:

Jingrui He, Co-Chair
Guoliang Xue, Co-Chair
Baoxin Li
Hanghang Tong

ARIZONA STATE UNIVERSITY

August 2021

ABSTRACT

Data mining, also known as big data analysis, has been identified as a critical and challenging process for a variety of applications in real-world problems. Numerous dataset are collected and generated everyday to store the information. The rise in the number of data volumes and data modality has resulted in the increased demand for data mining methods and strategies of finding anomalies, patterns, and correlations within large data sets to predict outcomes. The effective machine learning methods are widely adapted to build the data mining pipeline for various purposes like business understanding, data understanding, data preparation, modeling, evaluation, and deployment.

The major challenges for effectively and efficiently mining big data include (1) data heterogeneity and (2) missing data. Heterogeneity is the natural characteristic of the big data, as the data is typically collected from different sources with the diverse format. The missing value is the most common issue faced by the heterogeneous data analysis, which resulted from variety of factors including the data collecting processing, user initiatives, erroneous data entries, and so on.

In response to these challenges, in this thesis, three main research directions with application scenarios have been investigated: (1) Mining and Formulating Heterogeneous Data, (2) missing value imputation strategy in various application scenarios in both offline and online manner, and (3) missing value imputation for multi-modality data. Multiple strategies with theoretical analysis are presented, and the evaluation of the effectiveness of the proposed algorithms compared with state-of-the-art methods is discussed.

DEDICATION

Dedicated to my mom.

ACKNOWLEDGMENTS

First and foremost I would like to thank my adviser, Dr. Jingrui He, whose expertise, understanding, and patience added considerably to my graduate experience. I have been extremely lucky to have her as my mentor and I will always be indebted to her for teaching me how to become a researcher. Without her guidance and encouragement, this Ph.D. would not have achieved. I am also grateful to my thesis co-chair Dr. Guoliang Xue, my committee members Dr. Baoxin Li, and Dr. Hanghang Tong for their contributions and time serving on my committee.

During my Ph.D. study, I have received support and encouragement from my lab mates and friends. I am especially grateful to all the colleagues at the STAR lab and DATA lab. Thanks to Dawei Zhou, Yao Zhou, Arun Reddy Nelakurthi, Pei Yang, Xue Hu, Jun Wu, Lecheng Zheng, Dongqi Fu, Yikun Ban, Haonan Wang, Ziwei Wu, Wenxuan Bao, Yunzhe Qi, Liangyue Li, Chen Chen, Xing Su, Si Zhang, Boxin Du, Qinghai Zhou, Jian Kang, Lihui Liu, Baoyu Jing, Yuchen Yan, Zhe Xu, Shweta Jain, Rui Zhang, Scott Freitas, Haichao Yu, Ruiyue Peng, Rongyu Lin, Xiaoyu Zhang for their support.

I would finally like to express my gratitude to all my faculty and friends for making me who I am today.

TABLE OF CONTENTS

Page

CHAPTER Page

CHAPTER Page

LIST OF TABLES

LIST OF FIGURES

Figure Page

Figure Page

Chapter 1

INTRODUCTION

This is the era that we are seeing the significant advancements in the big data research and reaping the benefits of numerous industrial applications. Various data sets are collected for data mining purpose in the real-world problems, including the social media entertainment, e-commerce, healthcare, IoT, and so on. According to (Watson 2019), today in the United State, 50% of the IT companies are using big data analytic and over 35% will possibly use the data mining analytic in the future. Global revenues for big data and business analytics (BDA) solutions will reach $189.1 billion in 2019, up 12.0% from 2018. Revenues will continue to rise at this rate from 2018 through 2022, with a five-year prediction of 13.2 percent annual growth, totaling nearly $274.3 billion in sales by 2022. Over 2.5 quintillion bytes of data are created every day by humans and machines, and knowledge extracted from this data is being utilized to perfrom the better consumer behavior analysis and optimize the prices of the produce that greatly profit the e-commerce.

A major characteristic of big data is heterogeneity as the data includes the information from different sources and presented in various formats. The wide range of data types, formats, and contents demonstrates this characteristic. In real-world applications, learning from such a diverse set of data is in high demand. In e-commerce, for example, the recommender system has played a significant role. Customers and products are both linked to a wealth of heterogeneous data. Consumers demonstrate category-specific buying behavior and offer feedback in heterogeneous format, such as numerical rating scores and textual review content. The customers' feedback reveals their purchasing preference, and then the numerous related products

will be recommended to the target customers. Besides the recommender systems, the ailment-specific forums also plays an important role in the social media. These forums are developed for and utilized by people with the same sort of disease in the field of healthcare analytics. Their conversation themes are diverse, ranging from discussions on the condition of sickness to healthy eating plans and mental health management. The heterogeneous data analysis methods are playing an essential role to dive into these problems.

For all these applications, missing data is an inherent and common issue. Things become difficult, or even impossible to solve when missing values cannot be handled. The missing data imputation forms the first critical step to build the data mining pipelines. According to Strike (Strike *et al.* 2001) and Raymond and Roberts (Raymond and Roberts 1987), the missing data can be simply removed without having an impact on the data mining purpose when the data set only has a small amount of missing data, such as less than 10% or 15% for the entire data set. When the missing ratio surpasses 15%, however, (Acuna and Rodriguez 2004) points out that the serious thought must be given to know how to handle the missing data. It is important to note that not every data set follows the same missing value pattern. Small quantities of missing data can sometimes include critical information that cannot be overlooked, such as records having large sums of money spent in the online purchase activities but lacking personal information such as age, income, education, and so on.

Missing value imputation (MVI) is the most often utilized solution to deal with the incomplete data set, as opposed to the case deletion technique. In general, MVI is a procedure in which missing data is replaced with substituted values estimated by statistical or machine learning approaches. For several decades, statistical techniques such as mean/mode and regression have been used for this purpose (Little and Rubin 2019). The machine learning techniques such as k nearest neighbor, artificial neu-

ral network, and support vector machine techniques are being used in recent years (García-Laencina *et al.* 2010). In this thesis, we have studied how to impute the missing value with respect to different application domains and exploring a various state-of-the-art method for the following topics:

T1: Mining and Formulating Heterogeneous Data. The reliable understanding and the robust models/algorithms are critical to analyze the heterogeneous data. The core of the study is to identify the relationship across the heterogeneous data, to adapt and utilize hidden information in domain adaption situations to enhance the data mining performance.

T2: Imputing Missing Value in Multiple Scenario. Missing values arise in multiple real-world scenarios, including recommender systems and healthcare analysis. When customers rating only a few items, it results in the poor feedback for building a recommender system. The MVI methods are reviewed as the potential solution to offer a reconstruction of each user rating to the recommendation, enabling the accuracy and creation of a recommender system. Meanwhile, analyzing healthcare social media is essential to understand how to offer better support to patients with chronic health conditions. The missing clinical biomarker records are imputed by adopting the auxiliary data and exploring the cross-view knowledge when data show view heterogeneity.

T3: Imputation with Data Modality. Real-world data is inextricably linked to one another when it appears in heterogeneous formats (e.g., textual descriptions of the product, images, numerical rating score, and users' contextual review). By incorporating the multi-modality information and leveraging the correlations between modalities, the missing value can be estimated in the application-driven problems, and further alleviate several application-driven problems like cold-start problem and user bias problem.

The dissertation is organized as follows. Chapter 2 discusses the background and the impact of the existing related works of data heterogeneity and missing value imputation. Chapter 3 discusses the proposed works in the scenario of the recommender system. Chapter 4 presents the missing value imputation analysis of healthcare social media. Chapter 5 discusses the work of handling missing value imputation in the scenario of data multi-modality. Chapter 6 concludes the thesis.

Chapter 2

LITERATURE SURVEY

The machine learning methods for data imputation with data heterogeneity has been studied in the last decades for various application background like recommender system (Zhang *et al.* 2017), healthcare social media (Zhu *et al.* 2011; Aittokallio 2010; Harel and Zhou 2007), operation management (Tsikriktsis 2005), questionnaires and surveys (Baraldi and Enders 2010; De Leeuw 2001), and so on. For example, the data collected in a recommender system is usually organized as a matrix with one row per user and one column per item, with each item value representing the corresponding rating score. Most individual users naturally rate only a small set of items after the purchase, while the number of items might range from thousands to millions. The majority of the rating scores are either unnoticed or missing. Meanwhile, missing value issue also troubles the researchers in healthcare analysis, such as a lack of data collecting and reporting (Wells *et al.* 2013). As a result of the voluntary nature of the online user, there is frequently a lack of collection in the data of the disease-focused social network. The missing value is especially prevalent in the medical records, such as the Electronic Health Record (EHR) which is designed for the benefit of clinical and billing companies. The missing value of clinical data may cause fatal consequences when a clinical measurement shows a positive symptom and comorbidity. Almost all of the data recording fields are left blank for effectively handling the missing value, and the desire to deal with incomplete data comes from a variety of places. The studies and approaches for missing value imputation are necessary for their own benefit. The existing missing value imputation approaches, as well as related studies of data heterogeneity, are presented in this chapter.

2.1 TYPES OF MISSING VALUE

The missing data is evolving as a critical issue in real-world applications and disturbing the data analysis (Pedersen *et al.* 2017). The missing value imputation forms the first essential step of many data analysis pipelines to improving the data quality and enhances the model robustness (Young *et al.* 2011). In order to decide how to handle the missing data, it is meaningful to know why they are missing. In general, moving from the simplest type to the most general type, there are three types of missing mechanisms, i.e., missing completely at random (MCAR), missing at random (MAR), and missing not at random (MNAR). Each of the mechanism requires different analysis methods due to their own characteristics as:

• MCAR: In this case, the missing entries occur at completely random as the value of the missing entries has no dependence on the observed knowledge. Data are MCAR type when the probability of missing data on a variable is unrelated to any other measured variable and is unrelated to the variable with missing values itself (Osman *et al.* 2018). For example, online customer information, e.g., gender or contact numbers, is missing from the database, or a tube containing a blood sample is accidentally dropped and breaks, or when questionnaires are unintentionally lost are the typical cases of MCAR. Any kind of data imputation method can be adopted without bringing in the bias risk as no previous constraint specification (Janssen *et al.* 2010). Statistically, the MCAR mechanism can be expressed as (Fox *et al.* 2015):

$$f(M|Y,\phi) = f(M|\phi) \; for \; all \; Y, \; \phi \tag{2.1}$$

where Y and M denote the observed values and missing values respectively. ϕ is an unknown parameter and the function f denotes the conditional probability distribution.

• MAR: Compared with the MCAR, in which no specific constraint exists between

the missing data and observed data, missing at random means the observed variables can partially explain the missing data. There is a dependent relationship between the missing value and other variables. For example, when the blood pressure data is missing at random, the variables of age and gender are considered as the dependent variables to the blood pressure, compared with the variable of Estrogen Receptor (ER) or Progesterone Receptor (PR) that indicate the breast cancer statue. The MAR mechanism can be formally expressed as (Fox *et al.* 2015):

$$f(M|Y,\phi) = f(M|Y_{obs},\phi)\ for\ all\ Y_{miss},\ \phi \tag{2.2}$$

where Y_{obs} and Y_{miss} indicate the observed and missing components of variable Y. The underlying parameter ϕ can be estimated by relating Y_{obs} with other additional information and variables.

• MNAR: In this type of missing mechanism, the unobserved variables are assumed to be related to the values of that variable itself, i.e., the missing value is specifically related to what is missing. There is a direct dependent relationship between the values being missing and the nature of the variable, e.g., it occurs in disease-dedicated social network analysis that those heavy patients may be less likely to disclose their weight. Mathematically, MNAR can be expressed as:

$$f(M,Y|\theta,\phi) = f(Y|\theta)f(M|Y,\phi) \tag{2.3}$$

where θ denotes the distribution of Y estimated from observed information, and ϕ characterizes the distribution of the missing pattern.

Figure. 2.1 shows the latest categorization of the existing missing value imputation mechanisms (Pereira *et al.* 2020). Before this categorization, previous missing value imputation methods handle the missing value by two strategies: 1) Single Value Imputation (**SVI)** and 2) Multiple-Imputation (**MI**). Both of these two strategies

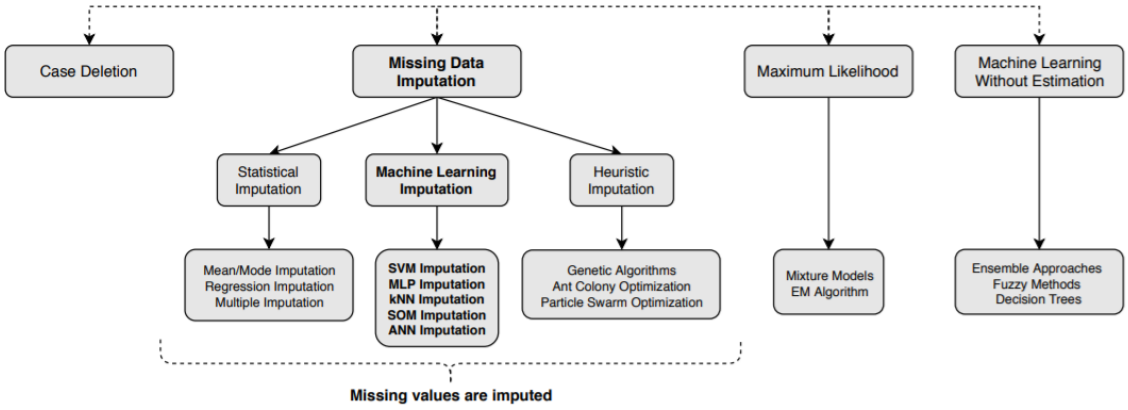


Figure 2.1: Various Mechanisms to Handle Missing Value.

are mainly focused on the MCAR and MAR missing mechanism, as the MNAR leads to the most difficult case when no assumption has been made about what is happening in the missing data. Single value imputation aims at estimating the unknown entries by a single value. For example, the most commonly used method is to replace the missing entries by the overall mean value of the observed entries (Donders *et al.* 2006) or using the most commonly observed values to recover the missing entries (Luengo *et al.* 2012). Another widely used method is regression imputation (Raghunathan *et al.* 2001) (also known as the predicted mean imputation). They are straightforward to understand, but tend to underestimate the diversity of the original data, and also ignoring the correlations between the samples. The single value imputation strategy is leading to the biased imputation result and causes the Type 1 error (i.e., the none existing relation is identified) (Greenland and Finkle 1995), which may not be suitable for many real-world applications (Enders 2010). Compared with single value imputation, the multiple-imputation strategy aims at predicting the missing entries value based on the distribution of observed knowledge, such as the expectation-maximization (EM) based method (Musil *et al.* 2002), matrix factorization (MF)

based method (Lange and Buhmann 2006). The MF-based method (Koren *et al.* 2009) has been proven successful in the Netflix competition. Multivariate imputation by chained equations (MICE) (Buuren and Groothuis-Oudshoorn 2010) is also a popular imputation method that can preserve the observed knowledge during the imputation process.

2.2 HETEROGENEOUS DATA ANALYSIS

Data collected from various sources and domains are exhibiting rich heterogeneity, such as the correlation among multiple views, the relatedness between multiple tasks, and the information across multiple labels. The rich heterogeneity allows us to dive into the data to learn the intrinsic knowledge behind the problem (Yang *et al.* 2020). The goal of learning from heterogeneous data is to leverage the information from multiple domains to improve the performance of machine learning models.

Multi-view learning is one of the most common methods to formulate data heterogeneity. The views are complementary to each other. The co-Training is proposed as one of the earliest methods for multi-view problems (Blum and Mitchell 1998). Canonical correlation analysis (CCA) (Hotelling 1992) and kernel canonical correlation analysis (KCCA) (Akaho 2006) are the typical works. They both mutually maximizing the correlations between the projections of two original views in the shared latent subspace. Then the SVM-2K (Farquhar *et al.* 2006) combines the KCCA with SVM and is solved as an optimization problem. The multi-view clustering (Bickel and Scheffer 2004; Yang and Wang 2018; Liu *et al.* 2013) and multi-view regression (Kakade and Foster 2007) further utilize Fisher's discriminant analysis to explore the latent subspace generated from the multi-view data.

In multi-task learning, it aims to improve the model performance in every single task by utilizing a small amount of labeled data from the related tasks. (Evgeniou

and Pontil 2007; Zhang *et al.* 2010) learn a common feature representation from all the tasks, which denotes the shared knowledge among the tasks. (Zhang *et al.* 2019c) proposes a clustering-based multi-task framework in heterogeneous data situations. Recently, various deep learning models are proposed to boost the performance of multi-task learning. (Liu *et al.* 2019a) proposes a task-specific attention model. (Jaderberg *et al.* 2016) proposes to learn an unsupervised auxiliary task in conjunction with the main task. (Xu *et al.* 2018) introduces a method to weigh the loss of auxiliary tasks relative to the main task loss. The theoretical survey study of deep learning-based multi-task learning is recommended (Crawshaw 2020).

Besides the above-mentioned methods, which mostly focused on single heterogeneity, the dual heterogeneity has been studied by researchers in recent years Yang *et al.* (2020). (He and Lawrence 2011) proposes a method to handle the duel heterogeneity for the combination of task heterogeneity and view heterogeneity. (Zhang and Huan 2012) learns a linear mapping for each view in each task, and (Yang and Gao 2014) adapts the multi-view for cross-domain classification.

2.3 OFF-LINE METHODS FOR MISSING VALUE IMPUTATION

Matrix Factorization (MF) is a frequently used data mining method for a variety of situations. It's been frequently used and customized for dimensionality reduction, data clustering, missing value imputation, and among other things. The main goal of the MF is to generate a set of low-rank matrices that can approximate the original data in terms of observed knowledge, similar to well-known methods like principal component analysis (PCA) (Jolliffe 2002) and singular value decomposition (SVD) (Golub and Van Loan 2012). To be more specific, MF assumes that the partially observed information, i.e., the matrix $\mathbf{M}$, can be estimated by the product of two low-rank matrices, i.e., the matrix $\mathbf{U}$ and $\mathbf{V}$, whose product $\mathbf{U}\mathbf{V}^\top$ represents the

minimum Euclidean distance with respect to the observed information in matrix $\mathbf{M}$. The matrices $\mathbf{U}$ and $\mathbf{V}$ are used as factorization factors, while the missing value in the matrix $\mathbf{M}$ is estimated in the $\mathbf{U}\mathbf{V}^\top$ produce.

In practice, the non-negative property often exists in many real-world applications, especially for the medical and healthcare domain like medical imaging analysis(Carr *et al.* 1997), gene expression (Gao and Church 2005), healthcare fraud detection (Zhu *et al.* 2011), and medical recommender system (Zhang *et al.* 2017). These applications naturally require the non-negative property for each entry in the data, however, such non-negative constraint is not satisfied in the MF. To overcome this issue, (Lee and Seung 2001) proposes the non-negative matrix factorization (NMF), which has incorporated the non-negative constraint into the MF framework. NMF produces two non-negative low-rank matrices (also known as dual-factors), whose multiplication has the minimum Euclidean distance (defined as the square root of the sum of the absolute squares of the difference between two matrices) regarding the input data. Each of the non-negative low-rank matrices is usually considered as the clustering result for the row-wise and column-wise knowledge of the original data, which reveals the user's emotion and preference in personalized doctor system (Zhang *et al.* 2017). The work (Wang and Zhang 2013) comprehensively reviews the existing NMF methods used in various applications. Meanwhile, a collection of the medical and healthcare data is commonly presented as a patient-by-medical measurement item' matrix, in which the missing entries commonly exist. Several NMF extension methods handle the missing value issue from various perspectives. (Xu *et al.* 2012) contributes to recovering the missing data from the partially observed information by taking advantage of MF. Graph Regularized Non-negative Matrix Factorization (GNMF) (Cai *et al.* 2011) incorporates the samples' pairwise similarity by introducing the graph regularizer into the traditional NMF to explore insight

into the intrinsic geometric structure of the data, which reduce the side effects of the unknown entries. A convex and semi-NMF (Ding *et al.* 2010) method expanded the application domain by relaxing the non-negative constraint, and (Wang *et al.* 2015a) incorporates the guidance constraints to align with existing medical knowledge. However, when applying the double orthogonality in dual-factor matrix factorization, it is very restrictive and gives a rather poor matrix low-rank approximation. Thus (Ding *et al.* 2006) proposed the tri-factor factorization method subject to the double orthogonal constraints on both factorization factors, which allows the different cluster number of row and column clustering. (Gu *et al.* 2011) proposed to solve the common scale transfer problem by leveraging normalized cut-like constraints. Recent work incorporates the deep learning model with matrix factorization for the missing value imputation task (Liu *et al.* 2019b).

Besides the MF-based collaborative filtering (CF), content-based filtering (CBF) algorithms relieve the recommender problem by using auxiliary modalities/information such as product descriptions, photos, and user reviews and explore the relationships between different data modalities, which is known as multi-modality. The data in various domains, such as computer vision, clinical, and recommender systems, is naturally collected with multiple modalities. Because multiple modalities of a subject give complementary information, many clinical applications, such as tumor detection (Xu *et al.* 2016; Zhang and Metaxas 2016) and brain illness diagnosis (An *et al.* 2016; Li *et al.* 2014; Wang *et al.* 2016), require high-quality multi-modality data in order to get appropriate diagnostic findings. In addition to anatomical features provided by other popular modalities like magnetic resonance imaging (MRI), for example, the positron emission tomography (PET) modality is frequently utilized to show metabolic information. The missing value imputation for image processing is formulated as a conditional image generation task (Cai *et al.* 2018), where the deep learning

models have achieved success (Dong *et al.* 2017; Mathieu *et al.* 2015). In terms of computer vision tasks, generative adversarial networks (GANs) have been suggested and have shown to be effective for missing value imputation tasks. A generator network and a discriminator network make up the GAN framework. The discriminator is used to differentiate imputed values in pictures from the data set, while the generator transfers latent representations to pictures. The GAN model can be readily adapted to conditional GANs by including conditional information in the latent representations under various application scenarios. The encoder-decoder architecture is used as the generator network in most recent studies on conditional picture creation challenges, which encodes conditional information to latent representations.

Besides the GAN-based model, graph spectral analysis is also widely used when data naturally arise in the graph structure in real-world problems including social media (Dong *et al.* 2019), recommender system (Ying *et al.* 2018; Wu *et al.* 2018b), drug-target and molecular analysis (Torng and Altman 2019; You *et al.* 2018), and more. Compared with the grid structure data, e.g., image, the relationship information among entities has been encoded in the graph model and provides us insight into knowledge underlying the data. The non-Euclidean nature of the graph-structured data requires an analysis mechanism to first quantify the complex pattern of graph structure data in order to make any further exploration. Graph spectral perspective has been proposed in the past decades as an auxiliary method to conduct relationship analysis for the graph structural data, and thus, graph convolution processing and graph filtering processing have attracted much attention (Zhang *et al.* 2019a). The basic idea of graph spectral is motivated by the traditional signal Fourier transform, which conducts the signal analysis by transforming the sequential signal from the time domain to the frequency domain. The analogy to this process, the non-Euclidean graph structural data is transformed from its original domain to the so-called graph

spectral domain. The graph convolution operation is further proposed based on graph spectral processing as the aggregations of node representations from the node neighborhoods. More recently, through revisiting deep learning with graph spectral theory, graph knowledge is decoded by the deep model as a graph convolution network by introducing graph filters. In particular, the authors of (Hammond *et al.* 2011) shows that the Chebyshev polynomial approximation can well estimate the graph filters, and the authors of (Defferrard *et al.* 2016) introduces such graph filters into convolution neural networks for handling the graph-structured data. Furthermore, the authors of (Kipf and Welling 2016) simplifies the graph convolution process and it is widely applied since its inception (Zhang *et al.* 2018).

2.4 ON-LINE METHODS FOR MISSING VALUE IMPUTATION

E-commerce enterprises frequently use online missing value imputation methods, in which the consumers' shopping preferences are considered as missing data. Most machine learning models in recommender systems use both long-term and short-term data to determine the users' preferences. Customers who shop online are sometimes greeted with thousands of options to examine but impossible to make a final decision immediately. In recent years, there has been a surge in interest in industrial applications of recommender systems, since they help to assess the missing value of a customer's interest, decrease consumer diversions, and disclose a reasonable amount of products that are most relevant to the customers' purchasing goal. In these ranking systems, which take the form of search or recommendation systems, products are rated in descending order of relevance to the client. (Nigam *et al.* 2019; Wu *et al.* 2018a; Zhao *et al.* 2020b; Li *et al.* 2018; Hu *et al.* 2018; Yan *et al.* 2018). The related works are summarized from literature and categorized into two aspects: (1) Session-based methods, and (2) Multi-armed Bandit (MAB) based methods.

The session-based method aims at exploring the customers' online activities and behavior within a short period of time, also known as a session. The within-session ranking job uses the temporal nature of the user's browsing activity from the same session to estimate what action the user will do next inside the current shopping session (Li *et al.* 2018; Yu *et al.* 2016). Deep learning models have been widely adopted in numerous companies and applications as a result of significant advancements (e.g., solving cold-start, batch normalization, and dropout to avoid overfitting) (Zhang *et al.* 2019b). Recurrent neural networks (RNNs) are introduced for this within-session rating job in (Hidasi *et al.* 2015) and gained substantial momentum due to the improved prediction performance for the next-item recommendation. Various upgrades have been proposed expressly for forecasting short-term user behavior during the same shopping session, and this has been an active study topic in recent years.

Given that long-term memory, models are insufficient to address drift in user interests. (Liu *et al.* 2018) proposes a short-term attention priority model that uses a short-term memory model based on recent clicks to capture users' general (long-term) interest as well as their within-session interest. Simultaneously, (Li *et al.* 2018) investigates a behavior-intensive neural network for the personalized next-item recommendation that took into account the both users' long-term preferences and within-session purchase intent. As RNNs have demonstrated and emerged as the most powerful technique for modeling sequential data for this task, (Loyola *et al.* 2017) proposes an encoder-decoder neural architecture with an attention mechanism added to capture user session intents and intersession dependencies based on machine translation. In addition to sequential models, (Qiu *et al.* 2019) utilizes graph neural networks to predict user preference in session by generating a session graph and then modeling a weighted attention layer. Authors in (Guo *et al.* 2019) proposes a matrix factorization-based attention model to address large-volume and high-velocity session

streaming data, and (Liu *et al.* 2019b) handles the missing value issue for the matrix factorization to address the uncertainty that arises in a user's within-session behavior. The majority of the earlier research cited above do not seek for interpretability of its findings. The most recent work (Bai *et al.* 2018, on the other hand, uses item data from the product catalog to create a simple algorithm that learns interpretable user profiles to aid in within-session customization, while it also provides an attribute-aware neural attentive model for the next shopping basket recommendation, but due to its complexity, it does not appear to be easily adaptable for the real-time scenario.

Requiring a responsive and scalable ranking system that can adapt to the dynamic nature of shifting user preferences has led to increasingly wider industry adoption of multi-armed bandit (MAB) in modern-day ranking systems. Even though the instantly match relevant items for users still remains a challenge, especially in the cold start setting with constant new users or items (i.e, cold-start), the theoretical foundation and analysis of MABs have been well-studied with popular approaches include ϵ-greedy (Sutton and Barto 2018), Upper Confidence Bounds (Auer *et al.* 2002), Thompson sampling (Chapelle and Li 2011), EXP3 (Auer *et al.* 2003), and others (Sutton and Barto 2018) to fit the real-world scenario. In the(Zhao *et al.* 2020b) setting, the goal is to maximize user satisfaction (i.e., exploitation), while quickly learning (i.e., exploration) users' preferences by exploring unseen content. (Hu *et al.* 2018) proposes to use reinforcement learning to learn an optimal ranking policy to maximize the expected accumulative rewards in a search session. (Yan *et al.* 2018) builds a scalable deep online ranking system (DORS) by designing the MABs as the last pass to dynamically re-rank items based on user real-time feedback and shows significant improvement in both users satisfaction and platform revenue. The bandit recommender system is the vision of the multi-arm bandit (Sutton and Barto 2018) that using bandit methods to recommend next items to users by considering

all the candidate arms of the bandit. Furthermore, authors from (Sanz-Cruzado *et al.* 2019) proposes a multi-armed nearest-neighbor bandit to achieve collaborative filtering for the interactive recommendation, by modeling users as arms and exploring the users' neighborhood. (Wang *et al.* 2019) proposeds an interactive collaborative topic regression model that infers the clusters of arms via topic models (Blei *et al.* 2003) and then utilizes dependent arms for the recommendation. In this methods, it is common to address the problem by treating each arm in the bandit to represent a single item (Zhao *et al.* 2020b), product category (Yan *et al.* 2018) or a context (Li *et al.* 2016; Hu *et al.* 2018; Li and Kar 2017).

Chapter 3

MISSING VALUE IMPUTATION FOR RECOMMENDER SYSTEMS

In this chapter, the works for missing value imputation in the scenario of building a recommender system are described from two perspectives: (1) offline trained strategy, and (2) online trained strategy. The foal of offline trained strategy is defined as the problem of collective matrix completion. The graph spectral analysis is used to incorporate the user-neighborhood similarity as well as the intrinsic cross-category information. Meanwhile, the goal of the online trained strategy is to estimate customers' purchase preferences by exploring his/her purchase intent and shopping preference in the attribute-level, such as color, size, shape, and material, and quickly learn a buyer's fine-grained preferences based on their most recent activity (e.g., browsing, click, add-to-cart, check out) in a short time period.

3.1 NOTATIONS

The boldface uppercase letters denote the matrices (e.g., $\mathbf{X}$, $\mathbf{M}$). The boldface lowercase letters denote the vectors (e.g., $\mathbf{u}$, $\mathbf{v}$). Let $\mathbf{X}_{ij}$ denote the entry in the i^{th} row and j^{th} column of matrix $\mathbf{X}$, $\mathbf{X}^\top$ denotes its transpose, and $\mathbf{u}_i$ denotes i^{th} entry of vector $\mathbf{u}$. The uppercase Greek letters are used to represent scalars. All vectors are column vectors unless otherwise specified. For the matrix dimension, there are T incomplete matrices $\{\mathbf{X}_t\}_{t=1}^T \subset R^{m\times n_t}$ and two set of sub-matrices $\{\mathbf{U}_t\}_{t=1}^T \subset R^{m\times c_t}$ and $\{\mathbf{V}_t\}_{t=1}^T \subset R^{n_t\times c_t}$. The products $\{\mathbf{U}_t\mathbf{V}_t^\top\}_{t=1}^T$ are treated as the estimation $\{\tilde{\mathbf{X}}\}_{t=1}^T$ of the corresponding matrix respectively.

3.2 MISSING VALUE IMPUTATION WITH OFFLINE STRATEGY

3.2.1 COLLECTIVE MATRIX COMPLETION

The goal of collective matrix completion (CMC) is to collectively complete multiple incomplete matrices by leveraging the cross-matrix information. Each matrix, also known as one view, corresponds to one type of measurement, while multiple views contain complementary information from various sources. CMC benefits from the correlation among multi-view data and aims to predict their missing entries with high accuracy. The CMC mechanism can be expressed as:

$$\{\tilde{\mathbf{X}}_t\}_{t=1}^T = f(\{\mathbf{X}_t\}_{t=1}^T, \Lambda) \tag{3.1}$$

where $\{\mathbf{X}_t\}_{t=1}^T \subset R^{m \times n_t}$ denote T incomplete views and Λ denotes the auxiliary information including view-specific knowledge and cross-view knowledge. $\{\tilde{\mathbf{X}}_t\}_{t=1}^T$ is treated as the results that contains the original observation data and prediction of missing value simultaneous. There are two kind of approaches extensively applied in the CMC studies when formulating the cross-view knowledge Λ:

(1) Matrix Factorization based Low-rank Latent Structure: For the matrices $\{\mathbf{X}_t\}_{t=1}^T$, two set of sub-matrices $\{\mathbf{U}_t\}_{t=1}^T$ and $\{\mathbf{V}_t\}_{t=1}^T$ are generated to provide a good approximation to the observed values as:

$$\mathbf{X}_t \approx \mathbf{U}_t \mathbf{V}_t^\top \tag{3.2}$$

for different views, when I assume the number of data is same, an additional cross-view factor V^* or V_t^* is proposed to emphasize the cross-view knowledge as following two way:

$$\sum_{t=1}^{T} \|\mathbf{X}_t - \mathbf{U}_t \mathbf{V}^{*\top}\|_F^2 \tag{3.3}$$

where the $\mathbf{V^*}^\top$ denotes the shared consensus directly on the view features, or,

$$\sum_{t=1}^{T} \|\mathbf{X}_t - \mathbf{U}_t \mathbf{V_t}^\top\|_F^2 + \|\mathbf{V_t}^\top - \mathbf{V^*}^\top\|_F^2 \tag{3.4}$$

where $\mathbf{V^*}^\top$ is proposed to be shared among all view indirectly. The single measurement matrix $\mathbf{V}^*$ can be learned from the data when optimizing the above function (3.3) or (3.4) by minimizing the overall Euclidean distance to each view (Liu *et al.* 2013).

(ii) Constraint-based Metric: Another way to incorporate the cross-view knowledge is by enforcing the user-user / item-item similarity in the formulation. The samples' pairwise similarity can be enforced by minimizing:

$$\sum_{t=1}^{T} \|\mathbf{X}_t - \mathbf{U}_t \mathbf{V}_t^\top\|_F^2 + Tr(\mathbf{U}_t^\top \mathbf{L} \mathbf{U}_t) \tag{3.5}$$

where graph regularizer $Tr()$ measures the smoothness of low dimensional representation as:

$$\begin{aligned} Tr(\mathbf{U}_t^\top \mathbf{L} \mathbf{U}_t) &= Tr(\mathbf{U}_t^\top \mathbf{D} \mathbf{U}_t) - Tr(\mathbf{U}_t^\top \mathbf{A} \mathbf{U}_t) \\ &= \sum_{j=1}^{N} \mathbf{u}_j^\top \mathbf{u}_j \mathbf{D}_{jj} - \sum_{j,l=1}^{N} \mathbf{u}_j^\top \mathbf{u}_l \mathbf{A}_{jl} \\ &= \frac{1}{2} \sum_{j,l}^{N} \|\mathbf{u}_j - \mathbf{u}_l\|^2 \mathbf{A}_{jl} \end{aligned} \tag{3.6}$$

where L denotes the weighted graph Laplacian matrix, which generated form the concatenation of each views. In specific, for N user, $\mathbf{L} = \mathbf{D} - \mathbf{A}$ that $\mathbf{D}$ denotes the degree matrix of $\mathbf{A}$ as a diagonal matrix as $\mathbf{D}(i,i) = \sum_{j=1}^{N} \mathbf{A}(i,j)$, and $\mathbf{A}(i,j)$ denotes the similarity between user i and user j taking overall information among all views.

3.2.2 CROSS-VIEW KNOWLEDGE MODELING VIA GRAPH SPECTRAL ANALYSIS

To formulate the cross-view knowledge in the graph spectral domain, I first define the problem and given the preliminary about graph spectral analysis. Then I present the proposed model mathematically.

PROBLEM DEFINITION: QUANTIFYING CROSS-MATRIX INFORMATION

Input: *(1) Incomplete matrices* $\{\mathbf{X}_t\}_{t=1}^T$.

Output: *(1) Matrix completion results* $\{\tilde{\mathbf{X}}_t\}_{t=1}^T$. *(2) Cross-view knowledge* $\boldsymbol{\mathcal{W}}_k$.

There are two main challenges arise as:

(**C1**) how to capture the cross-matrix information when data implicate the non-Euclidean structure, e.g., graph-structured data.

(**C2**) how to quantify the matrices' interactive impacts. Either positive or negative impacts exist between the matrices, e.g., how much knowledge does the view 2 contribute to predicting the missing entries in view, or vice versa.

Preliminary: Given a graph $\mathcal{G}$ with m nodes, presented as $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{A})$, with the adjacency matrix $\mathbf{A} \in R^{m \times m}$, vertex set $\mathcal{V}$ and edge set $\mathcal{E}$. The normalized graph Laplaican matrix is defined as :

$$\boldsymbol{\Delta} = \mathbf{I} - \mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}} \tag{3.7}$$

with the diagonal degree matrix $\mathbf{D}_{ii} = \sum_j^m \mathbf{A}_{ij}$ and identity matrix $\mathbf{I} \in R^{m \times m}$. As $\boldsymbol{\Delta}$ is positive semidefinite, it has a complete set of eigenvalues $\boldsymbol{\Lambda} = (\lambda_1, \lambda_2, \ldots, \lambda_m)$ and eigenvectors $\boldsymbol{\Phi} = (\phi_1, \phi_2, \ldots, \phi_m)$ for its eigendecomposition $\boldsymbol{\Delta} = \boldsymbol{\Phi}\boldsymbol{\Lambda}\boldsymbol{\Phi}^\top$. In graph spectral theory (Hammond *et al.* 2011), eigenvalues $\{\lambda_i\}_{i=1}^m$ are identified as graph spectral frequencies and eigenvectors $\{\phi_i\}_{i=1}^m$ are identified as graph Fourier basis.

Based on the eigenbasis, for a graph signal $\mathbf{x} \in R^m$, the graph Fourier transform is defined as:

$$\tilde{\mathbf{x}} = \mathbf{\Phi}^\top \mathbf{x} = \sum_{i=1}^{N} \mathbf{x}_i \phi_i \tag{3.8}$$

and its inverse transform $\mathbf{x} = \mathbf{\Phi}\tilde{\mathbf{x}}$. The graph convolutional operation $*_\mathcal{G}$ for the graph signals $\mathbf{x}$ and $\mathbf{y}$ is then defined on the graph spectral domain as:

$$\mathbf{x} *_\mathcal{G} \mathbf{y} = \mathbf{\Phi}(\mathbf{\Phi}^\top \mathbf{x}) \odot (\mathbf{\Phi}^\top \mathbf{y}) = \mathbf{\Phi} g_\theta(\mathbf{\Lambda})\tilde{\mathbf{x}} \tag{3.9}$$

where $\odot$ denotes element wise product. $g_\theta(\mathbf{\Lambda})$ is recognized as θ-parameterized graph filter. More recently, the graph filter $g_\theta(\mathbf{\Lambda})$ is re-modeled to decrease its complexity by being expanded by the Chebyshev polynomial as:

$$g_\theta(\mathbf{\Lambda}) = \sum_{k=0}^{K} \theta_k T_k(\tilde{\mathbf{\Delta}}) = \sum_{k=0}^{K} \theta_k \mathbf{\Phi} T_k(\tilde{\mathbf{\Lambda}}) \mathbf{\Phi}^\top \tag{3.10}$$

where the modified graph Laplacian $\tilde{\mathbf{\Delta}} = \frac{2\mathbf{\Delta}}{\lambda_{max}} - \mathbf{I}_m$ and its eigenvalues $\tilde{\mathbf{\Lambda}}$ fall into the range $[-1, 1]$. $T_k(\cdot)$ represents the k-th order Chebyshev polynomial abiding by the recursive manner $T_k(\lambda) = 2\lambda T_{k-1}(\lambda) - T_{k-2}(\lambda)$ with $T_0(\lambda) = 1$ and $T_1(\lambda) = \lambda$.

Note that this expression only contains the K-th localized neighborhood knowledge of the central node, i.e. only the nodes within K steps away from the central node, since the highest order is K for K-th polynomial when taken Laplacian matrix as input.

3.2.3 PROBLEM FORMULATION

For each view separately, each matrix is expanded by the Chebyshev polynomial thus each matrix is reconstructed as the weighted combination of its graph structure knowledge from K^* level. The expectation is that K^*-th order graph filters are capable enough to preserve the observed knowledge as much as possible. The completion results $\{\tilde{\mathbf{X}}_t\}_{t=1}^T$ are expected to be consistent with the observed entries in $\{\mathbf{X}_t\}_{t=1}^T$ as

the reconstruction error between $\{\tilde{\mathbf{X}}_t\}_{t=1}^T$ and $\{\mathbf{X}_t\}_{t=1}^T$ is minimum, which is defined as solving the problem:

$$\begin{aligned} &\min_{\mathbf{U}_t,\mathbf{V}_t} \sum_{t=1}^{T} \left\|\mathbf{X}_t - \mathbf{U}_t\mathbf{V}_t^\top\right\|_{F,\mathbf{\Omega}_t}^2 \\ &s.t.\, \{\mathbf{U}_t\}_{t=1}^T \subset R_+^{m\times c_t},\, \{\mathbf{V}_t\}_{t=1}^T \subset R_+^{n_t\times c_t} \end{aligned} \tag{3.11}$$

where $\mathbf{U}_t \in R^{m\times c_t}$ and $\mathbf{V}_t \in R^{n_t\times c_t}$. R_+ denotes the non-negative real numbers and the completion results are $\{\tilde{\mathbf{X}}_t = \mathbf{U}_t\mathbf{V}_t^\top\}_{t=1}^T$. The index matrix $\mathbf{\Omega}_t \in R^{m\times n_t}$ contains $\mathbf{\Omega}_{t[i,j]} = 1$ if $\mathbf{X}_{t[i,j]}$ is observed, otherwise 0. To simplify the expression, $\|\mathbf{X}\|_{F,\mathbf{\Omega}}^2$ is equivalent to the expression of $\|\mathbf{X}\odot\mathbf{\Omega}\|_F^2$, in which $\odot$ denotes the Hadamard product. In Eq. (3.11), the factor $\mathbf{U}_t$ is polynomial expanded by Eq. (3.10) as:

$$\begin{aligned} &\min_{\mathbf{U}_t,\mathbf{V}_t,\theta_{k,t}} \sum_{t=1}^{T} ||\mathbf{X}_t - (\sum_{k=0}^{K} \theta_{k,t} T_k(\tilde{\mathbf{\Delta}}_{r,t})\mathbf{U}_t)\mathbf{V}_t^\top||_{F,\mathbf{\Omega}_t}^2 \\ \Leftrightarrow &\min_{\mathbf{U}_t,\mathbf{V}_t,\theta_{k,t}} \sum_{t=1}^{T} ||\mathbf{X}_t - (\theta_{0,t}T_0(\tilde{\mathbf{\Delta}}_{r,t})\mathbf{U}_t + \theta_{1,t}T_1(\tilde{\mathbf{\Delta}}_{r,t})\mathbf{U}_t + \\ &\qquad\qquad \cdots + \theta_{K,t}T_K(\tilde{\mathbf{\Delta}}_{r,t})\mathbf{U}_t)\mathbf{V}_t^\top||_{F,\mathbf{\Omega}_t}^2 \end{aligned} \tag{3.12}$$

constrained by $\{\mathbf{U}_t\}_{t=1}^T \subset R_+^{m\times c_t}$ and $\{\mathbf{V}_t\}_{t=1}^T \subset R_+^{n_t\times c_t}$. For the t-th view, parameter $\theta_{k,t}$ weights the k-th order graph filter $T_k(\tilde{\mathbf{\Delta}}_{r,t})$. $\tilde{\mathbf{\Delta}}_{r,t}$ denotes the normalized row-wise Laplacian matrix. To be more specific, the factor $\mathbf{U}_t$ is described as the weighted combination of K-th order graph structure knowledge, which is purposeful to reinforced the estimation of the matrix $\mathbf{U}_t$ by the localized graph knowledge from K-th level neighboring information.

The Matrix-Stitch Unit $\boldsymbol{\mathcal{W}}_k$ is proposed to formulate the cross-view knowledge. For illustration purpose, only two views are considered ($T = 2$), while in practice, the Matrix-Stitch Unit is feasible to the arbitrary number of views ($T \geq 2$), which has been demonstrated in the experiments. Based on the Eq. (3.12) when ($t = 1, 2$, $k = 1, 2, \ldots, K$), the factors $\mathbf{U}_t$ are expanded by $T_k(\tilde{\mathbf{\Delta}}_{r,t})$ and $\theta_{k,t}$, where $\theta_{k,t}$

is the learnable weighted parameter reflecting how does the k-th localized graph knowledge $T_k(\tilde{\mathbf{\Delta}}_{r,t})$ impact in each view separately. The Matrix-Stitch Unit considers the impacts from both view itself and all the other views. The unit $\boldsymbol{\mathcal{W}}_k$ is designed as a weight matrix between the parameters $\{\theta_{k,t}\}_{t=1}^{T=2}$ for each level of the graph localized knowledge as:

$$\tilde{\mathbf{\Theta}}_k = \begin{bmatrix} \tilde{\theta}_{k,1} \\ \\ \tilde{\theta}_{k,2} \end{bmatrix} = \begin{bmatrix} w_{11} & w_{12} \\ \\ w_{21} & w_{22} \end{bmatrix}_k \begin{bmatrix} \theta_{k,1} \\ \\ \theta_{k,2} \end{bmatrix} = \boldsymbol{\mathcal{W}}_k \mathbf{\Theta}_k \tag{3.13}$$

where $\tilde{\mathbf{\Theta}}_k, \mathbf{\Theta}_k \in R^{T\times 1}$ and the matrix-stitch unit $\boldsymbol{\mathcal{W}}_k \in R^{T\times T}$. There are total K^* units $\boldsymbol{\mathcal{W}}_k$ when adopting the K^* order graph filter. Incorporating the Eq. (3.13) into Eq. (3.12), the objective function of C^4 is written as:

$$\begin{aligned} &\min_{\mathbf{U_t},\mathbf{V}_t,\mathbf{\Theta}_k,\boldsymbol{\mathcal{W}}_k} \sum_{t=1}^{T} ||\mathbf{X}_t - (\sum_{k=0}^{K} \tilde{\mathbf{\Theta}}_{k_{[t,1]}} T_k(\tilde{\mathbf{\Delta}}_{r,t})\mathbf{U}_t)\mathbf{V}_t^\top||^2_{F,\mathbf{\Omega}_t} \\ \Leftrightarrow &\min_{\mathbf{U}_t,\mathbf{V}_t,\mathbf{\Theta}_k,\boldsymbol{\mathcal{W}}_k} \sum_{t=1}^{T} ||\mathbf{X}_t - (\sum_{k=0}^{K} \boldsymbol{\mathcal{W}}_{k_{[t,:]}} \mathbf{\Theta}_k T_k(\tilde{\mathbf{\Delta}}_{r,t})\mathbf{U}_t)\mathbf{V}_t^\top||^2_{F,\mathbf{\Omega}_t} \end{aligned} \tag{3.14}$$

constrained by $\{\mathbf{U}_t\}_{t=1}^{T} \subset R_+^{m\times c_t}$ and $\{\mathbf{V}_t\}_{t=1}^{T} \subset R_+^{n_t\times c_t}$. Due to the space limitation, the C^4 updating procedure is summarized in Algorithm 1.

3.2.4 ADAPTED DATA-DRIVEN FILTER ORDER

In this subsection, the proposed techniques for selecting the filter order $K*$ are introduced. The problem definition is as follows:

Problem: *Selecting Filter Order K^**

Input: *Incomplete matrices $\{\mathbf{X}_t\}_{t=1}^{T}$ with missing entries.*

Output: *Adapted graph filter order K^*.*

The output K^* denotes the adapted graph filter order derived from the cross-matrix information observed in the input matrices $\{\mathbf{X}_t\}_{t=1}^{T}$. The order K^* plays a decisive role in adopting the graph structure knowledge, i.e., only the nodes within maximum

Algorithm 1 - C^4 Updating Procedure

1: **Input**:

$\{\mathbf{X}_t\}_{t=1}^T$: multiple matrices with missing entries.

$\{\tilde{\mathbf{\Delta}}_{r,t}\}_{t=1}^T$: normalized row-wise Laplacian matrices.

K^*: graph filter order.

2: **Initialization**:

Initialize $\{\mathbf{U}_t\}_{t=1}^T$, $\{\mathbf{V}_t\}_{t=1}^T$, $\mathbf{\Theta}_k$ and $\boldsymbol{\mathcal{W}}_k$ randomly.

3: **Repeat**:

Perform Stochastic Gradient Descent (SGD) algorithm to update $\{\mathbf{U}_t\}_{t=1}^T$, $\{\mathbf{V}_t\}_{t=1}^T$, $\mathbf{\Theta}_k$ and $\boldsymbol{\mathcal{W}}_k$ one at a time.

4: **Until**: Eq. (3.14) converges.

5: **Output**: $\{\tilde{\mathbf{X}}_t\}_{t=1}^T$: completion results

K^* steps away from the central node are taken into consideration. The influence of filter order comes from two perspectives:

- The low-order graph filters capture the nearest neighborhood knowledge surrounding each node, which shows the similar patterns existing in each view.
- As the order increases, the less similarity has been preserved by the far-away neighborhoods. Even worse, I found that the model would be impaired when incorporating the far-away neighborhoods into cross-matrix information.

In the model, the order K^* is proposed to be settled with respect to the minimum information loss considering from the graph spectral domain. I settle the order K^* for each data set which brings in the minimum effect when removing the graph filters higher than K^*. The superiority of this strategy is shown in the experiments.

Table 3.1: Multi-View Amazon Review Data Sets.

ID	Views	|User|	|Item1|	|Item2|	|Item3|	|Rating|
1	**Electronics** & **Video Games**	6352	12836	8059	-	39574
2	**Patio** & **Tools**	3778	4077	7813	-	27712
3	**Beauty Product** & **Clothing**	1318	3406	7261	-	12691
4	**Art** & **Musical Instruments**	1412	602	988	-	8520
5	**Electronics** & **Kindle Store**	1050	3956	1614	-	7431
6	**Beauty Product** & **Jewelry**	266	1458	730	-	3870
7	**Kindle Store** & **Software**	190	637	627	-	2885
8	**Electronics** & **Video Games** & **Software**	1724	4383	2487	3845	12741
9	**Patio** & **Tools** & **Pet Supplies**	652	812	1564	715	8125
10	**Beauty Product** & **Clothing** & **Jewelry**	571	1845	2377	492	4298

3.2.5 EVALUATION

Data Sets: Table. 3.1 shows ten data sets collected from Amazon datum (McAuley and Leskovec 2013). Seven of them contain two views (ID 1-7) and three of them contain three views (ID 8-10). Taking data set (ID 1) as an example, view 1 '*Electronics*' contains 6352 users and their 39574 ratings for 12836 products, and view 2 '*Video Games*' contains 27712 ratings for 12,836 products from the same users' group. In each view, 30% ratings of each item are removed and serves as the ground-truth for the complete results.

Baselines: The proposed model is compared with 8 state-of-the-art methods, including GROUSE (Balzano and Wright 2013), IALM (Lin *et al.* 2010), LMaFit (Wen *et al.* 2012), MC-NMF (Xu *et al.* 2012), OR1MP (Wang *et al.* 2015b), RMAMR (Ye

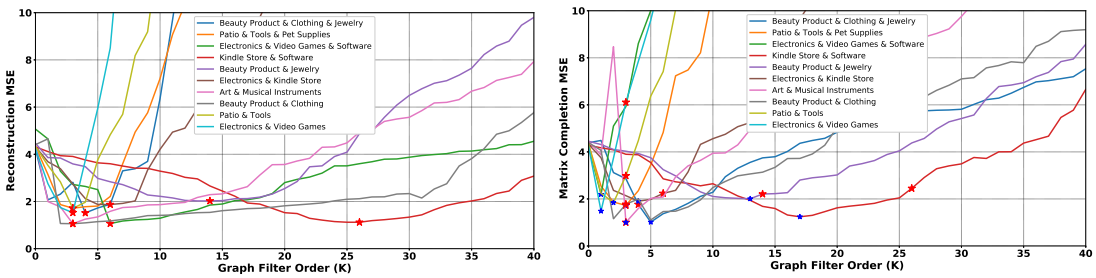


Figure 3.1: (Left) Red stars (K^*) leads to minimum reconstruction MSE. (Right) Red stars are the same stars in Left. Blue stars denote ideal filter order identified through the offline searching.

et al. 2015), ScGrassMC (Ngo and Saad 2012), and multiNMF (Liu *et al.* 2013). Parameters are initialized as suggested in (Sobral and Zahzah 2016).

Table 3.2: Matrix Completion MSE W.R.T. Ground-truth and Missing Entries.

Dataset ID	C^4	GROUSE	IALM	LMaFit	MC-NMF	OR1MP	RMAMR	ScGrassMC	multiNMF
1	1.206±0.031	**1.181±0.162**	1.689±0.004	1.429±0.002	1.986±2.542E-6	1.590±4.590E-31	1.344±0.003	1.347±1.275E-30	3.817±1E-9
2	**1.350±0.007**	1.368±0.133	1.423±0.012	1.452±0.004	1.946±1.104E-5	1.464±1.653E-30	1.416±0.007	1.943±1.275E-31	2.298±1E-9
3	**1.027±0.004**	1.282±0.087	1.446±0.005	2.005±0.002	2.045±1.447E-5	1.392±2.040E-31	1.536±0.003	1.733±1.275E-30	4.301±1E-9
4	**1.016±0.002**	1.320±0.087	1.478±0.008	1.664±0.004	2.059±3.419E-6	1.796±4.590E-31	1.506±0.005	1.814±2.648E-31	2.961±1E-9
5	1.341±0.092	**1.268±0.093**	1.807±0.003	1.894±0.002	2.008±4.946E-6	1.475±1.275E-30	1.387±0.006	1.641±8.161E-31	2.888±1E-9
6	**1.235±0.031**	1.328±0.134	1.494±0.007	2.082±0.008	2.057±8.806E-6	1.808±2.684E-31	1.498±0.001	2.047±3.264E-30	4.253±1E-9
7	**1.174±0.056**	1.217±0.015	1.759±0.003	2.028±0.008	1.984±3.216E-5	1.444±2.040E-31	1.517±0.006	1.903±4.590E-31	2.403±1E-9
8	**1.256±0.081**	1.335±0.153	1.812±0.015	1.896±0.002	2.009±8.37E-6	1.590±1.154E-9	1.437± 0.005	1.676± 5.478E-32	3.674±1E-9
9	**1.207±0.048**	1.310±0.089	1.504±0.020	2.080±0.001	2.058±1.27E-5	1.264±1.348E-9	1.9431±0.004	2.044±2.191E-31	2.479±1E-9
10	**1.243±0.032**	1.279±0.032	1.732±0.003	1.880±0.004	1.981±5.14E-5	1.292±1 .674E-9	1.742±0.002	1.841±4.213E-32	2.738±1E-9

Experimental Results: As shown in Table 3.2, the completion results are evaluated by the mean squared error (MSE) between the ground-truth and prediction values. The model C^4 achieves the best completion performance compared with state-of-the-art methods.

Filter Order Discussion: The best filter order can be found by iterating over

all possible values for various data sets; however, big data sets make this impossible. Based on the sample proportion of the observed data $\{X_t\}_{t=1}^{T}$, I estimate the order K^*. The larger the percentage sampled, the more precise order K^* may be approximated within the region of feasible calculation cost. Red stars in the (Left) (graph filter order vs. reconstruction MSE) of Fig. 3.1 denote K^* filter order estimations. The optimum filter order determined through offline iterative searching is indicated by the blue stars in Fig. 3.1 (Right). The estimation filter orders (K^* in red stars) are almost always close to the ideal order (blue stars).

I further check the assumption that greater K does not represent true cross-matrix information, in addition to the order estimation. The completion performance drops as the MSE value bounces back at a certain point of increasing filter order since the cross-matrix information is damaged when including nodes far away from the neighborhood, as seen in Fig. 3.1 (Right).

3.3 MISSING VALUE IMPUTATION WITH ONLINE TRAINED STRATEGY

When shopping online, the customers' real-time shopping preference is treated as missing data. It is impossible to visually present the customers' preferences, not matter by numerical or contextual measurement. However, being aware of the users' shopping intent is valuable for building an efficient recommender system for the goods of e-commerce. Customers often express and refine their purchase preferences by exploring different items in the product catalog based on varying attributes, such as color, size, shape, and material. As such, it is increasingly important for e-commerce ranking systems to quickly learn a buyer's fine-grained preferences and re-rank items based on their most recent activity within the session.

Just as a shopper might browse the aisles of a shop, online shoppers also spend time on a retailer's website searching and clicking on items before they decide what

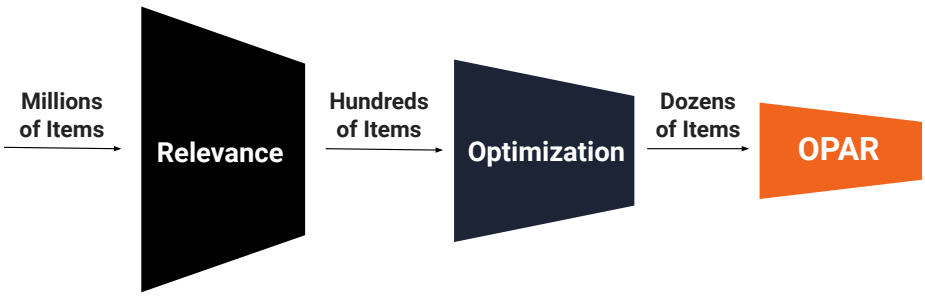


Figure 3.2: The first two components show a typical 2-stage ranker, where the first-pass narrows down the product catalog to relevant items, while the second-pass performs fine-grained re-ranking to optimize for a business metric. The proposed strategy, *OPAR*, is responsible for within-session, online personalization that can be effective on its own or as a third-pass ranker on top of a 2-stage ranking system.

they want to buy. This process is an attempt to refine their purchase intent as they learn more about the product catalog. For example, a buyer might be interested in purchasing a ring; however, they often must click on a number of different rings before they understand possible styles, shapes, colors, and materials that are available. Eventually, the buyer might decide that they have a preference for an emerald gemstone, with a circular shape, and a gold band. Shifting to looking for a necklace, the buyer must refine their preference again. Often the buyer's preference for attributes like colors and materials changes quickly over the course of one visit. An intelligent ranking system must continually serve content that stays relevant to the buyer's changing preference, a capability I refer to as within-session personalization.

The missing value imputation with online trained strategy is focused on multiple goals to balance the customers' online shopping experience:

- online retailers surface missing content that is relevant to the shopper's buying mission.
- sellers aim show content that is likely to improve a business metric (eg. conver-

sion rate, or GMV).

To balance these goals, many production ranking systems leverage a 2-stage ranking process (Figure 3.2): the first pass (commonly referred to as candidate set selection) narrows hundreds of millions of items from the product catalog down to a few hundred relevant items (Nigam *et al.* 2019; Zhao *et al.* 2020a; Huang *et al.* 2020); the second pass then re-ranks the top few hundred relevant items in a way that optimizes for specific user action (such as a click or purchase) (Wu *et al.* 2018a; Guo *et al.* 2020; Pobrotyn *et al.* 2020; Haldar *et al.* 2020). In order to maximize prediction accuracy, these systems often train on billions of historical data points that may span over the course of months or years and thus cannot react quickly enough to the buyer's changing preference within a shopping visit.

3.3.1 DEFINITIONS AND STATEMENTS

Definition 1: A session contains a set of actions taken by a buyer while interacting with an e-commerce platform to complete a purchasing mission (e.g. search, click, add-to-cart). They could convey their purchasing intent either explicitly through searches and query reformulations, or implicitly through product catalog exploration based on various criteria. The session usually terminates when the buyer makes a purchase or abandons the site after a long period of inactivity (e.g., 30 minutes). Note that though I'm focusing on product search here, the principles should apply to any ranking or recommendation problem.

Let us define a session $S = \{[Q_t, I_t, A_t]\}_{t=1}^{T}$ that is a sequence of T user actions within a session, in which T can vary across sessions. The session starts at $t = 1$ and ends at T with a purchase (or becomes inactive). At each time step, item list $I_t \in R^{M \times 1}$ contains M candidate items to be re-ranked for query Q_t, and then how

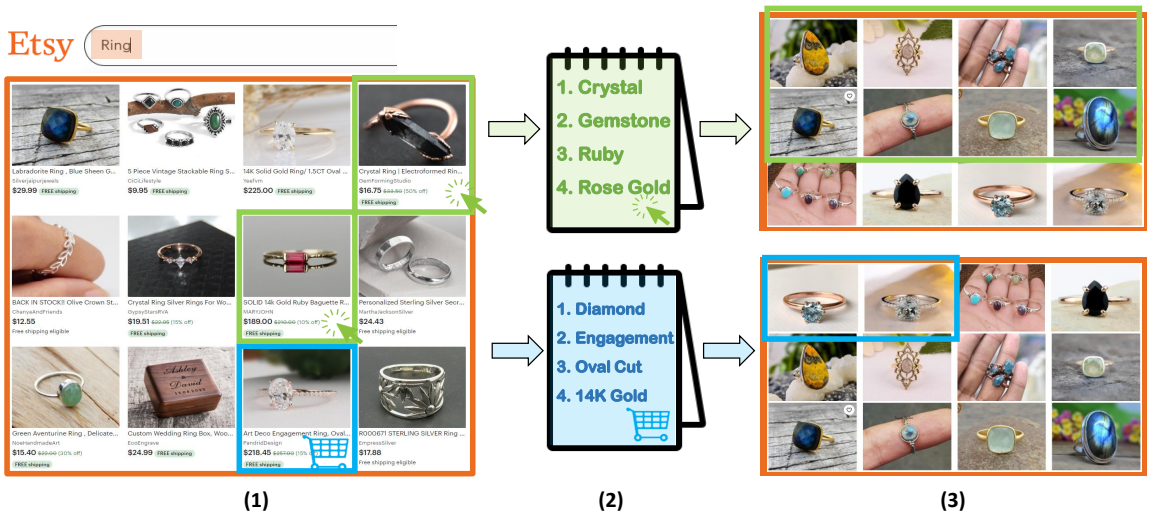


Figure 3.3: Example of attribute and action-aware re-ranking by *OPAR*. From left to right: (1) shows search results for the query "Ring". User 1 clicked on two gemstone rings (outlined in green), while User 2 adds a diamond ring to their cart (outlined in blue) (2) The attribute of the clicked items are "Crystal", "Gemstone", "Ruby" and "Rose Gold", while the add-to-cart item has the attributes "Diamond", "Engagement", "Oval-Cut" and "14k Gold" (3) On a subsequent search page, *OPAR* re-ranks items based on each user's diverging preferences.

the user engages with the list of items is represented by A_t:

$$A_t(x_i) = \begin{cases} 0, \text{ no action on } x_i \\ 1, \ x_i \text{ is purchased} \\ 2, \ x_i \text{ is added to cart} \\ 3, \ x_i \text{ is clicked} \end{cases} , \forall x_i \in I_t. \tag{3.15}$$

Definition 2: Attribute is a basic unit (e.g. size, color) that describes the product characteristics of an item. The attributes are determined by taxonomists based on the product category while the value of the attributes (e.g. large, green) are volunteered by the seller, or inferred by machine-learned classifiers. Given an item, its applicable product attributes are often configured by sellers or inferred by ML classifiers to

(a) Jewelry (b) Clothing (c) Supplies & Tools (d) Home and Living

Figure 3.4: Top Attribute-Value Pairs For Top Categories.

improve coverage rate and reduce human mislabels. These attribute-value pairs help buyers efficiently navigate through an overwhelmingly large inventory. Thus, each item x_i is represented as the composition of its attributes, with H_{x_i} denoting the total number of attributes associated with x_i : $x_i = \{atr_1, atr_2, \ldots, atr_{H_{x_i}}\}$.

Figure 3.4 shows four category-specific word clouds of attributes-value pairs exhibited in items from top categories at Etsy [1] , one of the largest e-commerce platform for handmade, vintage, and craft supplies. Some of the most common attributes are universal: size, color, and material. Others are category-specific: sleeve length, earring location, and craft type. Lastly, some attributes (e.g. holiday, occasion, recipient) describe how or when the item can be used.

Based on the definitions, the goal is constructed by two parts, i.e., (1) impute users' within-session shopping preference based on product attributes, and (2) re-rank a list of candidate items based on the user's inferred within-session preference on item attributes.

Goal 1: *Impute users' in-session attribute preferences*

Input: For session S, (1) item lists $\{I_t\}_{t=1}^{T}$ with each item $x_i = \{atr_{H_{x_i}}\}$ as composition of product attributes; and (2) session-level record of user actions on shown

[1]E-commerce platform for Handmade products at https://www.etsy.com

items, $\{A_t\}_{t=1}^{T}$.

Output User's preference $\mathbf{\Theta}$ on attributes as beta-distributed: $\mathbf{\Theta} = \{\theta_{atr_n}\}_{n=1}^{N} \sim \{Beta(\alpha_{atr_n}, \beta_{atr_n})\}_{n=1}^{N}$, where N denotes the total number of attributes encountered in session S.

For a user, I model their within-session preference on an attribute as a latent value $\theta_{atr_n} \in [0, 1]$ denoting the probability that they would like the attribute exhibited in the item. Motivated by Thompson Sampling (Agrawal and Goyal 2012), let θ_{atr_n} be beta-distributed, with $\alpha_{atr_n}, \beta_{atr_n}$ be the two parameters of the distribution. In Section 3.3.7 I show a method on estimating the parameters of attributes from historical data. From the list of shown items I_t, the user engages on a subset of items (denoted in A_t) to express their preference for item attributes according to $\mathbf{\Theta}$. Given the feedback, I propagate rewards from the user actions to the associated attributes with increments, $\delta_{A_t(x_i)}$, and update the posterior distribution of $\mathbf{\Theta}$, with rewards normalized at x_i by its cardinality (number of associated attributes on that item).

Goal 2: *Sequentially re-rank I_t based on user preference* $\mathbf{\Theta}$ to optimize in-session personalization.

Input: At time t, (1) Candidate list of items I_t, and (2) user in-session preference $\mathbf{\Theta}$.

Output: Sequentially learn $f_t : I_t \times \mathbf{\Theta} \rightarrow \tilde{I}_t$.

To achieve the above target goals, I propose the in-session multi-armed bandit model *OPAR* to re-rank the recommended item list based on users' in-session actions. *OPAR* serves as the 3-rd pass as a ranking problem in the attribute-level, which considers each item attribute as one arm and explores the user in-session behavior at the attribute-level. Arms are pulled and user actions (i.e., click, add-to-cart) are observed on items to collect rewards on the associated attributes. Details are as follows in the next section.

3.3.2 PROBLEM FORMULATION AND ALGORIGHM

The proposed *O*nline *P*ersonalized *A*ttribute-based *R*e-ranker (*OPAR*) for customers' shopping preferences imputation consists of three major components: (1) score and re-rank based on the Thompson sampling approach, (2) attribute-level parameter updates, and (3) overall procedure of *OPAR*. In general, each attribute is treated as one arm and formulated by Beta distribution. At any time step, users' in-session actions trigger the pulling of arms according to its posterior probability of being chosen as the best arms. The entire updating process is summarized in Algorithm 2. Details are introduced as follows.

3.3.3 SCORING AND RE-RANKING ITEM LIST

Given attribute-level bandits with each arm as an item attribute, the imputation process is triggered on how to score and re-rank items, motivated by the Thompson Sampling approach on (Agrawal and Goyal 2012). Shown in Algorithm 2, each attribute distribution Beta($\alpha_{atr_h}, \beta_{atr_h}$) is updated at each step t based on the users' actions A_t. To the next step $t+1$ with the given recommended item list I_{k+1}, the proposed *OPAR* model aims to rerank the list I_{k+1} based on the MAB process with respect to the updated attribute distributions.

Let N denote the number of attributes associated with item list I_t. For each attribute in $\{atr_h : atr_h \in x_i, \forall x_i \in I_t\}$, I randomly sample θ_{atr_h} from its corresponding distribution, denoting the probability that the user is interested in the attribute, atr_h, at time t:

$$\theta_{atr_h} \sim Beta(\alpha_{atr_h}, \beta_{atr_h}). \tag{3.16}$$

Algorithm 2 OPAR Algo: Re-Ranking & Parameter Update

1: **Input**:

Given a session $S = \{[Q_t, I_t, A_t]\}_{t=1}^{T}$

$\{\delta_i\}_{i:\text{ actions}}$: action-aware increments on attribute parameters

γ: hyper-parameter to control intensity on negatives

$\mathcal{U}_t$: the associated attributes from engaged items

$\mathcal{V}_t$: the associated attributes from impressed items

$|\cdot|_0$: cardinality operator

2: **Repeat for** $[Q_t, I_t, A_t] \in S$:

(1) Rerank on the Item List $f : I_t \to \tilde{I}_t$

sample $s_{atr_h} \sim Beta(\alpha_{atr_h}, \beta_{atr_h})$, $\forall atr_h \in N_S$

for $x_i \in I_t$

Given $x_i = \{atr_h\}_{h=1}^{H_{x_i}}$ as associated attributes in x_i

$score(x_i) = \sum_{atr_h \in x_i} g(s_{atr_h})$

end

(2) Update attribute parameters given A_t

Let $\mathcal{U}_t = \cup\{atr_h : \forall atr_h \in x_i \, if \, A_t(x_i) \neq 0, \forall x_i \in I_t\}$

Let $\mathcal{V}_t = \cup\{atr_h : \forall atr_h \in x_i \, \forall x_i \in I_t\}$

if $A_t(x_i) \neq 0$, item x_i has positive actions, **then**

$\alpha_{atr_h} += \delta_{A_t(x_i)} \times \{1 - Exp\,(-|\mathcal{U}_t|_0)\}$, $\forall atr_h \in x_i$

else if $A_t(x_i) = 0$, no action on item x_i, **then**

$\beta_{atr_h} += \delta_{A_t(x_i)} \times \{1 - Exp\,(-\gamma|\mathcal{V}_t \backslash \mathcal{U}_t|_0)\}$, $\forall atr_h \in x_i$

3: **Output**: All re-ranking results $[\tilde{I}_t]_{t=1}^{T}$

Then, each item $x_i \in I_t$ is scored and ranked by:

$$score(x_i) = \sum_{atr_h \in x_i} g(\theta_{atr_h}), \tag{3.17}$$

where $g(\theta_{atr_h}) = \frac{1}{rank(\theta_{atr_h})}$ is a harnomic function of the index that θ_{atr_h} is ranked among $[\theta_{atr_h}]_{h=1}^{H_{x_i}}$, with a tie-breaker uniformly at random. A larger $score(x_i)$ indicates higher satisfication with item x_i given users' short in-session preference on the attributes. Lastly, I present the user $\tilde{I}_t$, which is reranked list of the items based on $[score(x_i)]_{x_i \in I_t}$. Lists I_{t+1} is then re-ranked based on the $[score(x_i)]_{x_i \in I_{t+1}}$ and presented as $\tilde{I}_{t+1}$.

3.3.4 ATTRIBUTE PARAMETER UPDATES

With the feedback gathered from the user action A_t, the attribute parameters are updated as follows. Let $\mathcal{U}_t$ denote the set of attributes associated from items with positive actions (i.e., click, add-to-cart, purchase), and $\mathcal{V}_t$ be union of all attributes exist in $x_i \in I_t$:

$$\mathcal{U}_t = \cup\{atr_h : \forall atr_h \in x_i \,\text{if}\, A_t(x_i) \neq 0,\, \forall x_i \in I_t\}$$
$$\mathcal{V}_t = \cup\{atr_h : \forall atr_h \in x_i,\, \forall x_i \in I_t\}$$

For a given atr_h, let $\tilde{\mathcal{Y}}_{t,atr_h}$ and $\tilde{\mathcal{Z}}_{t,atr_h}$ denote the set of items associated with positive user action and no-action, respectively,

$$\tilde{\mathcal{Y}}_{t,atr_h} = \{x_i \in I_t : atr_h \in x_i \text{ and } atr_h \in \mathcal{U}_t\}$$
$$\tilde{\mathcal{Z}}_{t,atr_h} = \{x_i \in I_t : atr_h \in x_i \text{ and } atr_h \in \mathcal{V}_t \backslash \mathcal{U}_t\}$$

Then, the Beta distribution of each attribute is updated as follows:

$$\begin{aligned} \alpha_{atr_h} += & \sum_{\tilde{\mathcal{Y}}_{t,atr_h}} \delta_{A_t(x_i)} \left(1 - e^{-|\mathcal{U}_t|_0}\right), \forall atr_h \in \mathcal{U}_t \\ \beta_{atr_h} += & \sum_{\tilde{\mathcal{Z}}_{t,atr_h}} \delta_{A_t(x_i)} \left(1 - e^{-\gamma|\mathcal{V}_t \backslash \mathcal{U}_t|_0}\right), \forall atr_h \in \mathcal{V}_t \backslash \mathcal{U}_t, \end{aligned} \tag{3.18}$$

where $|\cdot|_0$ denotes the cardinality operator and γ controls intensity on implicit no-actions.

3.3.5 OPAR ALGORITHM PROCEDURE

In summary, given a session $S = \{[Q_t, I_t, A_t]\}_{t=1}^{T}$, *OPAR* can be summarized with the following steps, with the pseudo code of $OPAR_w$ shown in Algorithm 1.

1. Initialize attribute dictionary $atrDic \in R^{N\times 2}$, which contains N pairs of parameters for attributes, where each row of $atrDic$ denotes the Beta distribution parameter set $(\alpha_{atr}, \beta_{atr})$ for a given attribute. Different initialization have been experimented, including uniform, random or estimated based on held-out historical data sets (shown in Section 3.3.7).

2. At time t, scoring each item $x_i \in I_t$ based on Eq. (3.17): it first aggregates over the associated attribute preferences sampled in Eq. (3.16), and then re-rank items based on scores in Eq. (3.17) and present as $\tilde{I}_t$. More details in Section 3.3.3.

3. At time t, receiving the observation A_t on I_t, and then update the distribution of all attributes associated with item x_i in the $atrDic$ based on the Eq. (3.18) described in Section 3.3.4.

   OPAR: attribute-based bandits with *equal* action-weighting for actions in $\{\text{click}, \text{add-to-cart}, \text{purchase}\}$. This means that for positive actions, $\delta_{\text{click}} = \delta_{\text{add-to-cart}} = \delta_{\text{purchase}}$.

   OPAR$_w$: extend *OPAR* to weight action-aware updates as follows, $\delta_{\text{click}} \neq \delta_{\text{add-to-cart}} \neq \delta_{\text{purchase}}$, and hypertune them.

4. Iterative updating according to step (2) and (3) until the end of the session.

Table 3.3: Etsy Real-world Session-based Dataset Over 3 weeks

ID	Category	|Session (User)|	|Query|	|Item|	|Attributes|	|Actions|
1	**Clothing**	4642	46091	1100040	2495	58932
2	**Home & Living**	9073	103959	2282542	2455	134416
3	**Paper & Party Supplies**	4419	35132	691919	1666	55037
4	**Craft Supplies & Tools**	10913	123662	2536492	2799	171363
5	**Accessories**	5813	38215	897533	2419	49342
6	**Electronics & Accessories**	1638	10505	216860	1302	14354
7	**Jewelry**	5585	67507	1530285	2266	79874
8	**Overall Category**	26442	474594	9295453	3363	624882

3.3.6 EXPERIMENTS

Data Collection: The data set is collected and sampled from a month of user search logs at Etsy, one of the largest e-commerce platforms for handmade, vintage items, and craft supplies. To avoid bot traffic and ensure sufficient user activities within sessions, filters are added to only include search sessions with at least 10 search events (i.e., queries, browses, clicks, add-to-carts) and at least one *purchase* as I want to focus on sessions with strong shopping missions. Using an existing query classifier, each query is classified into a top probable category predicted. The most probable category (e.g. jewelry, home and living) is predicted associated with the first query of each session, and then bucket the entire session into one of 7 categories. Based on the predicted category of the first query of the session, the entire session is split into one of 7 categories, which helps to understand shopping behaviors within each category.

Table 3.3 shows statistics of each data set, representing the 7 most popular cat-

egories on the platform with nearly 500k search queries from 26k sessions and 620k user actions combined on nearly ten million items, with cardinalities computed within each data set. The evaluation is not performed on existing public data sets, because (to the best of the knowledge) there is no existing data set that includes all meta-data needed for the study (e.g. query, item attribute, user interaction logs).

3.3.7 EXPERIMENTAL SET0UP

Each of the 8 data sets is split into 2 parts (with sessions ordered chronologically). The first two-thirds of the data is a held-out data set. Focused on online learning, I only use within-session data, the held-out data set is mainly used for estimating the parameters of the Beta distributions, $\{(\alpha_{atr}, \beta_{atr})\}_{\forall atr}$, for user preferences on attributes and initializing OPAR with priors in the testing data set, and to aggregate attribute counts associated with engaged items to determine attribute popularity, powering the "Atr-POP" algorithm. The remaining data is used as testing data set, on which to report re-ranking performance for *OPAR* and other baseline algorithms on in Table 3.4.

While *OPAR* can function as a stand-alone ranking algorithm, OPAR (as well as other baselines) is evaluated on top of an existing 2-pass ranking system (as described in Figure 3.2). More formally, each session in the testing data set, $S = \{[Q_t, I_t, A_t]\}_{t=1}^{T}$ contains a sequential list of query content Q_t, a candidate set I_t of items to be re-ranked, truncated to the size to 48 (i.e., 48 items are shown per search page on the platform) after the second-pass re-ranking on hundreds of items from the system, and logged user actions A_t on I_t (e.g. click, purchase). In the experiments, I_t is a truncated list of the top 48 items returned by an existing 2-pass ranker, indicating that this list comprises of the most relevant items to the query. As shown in experimental results, applying *OPAR* adds an effective layer of attribute-based personalization in

real-time that was not feasible with the underlying system. In order to simulate an online environment, only within-session user interactions leading up to the current time step are used for ranking predictions.

3.3.8 EVALUATION METRICS AND BASELINES

Evaluation Metrics: Below, I describe the offline metrics I use to evaluate *OPAR* on the testing data set, as well as the benchmark baselines. Following the general ranking metric Normalized Discounted Cumulative Gain (NDCG) (Wang *et al.* 2013), I propose a set of session-level ranking metrics to evaluate the model. Given a session, $S = \{[Q_t, I_t, A_t]\}_{t=1}^{T}$, it contains a sequential list of query content Q_t, a candidate set of items I_t with the initial ordering based on the 2nd-pass re-ranking, and user actions A_t that provides the groundtruth for relevances (i.e, clicks or purchases as relevances in evaluating click-NDCG and purchase-NDCG). Let $\tilde{I}_t$ be the re-ranked list of I_t given a re-ranking algorithm.

1. *Click-NDCG*: For each query Q_t issued in S that has at least one click in A_t (i.e, clicks as relevances), *click-NDCG*$_t$ measures the re-ranking performance of the item list $\tilde{I}_t$ (after re-ranking I_t) shown to the user at t. For all timestamp with at least a click, I first compute stepwise sequential re-ranking performance *click-NDCG*$_t$ as:

$$\textit{click-NDCG}_t = \textit{click-DCG}_t / IDCG_t, \forall t = 1, ...T, \tag{3.19}$$

   and *click-NDCG* of a session S is the average of *click-NDCG*$_t$ over events that have at least one click:

$$\textit{click-NDCG} = \text{Average}(\textit{click-NDCG}_t). \tag{3.20}$$

2. *Purchase-NDCG*: Following the above methodology, I compute the session-level re-ranking performance limit to search events with attributed purchases. A

session on a shopping site is defined as a sequence of events ending with a purchase or a significant duration of inactivity. Given that, *Purchase-NDCG* given a session is essentially *purchase-NDCG*$_T$.

For each re-ranking algorithm reported in Table 3.4, I compute *Click-NDCG* @k and *Purchase-NDCG*@k for each $k = \{4, 12, 24, 48\}$ by averaging *click-NDCG*$_s$ @k and *purchase-NDCG*$_s$@k given session s over all sessions in each data set. Note that k is a multiple of 4 as that this shopping site displays 4 items per row on desktops.

Baselines *OPAR*'s ranking performance is compared with 4 state-of-the-art baselines:

1. LambdaMART (Wu *et al.* 2010) is the boosted tree version of LambdaRank (Burges *et al.* 2007), which introduces the use of gradient boosted decision trees for solving a ranking task and won Track 1 of the 2010 Yahoo! Learning To Rank Challenge. A personalized search re-ranker is trained based on long-term user historical data to optimize for the user's purchasability on an item given the query issued and the user's historical preference.

2. Atr-KNN is derived from Item-KNN (Hidasi *et al.* 2015). Each item is presented by n-hot-encoding of associated attributes with n being the cardinality of all attributes. That is, its i^{th} entry equals to 1 if the referred attribute presents in the item, otherwise 0. Items in the list I_{t+1} are re-ranked based on their euclidean-distance from the last engaged item(s) in I_t. Note that the items $x_i \in I_t$ with no-action has no impact on this re-ranking.

3. Atr-POP reranks the candidate set, I_t, of items based on the attributes' popularity estimated with held-out historical records. This baseline is one of the most common solutions derived from (Hidasi *et al.* 2015) given its simplicity and efficacy.

4. GRU4Rec (Hidasi *et al.* 2015) applies recurrent neural networks (RNN) on short session-based data of clicked items to achieve session-based next-item recommendation. Each session is encoded as a 1-of-N vector, in which the i^{th} entry is 1 if the corresponding item is clicked else 0, with N denoting the number of items. While the user's consecutive clicks on items are used in the next item prediction, it is attribute-agnostic. The RNN model is trained and used on for the sequential session data and a session-parallel mini-batches algorithm is proposed for sampling.

While it is common for each arm in the bandits to represent a single item or product category, I skip it as a baseline here as this would incur higher exploration cost with potential latency bottleneck when scaling up to an inventory of hundred millions of items and also lose interpretability of product attributes.

3.3.9 EFFECTIVENESS OF ACTION-AWARE MABS

Table 3.4 shows experiment results of the proposed model (*OPARs*) against 4 baselines described in Section 3.3.8. The results can be categorized into two parts: (1) performance on the aggregated data sets over all categories (top-left); and (2) performance on each of the 7 category-specific data sets, representing different shopping missions and behaviors across categories (i.e, "Clothing", "Home & Living"). Across all 8 data sets for the re-ranking task, $OPAR_w$ outperform against all 4 baselines, including LambdaMART, Atr-KNN, Atr-POP, and GRU4Rec in both purchase-NDCG and click-NDCG.

For the overall data set (top-left), $OPAR_w$ shows over 6% lift in click-NDCG@48 compared to the best baseline, and over 20% increase in purchase-NDCG@48. Similar results are observed in each category-specific re-ranking. For k, the best improvement for $OPAR_w$ is achived at $k = 4$, ordering by @4 >> @12 >> @24 >> @48. With

attribute-based bandits, interactive feedbacks from the in-session user actions, even just fewer clicks, efficiency propagate rewards to associated attributes and quickly learns preferred attributes that matter the most to the user, thus optimize user purchase intent.

To explore users' in-session activity with different types of actions (i.e, click, add-to-cart), I run experiments with the action-aware bandit model, with $OPAR_w$ hyper-tuned rewards from clicks vs add-to-carts, to differentiate *types* of user actions. The results in Table 3.4 are reported from a tuned model that assigns larger weights to *clicks* than *add-to-carts*, with an intuition that there is a high topical drift observed in the user's browsing intent after items are added to carts. As shown in Table 3.4, collectively $OPAR_w$ outperforms $OPAR$ by 1.6% and 1.1% in purchase NDCG@4 and click NDCG@4, respectively. When segmenting by categories, $OPAR_w$ also outperforms $OPAR$ in almost all categories, except *Electronics & Accessories* and *Craft Supplies & Tools* on purchase NDCG.

3.3.10 INTERPRETABILITY OF WITHIN-SESSION SHOPPING MISSION

It is often observed that a user exhibits multiple purchase intents with diverse preferences within a session. Table 3.5 presents a record of a user's in-session activities. Figure 3.5 (top) shows the sequential improvement of $OPAR$ in session-level click-NDCG over time compared to the baseline, and Figure 3.5 (bottom) shows how $OPAR$ captures user's preference, θ_{atr_h}, on 5 attributes over time. The "Engaged Attributes" column in Table 3.5 maps out all attributes associated with the clicked items for the corresponding query.

As shown in Table 3.5, the user is interested in three categories as his/her purchase intents: first in "paper & party supplies", then drift to "women clothing" and "accessories", and lastly converted in "accessories" with a *purchase*. After the browsing

period from timestamp $t = 0$ with no user actions, β_{atr} for the attributes associated with the browsing-only items are incremented while no attributes have been updated with positive rewards for the given user. *OPAR* launches from a lower click-NDCG at the beginning, while obtains better re-ranking performance compared with baseline by learning that the user is interested in white prime color and is looking for the wedding occasion theme by the end of $t = 4$. From then *OPAR* outperforms the baseline in click NDCG while activated more attributes related to wedding themes in beach and tropical and expanded to floral crafting type and blue for prime color. The re-ranking performance continues to improve from $t = 5, .., 9$ as more items related to these attribute themes are discovered.

Starting from $t = 12$, the user starts to explore the 2nd categorical purchase intent, pivoting from "paper and party supplies" to "clothing" and "accessories". However, the latest activated attributes based on the engaged items on the first set of shopping queries still relevant. The user has a consistent preference in attributes, such as "Prime Color: Blue", "Occasion: Wedding", and "Wedding theme: Fairytale & princess" as she is searching for a "hat for beach wedding" and/or "bride hair decoration beach theme". Thus, for the second purchase intent starting at $t = 12$, I observe a high jump start in *OPAR*'s click NDCG at $t = 12$ comparing to the first intent at $t = 1$ and the metric continues to stepwise improve. As demonstrated in Figure 3.5 (bottom), "wedding theme" and "primary color: blue" are the top two performant attributes that *OPAR* learned and identified over time.

Figure 3.5: In-session *OPAR* Re-Ranking Performance.

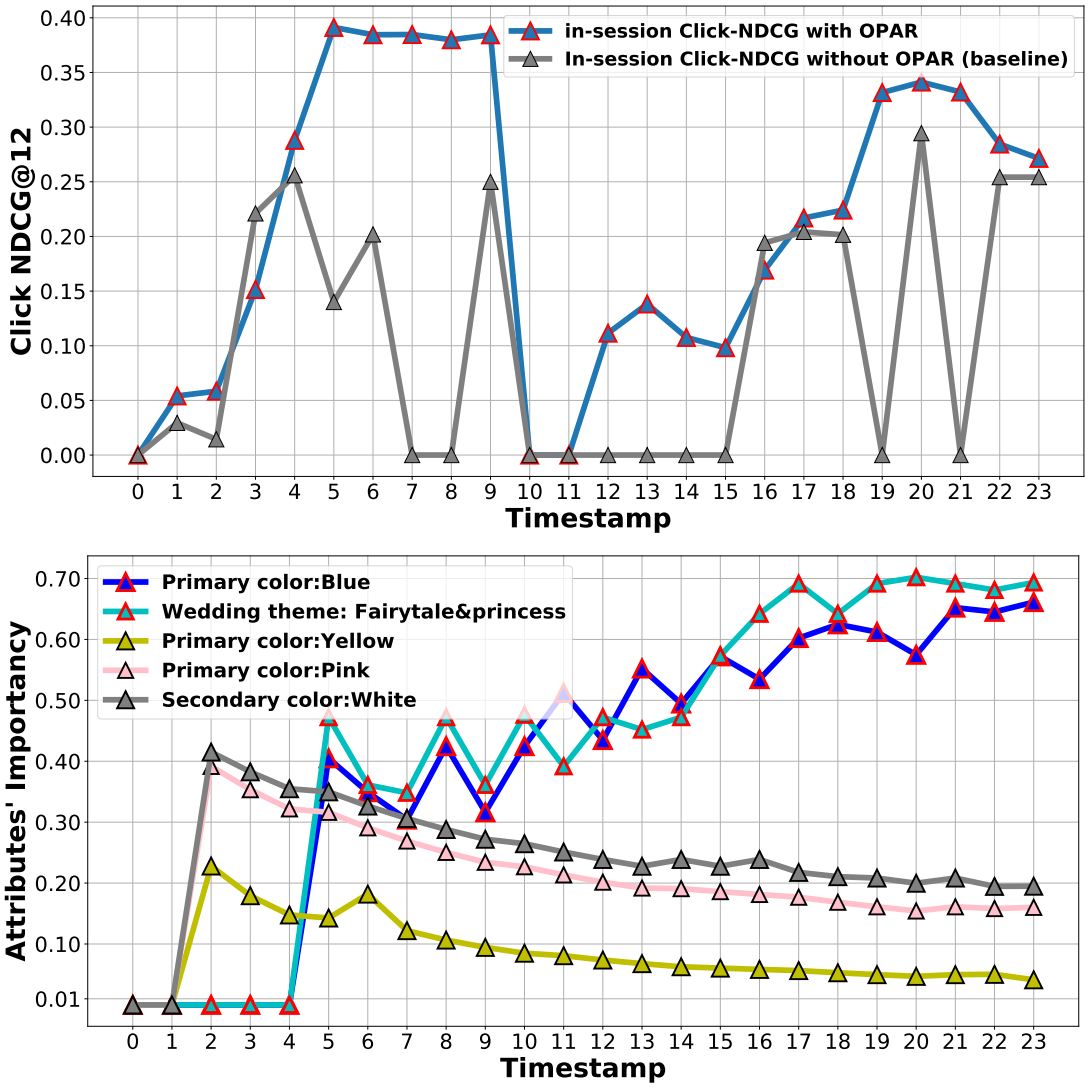


Table 3.4: Re-ranking Performance Comparison on 7 Category-specific Data Sets.

		Over All Category						Clothing					
		LambdaMART	Atr-KNN	Atr-POP	GRU4Rec	OPAR	$OPAR_w$	LambdaMART	Atr-KNN	Atr-POP	GRU4Rec	OPAR	$OPAR_w$
	@4	0.1795	0.0130	0.0749	0.0618	0.2994	**0.3042**	0.1948	0.0103	0.0516	0.0551	0.2384	**0.2494**
Purchase	@12	0.2629	0.0412	0.1323	0.1425	0.3505	**0.3607**	0.2670	0.0348	0.1269	0.0824	0.2685	**0.2744**
NDCG	@24	0.3162	0.1260	0.2112	0.2018	0.3718	**0.3900**	0.3019	0.0090	0.2193	0.1434	0.3209	**0.3263**
	@48	0.3724	0.2554	0.2861	0.2518	0.4512	**0.4578**	0.3774	0.2462	0.2784	0.2157	0.3976	**0.4030**
	@4	0.1459	0.0816	0.0705	0.0701	0.3120	**0.3158**	0.1328	0.0067	0.0690	0.0691	0.3058	**0.3197**
Click	@12	0.2265	0.1456	0.1264	0.1354	0.3213	**0.3229**	0.2137	0.0228	0.1224	0.1414	0.3126	**0.3257**
NDCG	@24	0.2955	0.2157	0.2021	0.1922	0.3318	**0.3489**	0.2821	0.0658	0.2045	0.1844	0.3274	**0.3424**
	@48	0.3815	0.3245	0.2813	0.2689	0.4047	**0.4051**	0.3711	0.2309	0.2807	0.2613	0.3988	**0.4061**

		Home & Living						Paper & Party Supplies					
		LambdaMART	Atr-KNN	Atr-POP	GRU4Rec	OPAR	$OPAR_w$	LambdaMART	Atr-KNN	Atr-POP	GRU4Rec	OPAR	$OPAR_w$
	@4	0.1755	0.0131	0.0649	0.0571	0.2920	**0.2952**	0.1822	0.0010	0.1255	0.0684	0.2828	**0.2965**
Purchase	@12	0.2670	0.0396	0.1226	0.1281	0.3391	**0.3436**	0.2667	0.0406	0.1692	0.0941	0.3367	**0.3497**
NDCG	@24	0.3218	0.0936	0.2066	0.1752	0.3838	**0.3879**	0.3276	0.1297	0.2469	0.1542	0.3796	**0.3905**
	@48	0.3874	0.2543	0.2789	0.2164	0.4462	**0.4491**	0.3876	0.2550	0.3216	0.1943	0.4291	**0.4399**
	@4	0.1481	0.0054	0.0601	0.0944	0.3201	**0.3219**	0.1585	0.0052	0.1084	0.0839	0.2825	**0.2874**
Click	@12	0.2294	0.0213	0.1175	0.1416	0.3244	**0.3256**	0.2394	0.0247	0.1586	0.1367	0.2931	**0.2973**
NDCG	@24	0.2978	0.0598	0.1973	0.1843	0.3485	**0.3491**	0.3103	0.0644	0.2300	0.1742	**0.3383**	0.3189
	@48	0.3835	0.2278	0.2746	0.2288	0.4032	**0.4086**	0.3911	0.2306	0.3104	0.2007	0.4017	**0.4072**

		Craft Supplies & Tools						Accessories					
		LambdaMART	Atr-KNN	Atr-POP	GRU4Rec	OPAR	$OPAR_w$	LambdaMART	Atr-KNN	Atr-POP	GRU4Rec	OPAR	$OPAR_w$
	@4	0.1912	0.0135	0.0739	0.0741	0.3101	**0.3268**	0.1954	0.0251	0.0683	0.0511	0.2166	**0.2178**
Purchase	@12	0.2735	0.0407	0.1296	0.1125	0.3673	**0.3781**	0.2828	0.0741	0.1431	0.0849	**0.2835**	0.2930
NDCG	@24	0.3272	0.1208	0.1970	0.1644	0.4084	**0.4188**	0.3324	0.1406	0.2510	0.1222	0.3304	**0.3361**
	@48	0.3844	0.2577	0.2820	0.2214	0.4366	**0.4750**	0.3869	0.2693	0.2917	0.1641	0.3962	**0.4020**
	@4	0.1458	0.0055	0.0749	0.0994	0.3118	**0.3166**	0.1502	0.0105	0.0673	0.0712	0.2495	**0.2605**
Click	@12	0.2262	0.0513	0.1290	0.1279	0.3241	**0.3293**	0.2324	0.0439	0.1358	0.1331	0.2656	**0.2708**
NDCG	@24	0.2955	0.2042	0.1953	0.1935	**0.3525**	0.3521	0.3006	0.1091	0.2398	0.1800	0.3155	**0.3212**
	@48	0.3811	0.2278	0.2815	0.2277	**0.4080**	0.4078	0.3848	0.2391	0.2885	0.2312	0.3548	**0.4029**

		Electronics & Accessories						Jewelry					
		LambdaMART	Atr-KNN	Atr-POP	GRU4Rec	OPAR	$OPAR_w$	LambdaMART	Atr-KNN	Atr-POP	GRU4Rec	OPAR	$OPAR_w$
	@4	0.2136	0.0501	0.0715	0.0814	0.2847	**0.2995**	0.1661	0.0060	0.0576	0.0718	0.3051	**0.3285**
Purchase	@12	0.3014	0.1109	0.1546	0.1223	0.3386	**0.3782**	0.2534	0.0766	0.1074	0.1142	0.3484	**0.3854**
NDCG	@24	0.3519	0.0176	0.2652	0.1674	**0.4257**	0.4152	0.3087	0.1470	0.1668	0.1847	0.3866	**0.3973**
	@48	0.4060	0.2965	0.2981	0.2416	0.4516	**0.4656**	0.3814	0.2460	0.2663	0.2367	0.4425	**0.4598**
	@4	0.1530	0.0267	0.0805	0.0641	**0.2074**	0.2051	0.0701	0.0027	0.0621	0.0614	0.3314	**0.3892**
Click	@12	0.2324	0.0703	0.1580	0.0939	0.2487	**0.2622**	0.1314	0.0106	0.1141	0.1021	0.3783	**0.3963**
NDCG	@24	0.3029	0.1410	0.2657	0.1345	**0.3158**	0.3120	0.1989	0.1276	0.1762	0.1647	0.3956	**0.4162**
	@48	0.3880	0.2560	0.3026	0.1667	0.3978	**0.4078**	0.3119	0.2192	0.2700	0.2144	0.4190	**0.4475**

Table 3.5: Multiple Purchase Intents within One Session

	Timestamp	Query	Query Taxonomy		Engaged Attributes
1st Purchase Intent	0	'flower girl basket'	paper and party supplies	(NO ACTION)	Browsing
	1-4	'flower girl basket wedding'	paper and party supplies	(CLICK)	'Prime Color: White', 'Occasion: Wedding', 'Holiday: Christmas', 'Wedding theme: Beach & tropical', 'Craft type: Floral arranging'
	5-9	'flower girl basket beach wedding'	paper and party supplies	(CLICK)	'Prime Color: Blue', 'Occasion: Wedding', 'Holiday: Christmas', 'Wedding theme: Beach & tropical', 'Secondary color: White', 'Craft type: Floral arranging'
	10-11	'two flower girl and one pillow'	paper and party supplies		Browsing
Purchase Intent Change					
2nd Purchase Intent	12-15	'hat for beach wedding'	clothing.women_clothing	(CLICK)	'Prime Color: Blue', 'Occasion: Wedding'
	16-22	'turquoise petals'	accesories	(CLICK)	'Prime Color: Blue', 'occasion: Bridal shower', 'Wedding theme: Fairytale & princess'
	23	'bride hair decoration beach theme'	clothing.women_clothing	(NO ACTION)	Browsing
Final Purchase	24	'turquoise petals'	accesories	(PURCHASE)	'Prime Color: Blue', 'Occasion: Bridal shower', 'Wedding theme: Fairytale & princess'

Chapter 4

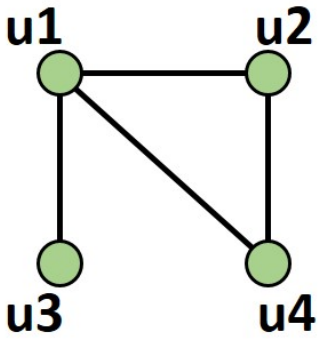
MISSING VALUE IMPUTATION FOR HEALTHCARE ANALYSIS

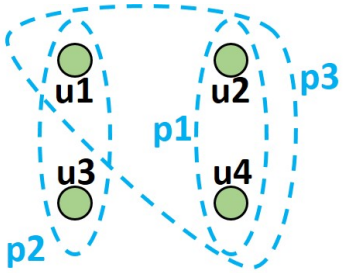
Patients with the same type of ailment, such as diabetes mellitus, can join and use condition-specific social networks. Their purpose is to facilitate information sharing and the establishment of support groups, which will assist sufferers maintain a healthy lifestyle while dealing with the disease.

Although users frequently report their biomarker measurements as their condition progresses, due to the voluntary nature of disease-specific social networks, such self-reported measurements contain a large amount of missing information, as very few users report their measurements every time they take the test. However, having a reasonable estimate of such missing information is critical for monitoring reasons, so that reminders or warnings may be sent in time to help users get back on track. In contrast, people frequently have access to heterogeneous auxiliary data in addition to the observed information in order to estimate missing information and improve imputation performance. For example, in addition to self-reported measurements, I can use the rich social relations present in a large number of frequent visitors, users, such as friend-friend relationships and follower-followee relationships, to estimate missing biomarker measurements from disease-specific social networks. Furthermore, auxiliary clinical data with potentially non-overlapping users, in addition to disease-specific social networks, may provide an essential trend about the advancement of biomarker measures, and thus can assist enhance the performance of missing value imputation.

In summary, there are two main challenges when imputing the missing value in disease-dedicated social networks:

	u1	u2	u3	u4
p1	0	1	0	1
p2	1	0	1	0
p3	1	1	0	1





(a) incident matrix. (b) user-user. (c) user-post hypergraph.

Figure 4.1: Illustration of the hypergraph representation. In sub-figure (a), for example, the user u1 leaves his/her comment in the post p1, as the corresponding value equals to 1 in the incident matrix, otherwise 0. In the sub-figure (b), which indicates the users' grouping information, user u1 and user u2 are connected as both of them have participated in the post p1, while this graph cannot tell us how many users are involved in the same post. The sub-figure (c) is the user-post hypergraph which contains the completed user grouping information of each post on the disease-dedicated social network.

- How to preserve the patient grouping information without losing the user-post relationship.

- How to propagate the patient grouping information from social media forum to clinical data analysis.

4.1 PATIENT GROUPING AND SIMILARITY MEASURES

The follower/followee relationship between two users is often used to quantify pairwise similarity in an online forum. When the follower/followee link is observed, two people are usually thought to be similar. When establishing the aim function

in modeling processing, the so-called graph regularized is proposed to measure the associated users' similarity.

However, in the context of disease-specific social networks, this pairwise similarity might result in a large loss of user grouping information, which discloses useful healthcare knowledge. On a disease-specific forum, for example, the graph regularizer is typically used to establish the users' pairwise similarity, or how comparable these two users are when it comes to physical/medial measurements such as normal blood sugar levels or the level of oral glucose tolerance test. However, it is common for more than two individuals to respond to the same discussion topic, indicating that they are both suffering from the same ailment. The same post connects multiple users (nodes) (edge). The pairwise similarity of the patients is insufficient to capture such categorization information.

I propose using the hypergraph structure to describe disease-specific social networks. Each hyperedge in the hypergraph structure corresponds to one thread (post) and connects several individuals who have engaged in the thread, allowing the aforementioned grouping information to be efficiently preserved. For example, on an online disease-specific forum, one group of users is connected by a topic where they primarily discuss glucose levels, while another group is connected by a topic where they discuss insulin pumps. When creating the analysis on the healthcare forum, these two groups of members present two different aspects. The hypergraph structure allows for the retention of such grouping information, i.e., users in the same group (post) are formulated to be close to each other in terms of the hypergraph regularizer, whilst users from different groups maintain a relatively large gap between them. Users who can be shown in both groups (overlapped users), i.e., he or she has participated in both threads at the same time, which can also be reserved in the hypergraph regularizer but regrettably not in the classic graph regularizer. The hyperedge weight shows the

popularity of each post (hyperedge), such as how many users replied or how long the debate lasted.

Fig. 4.1 shows a simple example of the hypergraph representation. $User = \{u_1, u_2, u_3, u_4\}$ and $Post = \{p_1, p_2, p_3, p_4, p_5\}$ denote the user set and post set respectively. The incident matrix in Fig. 4.1a has the entry $(p_i, u_j) = 1$ if user u_j participates in the post p_i; the traditional graph model in Fig. 4.1b shows how the pairs of users are connected when they participate in the same post, while user grouping information for each thread is lost. The traditional graph structure cannot reveal whether the same user left comments under multiple posts, while such kind of grouping knowledge loss is not expected for data mining purposes because the posts with the same user are likely to belong to the same topic, or contain the patients' daily continuous biomarker measurements. The hypergraph in Fig. 4.1c fully describes the user-post grouping relationship when I treat each post as one hyperedge. The connection between each user and the user grouping knowledge for each post is completed illustrated. Thus the high-order relationships among users can be captured by hypergraph structures without loss of any information.

4.2 PROBLEM DEFINITION

Shown in Fig. 4.2, our goal is to impute the missing value existing in the patient biomarker data (i.e., Y_0) by exploiting the information from heterogeneous auxiliary sources (e.g., clinical trial data, disease-dedicated social network) together with strict constraint on the observation information. Based on the matrix factorization framework, our goal is motivated by two aspects:

Latent Coherence: The latent coherence spreads among the similar users from heterogeneous auxiliary sources $\mathbf{M}$ (e.g., diabetic patients), while the samples in $\mathbf{M}$ do not necessarily overlap with the samples in $\mathbf{Y}_0$.

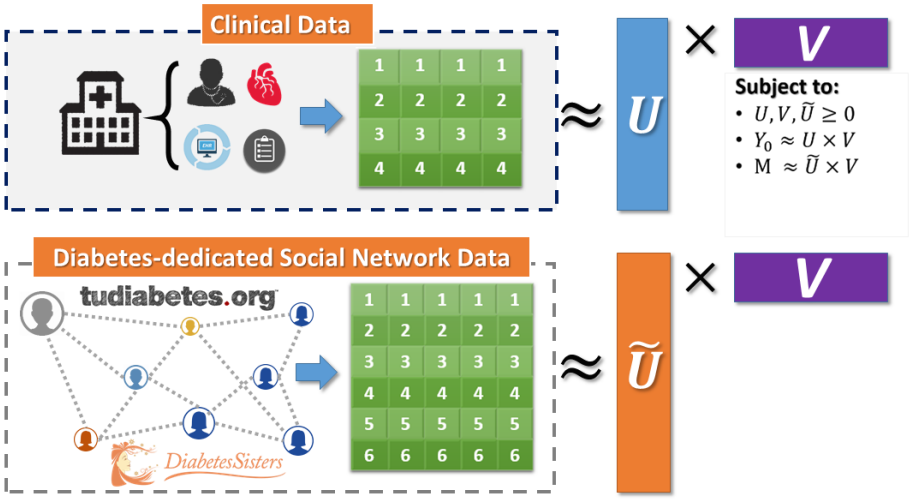


Figure 4.2: Missing Information Imputation With Auxiliary Data.

Observation Consistency: The incomplete data Y_0 is factorized into two low dimensional matrices $\mathbf{U}$ and $\mathbf{V}$, while the fact is $\{\mathbf{U}\mathbf{V}^\top\}_\Omega \neq (\mathbf{Y}_0)_\Omega$, where $(\cdot)_\Omega$ index to the observed data in $\mathbf{Y}_0$. The factorization process disturbs the original observed value in $\mathbf{Y}_0$ due to each factor $(\mathbf{U}, \mathbf{V})$ is estimated based on the global information from $\mathbf{Y}_0$. Ideally, I expect the missing entries in $\mathbf{Y}_0$ to be filled up by incorporating the auxiliary information, and meanwhile, keeping $\mathbf{U}\mathbf{V}^\top$ consistent with the observed information in $\mathbf{Y}_0$ as much as possible.

4.3 PROPOSED FRAMEWORK

Given the clinical biomarker data $\mathbf{Y}_0 \in R^{m \times n}$ with m users and n category of biomarker, the disease-dedicated forum data $\mathbf{M} \in R^{m' \times n}$ is collected with the number of m' users and the posts corresponding to the n topics. Let $\mathcal{V}$, $\mathcal{E}$ denote the use (vertex) set and forum post (hyperedge) set respectively. The forum is presented as the hypergraph $G(\mathcal{V}, \mathcal{E}, \mathcal{W})$ with the vertex set $\mathcal{V}$, hyperedge set $\mathcal{E}$, and the hyperedge

weight knowledge $\mathcal{W}$. The weighted hypergraph contains the hyperedge weight $w(e) \in \mathcal{W}$ associated with each hyperedge $e \in \mathcal{E}$. For each vertex $v \in \mathcal{V}$, the vertex degree $d(v)$ is defined as $d(v) = \sum_{\{e \in \mathcal{E} | v \in e\}} w(e)$. The incident matrix $\mathbf{H}$ in the size $|\mathcal{V}| \times |\mathcal{E}|$ indicates whether the the user-post connection, that

$$h(v_i, e_j) = \begin{cases} 1, \ if \ v_i \in e_j \ (user \ v_i \ appear \ in \ the \ post \ e_j) \\ 0, \ otherwise \end{cases} \tag{4.1}$$

For each hyperedge e, the hyperedge degree $\delta(e)$ is defined as $\delta(e) = \sum_{v \in \mathcal{V}} h(v, e)$, which indicates how many users leave their commons under the post e. The diagonal matrix $\mathbf{D}_v$ in the size $|\mathcal{V}| \times |\mathcal{V}|$ has its diagonal elements equal to the degree of each vertex, and the diagonal matrix $\mathbf{D}_e$ in the size $|\mathcal{E}| \times |\mathcal{E}|$ has its diagonal elements equal to the degree of each hyperedge.

Analogous to the definition of Laplacian matrix in the normal graph (Cai *et al.* 2008), the hypergraph Laplacian matrix $\mathbf{L}^h \in R^{m \times m}$ is defined as:

$$\mathbf{L}^h = \mathbf{D}_v - \mathbf{H}\mathbf{W}\mathbf{D}_e^{-1}\mathbf{H}^\top \tag{4.2}$$

The patients' grouping similarity is incorporated through the hypergraph constraint as:

$$\begin{aligned} Tr(\mathbf{U}^\top \mathbf{L}^h \mathbf{U}) =& Tr(\mathbf{U}^\top \mathbf{D}_v \mathbf{U}) - Tr(\mathbf{U}^\top \mathbf{H}\mathbf{W}_e \mathbf{D}_e^{-1} \mathbf{H}^\top \mathbf{U}) \\ =& \sum_{e \in \mathcal{E}} \sum_{(v_i, v_j) \in e} \frac{w(e)}{\delta(e)} \|v_i - v_j\|^2 \end{aligned} \tag{4.3}$$

where the distance between the nodes v_i and v_j within each hyperedge, weighted by the $\frac{w(e)}{\delta(e)}$, is inclined to short. By incorporating the hypergraph structure with the matrix factorization method, the formulation of our framework, named MI^2-HD, is to design as the following optimization problem:

$$\begin{aligned} \min_{\mathbf{Y}, \mathbf{U}, \tilde{\mathbf{U}}, \mathbf{V}} & ||\mathbf{Y} - \mathbf{U}\mathbf{V}^\top||_F^2 + \alpha ||\mathbf{M} - \tilde{\mathbf{U}}\mathbf{V}^\top||_F^2 + \beta Tr(\tilde{\mathbf{U}}^\top \mathbf{L}^h \tilde{\mathbf{U}}) \\ & s.t. \ \mathbf{U} \geq 0, \tilde{\mathbf{U}} \geq 0, \mathbf{V} \geq 0, \mathbf{Y}_\Omega \equiv (\mathbf{Y}_0)_\Omega \end{aligned} \tag{4.4}$$

$\mathbf{Y} \in R_+^{m \times n}$, $\mathbf{U} \in R_+^{m \times k}$, $\mathbf{V} \in R_+^{n \times k}$, $\mathbf{M} \in R_+^{m' \times n}$, $\tilde{\mathbf{U}} \in R_+^{m' \times k}$, and $\mathbf{L}^h \in R^{m \times m}$, where m and m' denote the number of user in original data and auxiliary data respectively, and n denotes the number of feature. The trade-off parameters $\alpha, \beta \geq 0$, that β controls the effectiveness of the hypergraph structure. Matrices $\mathbf{U}$ and $\tilde{\mathbf{U}}$ indicate the row clustering (sample grouping), and matrix $\mathbf{V}$ indicates the column clustering (measurement grouping). The latent coherence among the heterogeneous data $\mathbf{M}$ and original data $\mathbf{Y}_0$ is required by sharing the same measurement matrix $\mathbf{V}$, while users are constrained by their online activities observed from the disease-dedicated forum.

Thus, when minimizing the Eq. (4.4), the similarity of the vertices associated with the same hyperedge keeps constant. In other words, considering the practical disease-dedicated forum, the users who share their experience at the same post are expected to be relevant to each other, that this kind of grouping relation is encoded in the Eq. (4.3) as the similarity among these nodes keeping the same within each hyperedge. Users may discuss different topics in different posts, then the node (user) grouping information is altered regarding the hyperedge (post). For structural convenience, in this case, I set hyperedge weight to be equal, and the weight effect will be explored in our future.

Extension to Tri-Factorization: Closely related to MI^2-HD, I propose MI^2-HT by leveraging the tri-factor matrix factorization model, with non-negativity and orthogonality constraints on each factorization matrix. Compared with the two-factor matrix factorization mentioned above, which may provides a relatively weak low-rank approximation (Wang *et al.* 2011), (Ding *et al.* 2006) introduced one more factorization factor $\mathbf{S}$ into consideration. In this model, the observed data matrix $\mathbf{Y}_0$ is approximated by three factors $\mathbf{U}$, $\mathbf{S}$ and $\mathbf{V}$, that factor S is designed to absorb the different scales of $\mathbf{U}$ and $\mathbf{V}$. In the meanwhile, the auxiliary data $\mathbf{M}$ is also tri-factorized by sharing the same measurement matrix $\mathbf{V}$. The objective function of

MI^2-HT is formulated as:

$$\min_{\mathbf{U},\mathbf{V},\tilde{\mathbf{U}},\mathbf{S},\tilde{\mathbf{S}}} ||(\mathbf{Y}_0 - \mathbf{U}\mathbf{S}\mathbf{V}^\top)_\Omega||_F^2 + \alpha||\mathbf{M} - \tilde{\mathbf{U}}\tilde{\mathbf{S}}\mathbf{V}^\top||_F^2 + \beta Tr(\mathbf{U}^\top \mathbf{L}^h \mathbf{U}) \\ s.t.\, \mathbf{U}, \tilde{\mathbf{U}}, \mathbf{V}, \mathbf{S}, \tilde{\mathbf{S}} \geq 0, \mathbf{U}\mathbf{U}^\top = \mathbf{I}, \tilde{\mathbf{U}}\tilde{\mathbf{U}}^\top = \mathbf{I}, \mathbf{V}\mathbf{V}^\top = \mathbf{I} \tag{4.5}$$

where $\alpha, \beta \geq 0$, the matrix $\mathbf{M}$ denotes the auxiliary data, collected from the diabetes-dedicated social networks. To avoid ambiguity, the orthogonal constraint on factorization matrices $\mathbf{U}$, $\tilde{\mathbf{U}}$ and $\mathbf{V}$ require only one non-zero entry in each row, which forces each user/biomarker to only belong to a single clustering class.

4.4 OPTIMIZATION ALGORITHM

The proposed optimization problem is solved by the joint matrix factorization. A set of multiplicative updating rules are proposed to solve the optimization problem.

4.4.1 MI^2-HD UPDATING RULES

There are two iterative updating steps in MI^2-HD, as shown in Algorithm 3. Since Eq. (4.4) is convex for the variables $\mathbf{Y}$, $\mathbf{U}$, $\tilde{\mathbf{U}}$, $\mathbf{V}$ separately (See section Convergence Analysis), I propose to update $\mathbf{Y}$ and $\mathbf{U}$, $\tilde{\mathbf{U}}$, $\mathbf{V}$ separately. In the Algorithm 3, the convergence criteria is defined as the average changing rate of the imputation result. When the average changing of the imputation value is less than 1E-2, the updating process is considered as converged.

Fix Y, update U, $\tilde{U}$, V: I first introduce how to update $\mathbf{U}$, $\tilde{\mathbf{U}}$, $\mathbf{V}$ with fixing $\mathbf{Y}$ by minimizing the Eq. (4.4). The Eq. (4.4) is then extended into the following

form:

$$
\begin{aligned}
O =& Tr[(\mathbf{Y}-\mathbf{U}\mathbf{V}^\top)(\mathbf{Y}-\mathbf{U}\mathbf{V}^\top)^\top] + \beta Tr[\mathbf{U}^\top\mathbf{L}^h\mathbf{U}] + \alpha Tr[(\mathbf{M}-\tilde{\mathbf{U}}\mathbf{V}^\top)(\mathbf{M}^\top-\mathbf{V}\tilde{\mathbf{U}}^\top)] \\
=& Tr[(\mathbf{Y}_0-\mathbf{U}\mathbf{V}^\top)_\Omega(\mathbf{Y}_0^\top-\mathbf{V}\mathbf{U}^\top)] + \beta Tr(\mathbf{U}^\top\mathbf{L}^h\mathbf{U}) + \alpha Tr(\mathbf{M}\mathbf{M}^\top) \\
& - 2\alpha Tr(\mathbf{M}\mathbf{V}\tilde{\mathbf{U}}^\top) + \alpha Tr(\tilde{\mathbf{U}}\mathbf{V}^\top\mathbf{V}\tilde{\mathbf{U}}^\top)
\end{aligned}
$$

by introducing the Lagrangian function $\mathscr{L}$ and Lagrange multipliers $\mathbf{\Psi}_{ij}$, $\mathbf{Phi}_{ij}$, and $\mathbf{\Gamma}_{ij}$. Each multiplier $\mathbf{\Psi}_{ij}$, $\mathbf{\Phi}_{ij}$, and $\mathbf{\Gamma}_{ij}$ corresponds to the constraints $\mathbf{U}_{ij} \geq 0$, $\mathbf{V}_{ij} \geq 0$ and $\tilde{\mathbf{U}}_{ij} \geq 0$ respectively. The Lagrange function $\mathscr{L}$ can be written as:

$$
\mathscr{L} = O(\mathbf{Y},\mathbf{U},\mathbf{V},\tilde{\mathbf{U}}) + Tr(\mathbf{\Psi}\mathbf{U}^\top) + Tr(\mathbf{\Phi}\mathbf{V}^\top) + Tr(\mathbf{\Gamma}\tilde{\mathbf{U}}^\top)
$$

The partial derivatives of $\mathscr{L}$ with respect to U, V and $\tilde{U}$ are:

$$
\begin{aligned}
\frac{\partial\mathscr{L}}{\partial\mathbf{U}} =& -2Tr(\mathbf{Y}\mathbf{V}) + \frac{\partial Tr(\mathbf{U}\mathbf{V}^\top\mathbf{V}\mathbf{U}^\top)}{\partial\mathbf{U}} \\
& + \beta\mathbf{L}\mathbf{U} + \beta\mathbf{L}^\top\mathbf{U} + \mathbf{\Psi} \\
\frac{\partial\mathscr{L}}{\partial\mathbf{V}} =& -2Tr(\mathbf{Y}\mathbf{U}) + \frac{\partial Tr(\mathbf{U}\mathbf{V}^\top\mathbf{V}\mathbf{U}^\top)}{\partial\mathbf{V}} \\
& - \alpha\mathbf{M}^\top\tilde{\mathbf{U}} + \alpha\mathbf{V}\tilde{\mathbf{U}}^\top\tilde{\mathbf{U}} + \mathbf{\Phi} \\
\frac{\partial\mathscr{L}}{\partial\tilde{\mathbf{U}}} =& -2\alpha\mathbf{M}\mathbf{V} + 2\alpha\tilde{\mathbf{U}}\mathbf{V}^\top\mathbf{V} + \mathbf{\Gamma}
\end{aligned}
$$

by setting each partial derivative to 0, based on the Karush-Kuhn-Tucker (KKT) optimality conditions (Boyd and Vandenberghe 2004) $\mathbf{\Psi}_{ij}\mathbf{U}_{ij} = 0$, $\mathbf{\Phi}_{ij}\mathbf{V}_{ij} = 0$ and $\mathbf{\Gamma}_{ij}\tilde{\mathbf{U}}_{ij} = 0$, I can get:

$$
(-2\mathbf{Y}\mathbf{V} + 2\mathbf{U}\mathbf{V}^\top\mathbf{V} + \beta\mathbf{L}\mathbf{U} + \beta\mathbf{L}^\top\mathbf{U})_{ij}\cdot\mathbf{U}_{ij} = 0
$$

$$
(-\mathbf{Y}^\top\mathbf{U} + \mathbf{V}\mathbf{U}^\top\mathbf{U} - \alpha\mathbf{M}^\top\tilde{\mathbf{U}} + \alpha\mathbf{V}\tilde{\mathbf{U}}^\top\tilde{\mathbf{U}})_{ij}\cdot\mathbf{V}_{ij} = 0
$$

$$
(-2\alpha\mathbf{M}\mathbf{V} + 2\alpha\tilde{\mathbf{U}}\mathbf{V}^\top\mathbf{V})_{ij}\cdot\tilde{\mathbf{U}}_{ij} = 0
$$

Algorithm 3 Updating Rules for MI^2-HD

1: **Input**:

$\mathbf{Y}_0$: incomplete biomarker dataset.

$\mathbf{M}$: disease-dedicate forum dataset.

2: **Repeat**:

3: **Repeat**:

4: update U_{ik}^{t+1}, V_{jk}^{t+1}, $\tilde{U}_{ij}^{t+1}$ in Eq.(4.6)

5: **Until**: Eq.(4.4) converge

6: set $\mathbf{Y}_{\Omega}^{t'} = (\mathbf{Y}_0)_{\Omega}$

7: set $\xi = (\mathbf{Y}^{t'+1} - \mathbf{Y}^{t'})./\mathbf{Y}^{t'}$

8: **Until**: $\xi < 1E-2$

9: **Output**: $\mathbf{Y}$, $\mathbf{U}$, $\tilde{\mathbf{U}}$, $\mathbf{V}$

Eq. (4.6) leads to the following updating rules for MI^2-HD:

$$\begin{aligned}
\mathbf{U}_{ik} &= \mathbf{U}_{ik}\frac{(2\mathbf{YV} + \beta\mathbf{HW}_e\mathbf{D}_e^{-1}\mathbf{H}^\top\mathbf{U})_{ik}}{(2\mathbf{UV}^\top\mathbf{V} + \beta\mathbf{D}_v\mathbf{U} + \beta\mathbf{D}_v^\top\mathbf{U})_{ik}} \\
\mathbf{V}_{kj} &= \mathbf{V}_{kj}\frac{(2\mathbf{Y}^\top\mathbf{U} + \alpha\mathbf{M}^\top\tilde{\mathbf{U}})_{kj}}{2(\mathbf{VU}^\top\mathbf{U} + \alpha\mathbf{V}\tilde{\mathbf{U}}^\top\tilde{\mathbf{U}})_{kj}} \\
\tilde{\mathbf{U}}_{ij} &= \tilde{\mathbf{U}}_{ij}\frac{(\mathbf{MV})_{ij}}{(\tilde{\mathbf{U}}\mathbf{V}^\top\mathbf{V})_{ij}}
\end{aligned} \tag{4.6}$$

Fix U, $\tilde{\mathbf{U}}$, V, update Y: After the convergence of $\mathbf{U}$, $\tilde{\mathbf{U}}$, $\mathbf{V}$, I set $\mathbf{Y}_{\mathbf{\Omega}} = (\mathbf{Y}_0)_{\mathbf{\Omega}}$ to restore the observed information. Then repeating update $\mathbf{U}$, $\tilde{\mathbf{U}}$, $\mathbf{V}$ until $\mathbf{Y}$ converge.

4.4.2 MI^2-HT UPDATING RULES

Derivation of Eq. (4.5) follows the same procedure as the derivation of Eq. (4.4). Omitted for brevity, the updating rules for MI^2-HT are directly given in Algorithm 4.

Algorithm 4 Updating Rules for MI^2-HT

Input:

$\mathbf{Y}_0$: incomplete biomarker dataset.

$\mathbf{M}$: disease-dedicate forum dataset.

while $\epsilon > Convergence\,Criterion$ **do**

$U_{ik}^{t+1} \leftarrow U_{ik}^{t} \frac{(\mathbf{YVS}^\top)_{ij}}{(\beta \mathbf{L}^h \mathbf{U} + \beta (\mathbf{L}^h)^\top \mathbf{U})_{ij}}$

$V_{jk}^{t+1} \leftarrow V_{jk}^{t} \frac{(\mathbf{Y}^\top \mathbf{US})_{ij}}{(\alpha \mathbf{VS}^\top \mathbf{U}^\top \mathbf{US})_{ij}}$

$\tilde{U}_{ij}^{t+1} \leftarrow \tilde{U}_{ij}^{t} \frac{(\mathbf{M}\mathbf{V}\tilde{\mathbf{S}}^\top)_{ij}}{(\tilde{\mathbf{U}}\tilde{\mathbf{S}}\mathbf{V}^\top \mathbf{V}\tilde{\mathbf{S}}^\top)_{ij}}$

$S_{ik}^{t+1} \leftarrow S_{ik}^{t} \frac{(\mathbf{U}^\top \mathbf{YV})_{ij}}{(\mathbf{V}^\top \mathbf{VS}^\top \mathbf{U}^\top \mathbf{U})_{ij}}$

$\tilde{S_{ik}^{t+1}} \leftarrow \tilde{S}_{ik}^{t} \frac{(\tilde{\mathbf{U}}^\top \mathbf{MV})_{ij}}{(\mathbf{V}^\top \mathbf{V}\tilde{\mathbf{S}}^\top \tilde{\mathbf{U}}^\top \tilde{\mathbf{U}})_{ij}}$

$ObjValue^{t+1} = O(\mathbf{U}^{t+1}, \mathbf{V}^{t+1}, \mathbf{U}^{\tilde{t+1}}, \mathbf{S}^{t+1}, \mathbf{S}^{\tilde{t+1}})$

$\epsilon = ObjValue^{t+1} - ObjValue^{t}$

$t = t + 1$

end while

Output: $\mathbf{U}$, $\mathbf{V}$, $\tilde{\mathbf{U}}$, $\mathbf{S}$, $\tilde{\mathbf{S}}$

4.4.3 CONVERGENCE ANALYSIS

The Algorithm 3 is not jointly convex for all the variables $\mathbf{Y}$, $\mathbf{U}$, $\tilde{\mathbf{U}}$, $\mathbf{V}$, but convex in each of them separately. As shown in Eq. (4.4), when $\mathbf{Y}$ is fixed, the proof regarding the convexity of Eq. (4.4) with respect to variables $\mathbf{U}$, $\tilde{\mathbf{U}}$, $\mathbf{V}$ is analogous to (Cai *et al.* 2008); when $\mathbf{U}$, $\tilde{\mathbf{U}}$, $\mathbf{V}$ are fixed, the optimization problem is equivalent to $\min_{\mathbf{Y}} \|\mathbf{Y} - \mathbf{C}\|_F^2$, s.t. $\mathbf{Y}_{\mathbf{\Omega}} = (\mathbf{Y}_0)_{\mathbf{\Omega}}$, with respect to $\mathbf{Y}$ only. $\mathbf{C}$ is given as constant. To be more specific, the equality constraint can be rewritten as $\|\Lambda \mathbf{Y}^C - \mathbf{C}\|_F^2$, where $\mathbf{Y}^C$ denotes the column-wise concatenation of $\mathbf{Y}$, and $\mathbf{\Lambda}$ is a constant diagonal matrix with the diagonal elements equal to the column-wise concatenation of $\mathbf{\Omega}$. Thus, the local optima are feasible when Algorithm 1 is proved to be convex with respect to

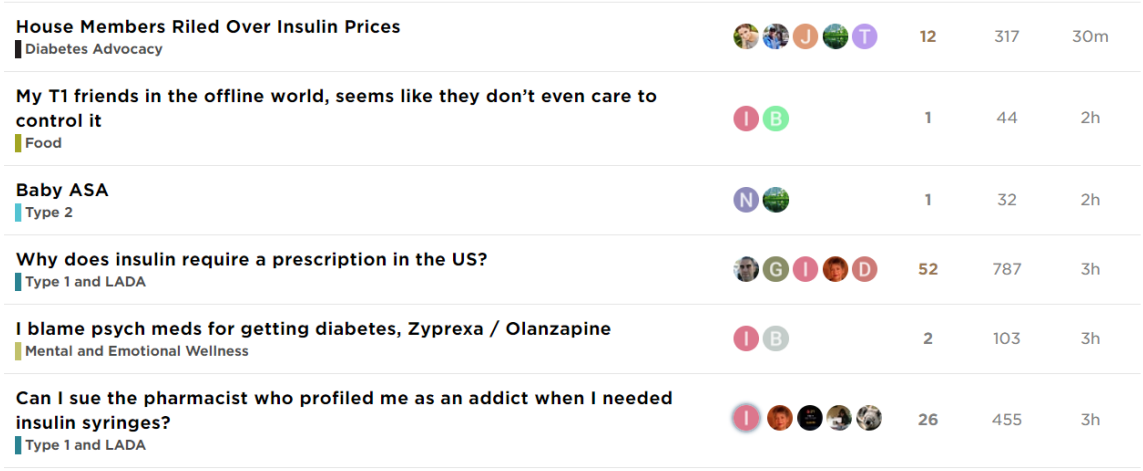


Figure 4.3: TuDiabetes Forum Screen Shot.

variables $\mathbf{Y}$, $\mathbf{U}$, $\tilde{\mathbf{U}}$, and $\mathbf{V}$ individually. The same proof procedure can be easily adapted to proof that Algorithm 4 also achieves the local optimal solution.

4.5 EVALUATION

4.5.1 EXPERIMENTAL SET-UP

There are two real-world data sets and one synthetic data set in our experiments: **Synthetic Data set**: As mentioned in the reference (Hofmann 2003), the Gaussian distribution is adopted to estimate the users' rating for the item when studying the user preferences, in which each community can be identified by a Gaussian distribution generated from the normalized user ratings. In our case, when I generate the synthetic data, I assume that each user can also be identified by a Gaussian distribution, which is generated according to the user's attendance to each post (topic). The factorization factors $\mathbf{U}$, $\mathbf{V}$, and $\tilde{\mathbf{U}}$ are generated based on the multi-variate Gaussian distribution. Each of them contains 200, 450 and 150 examples respectively. The trade-off parameter is selected in grid $[0.3, 3, 30, 300]$ to balance the effectiveness of each regularization term and avoid single term monopolizing the objective function.

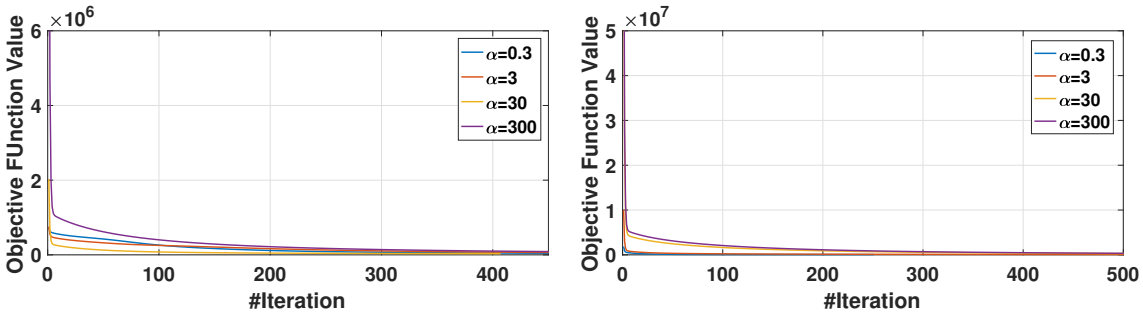
TuDiabetes Data Set: The TuDiabetes online forum consists of a community of people touched by diabetes and the disease-specific discussion about their diabetes condition. The set of discussions usually include Type I diabetes, Type II diabetes, gestational diabetes, diet, and exercise, etc. As the screenshot of the TuDiabetes forum shown in Fig. 4.3, the users tend to form the same groups with interest in a certain topic. In general, the Tudiabetes data set is a collection of 21,286 discussion posts with 294,272 users. The features for each user consist of the TF-IDF (Salton and Yang 1973) feature of his/her posts after the pre-processing steps (verb tense uniform, stop word removal).

OneID: The OneID data set (zhongqi 2015) contains the encrypted user online shopping activities, including the device-cookie pair, searching keywords, auction ID, shop ID and so on. The users' feature is extracted by the Geohash method (Geohashes 2008) from the raw encrypted information that each feature is converted into a vector of the same length. For detailed information, readers are recommended to see the reference.

I use a range of missing ratios ($mRatio$) to partition Y_0 into the observed portion and the missing portion. The missing entries are randomly selected based on the value of $mRatio$ in the grid $[0.3, 0.4, 0.5, 0.6, 0.7, 0.8]$, e.g., $mRatio$=0.4 means 40% entries in Y_0 are manually removed as missing, and replaced with value 0. The removed portion is used as ground truth to evaluate the imputation accuracy.

4.5.2 EXPERIMENTAL RESULTS

I consider the scenario when the data sparsity spreads over different sparsity ratio. The data sparsity ratio alters the imputation accuracy of our methods. In Fig. 4.5, the imputation value accuracy is compared with three other algorithms on synthetic data. The x-axis represents the missing ratio, and the y-axis shows the accuracy of the



(a) Convergence of MI^2-HD. (b) Convergence of MI^2-HT.

Figure 4.4: Convergence analysis with respect to the trade-off parameters. The x and y axes denote the iteration number and the objective function value respectively.

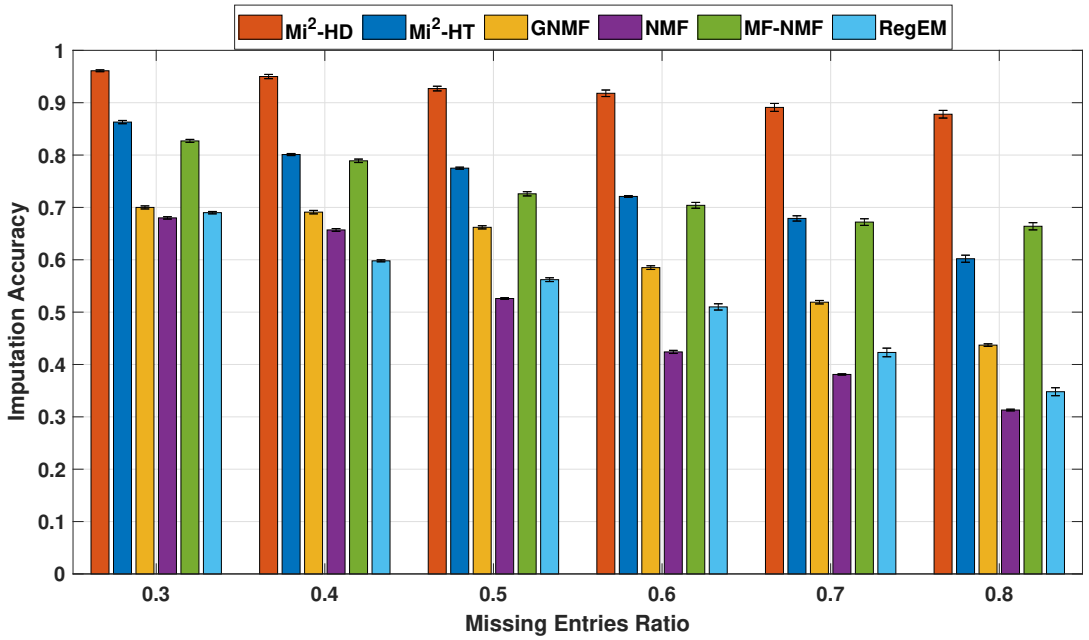


Figure 4.5: Comparison analysis of synthetic data. The first two bars of each bar-group represent the imputation accuracy of MI^2-HD and MI^2-HT. The x-axis represents the missing entries ratio which controls the percentage of missing entries, while the y-axis represents the imputation accuracy with respect to the certain missing ratio.

imputation value. It can be observed that the imputation performance is much more stable when the ratio of missing fraction getting increasing. I take cosine similarity Nguyen and Bai 2010 to measure the imputation accuracy between the imputation result and original value. To be more precise, cosine similarity is a measurement that measures the similarity between two non-zero vectors by calculating their cosine value of the angle in their inner product space. Each result is the average over 30-run results. For each running, $\mathbf{U}$, $\mathbf{V}$, and $\tilde{\mathbf{U}}$ are randomly initialized from multivariate Gaussian distribution. Compared with the other three methods, whose accuracies decline quickly along with the increasing ratio of missing entries, MI^2-HD algorithm decreases relatively slow and shows the highest imputation accuracy.

The 30-run average results on both the TuDiabetes data set and the OneID data set are shown in Fig. 4.6. The first two bars of each bar-group represent the imputation accuracy for our MI^2-HD and MI^2-HT. Overall, with the increasing of missing value fraction, our method shows stable high accuracy with the help of hypergraph structure and the heterogeneous auxiliary information. To be explicit, the experiment results verify the two main advantages of our method:

(1) As shown in Fig. 4.7a, by leveraging the hypergraph structure, the proposed MI^2-HD can improve the missing value imputation performance when compared with the traditional graph-based methods *GNMF*.Compared with the model MI^2-HT, the model MI^2-HD shows higher missing value imputation accuracy on both data sets. The reason is that in model MI^2-HT, the strict orthogonality constraints have been adopted on the factorization factors, i.e., $\mathbf{U}$, $\tilde{\mathbf{U}}$, and $\mathbf{V}$. The model MI^2-HT benefits from the mathematical property of the orthogonality constraint, which reduces the computational complexity dramatically. However, such an orthogonality constraint presents the one-to-one mapping relationship among the users and posts, which ignoring the user-grouping knowledge of the disease-dedicated social networks, even

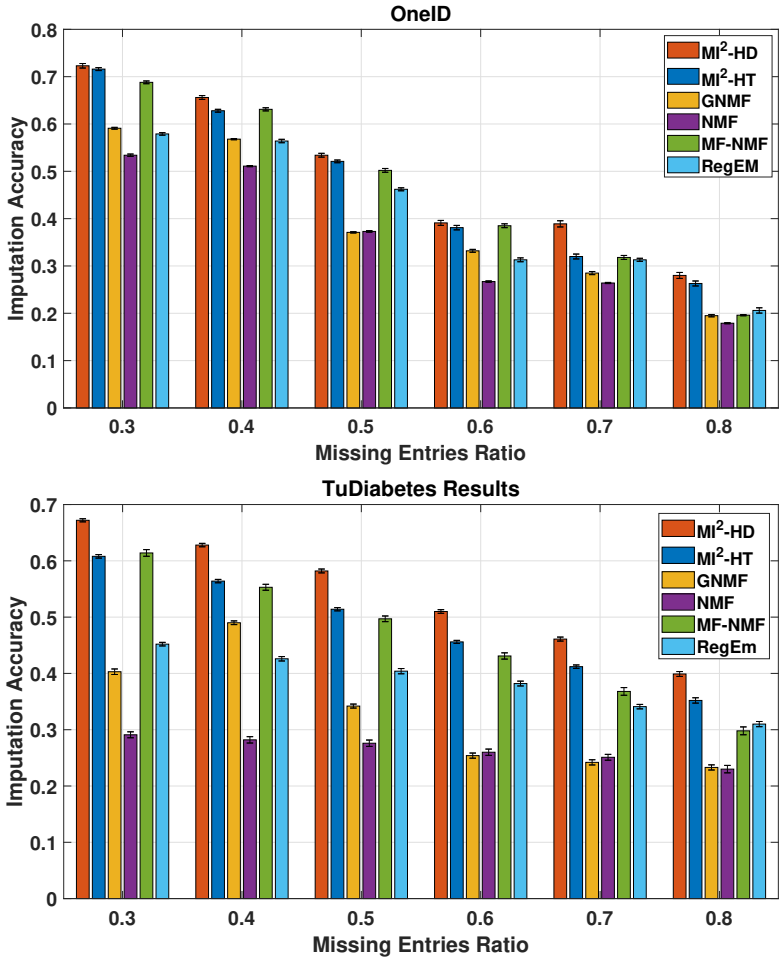
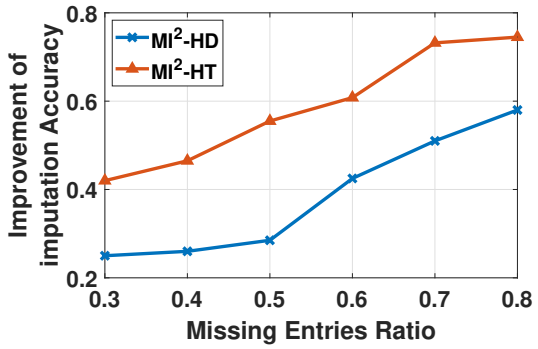


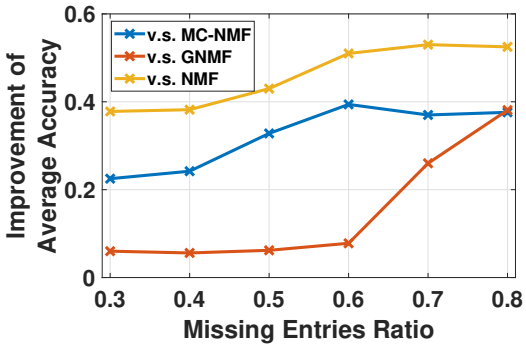
Figure 4.6: Upper: Experimental results on the OneID dataset; Lower: Experimental results on the TuDiabetes dataset. Each numerical value is averaged over 30-run repeated test, then the 30-run variance is shown in the error bar.

though I have addressed the user-grouping knowledge in the Eq. (4.5) by leveraging the hypergraph structure. The proper constraints give rise to better imputation accuracy, which I will attach great importance in our future works.

(2) As shown in Fig. 4.7b, the superiority of our methods is increasing along with the data sparsity growing up. The imputation accuracy and robustness are benefited from utilizing the heterogeneous data by sharing the measurement matrix V between



(a) Hypergraph structure v.s. Pairwise graph structure.



(b) Leveraging heterogeneous auxiliary information v.s. None.

Figure 4.7: Experimental Analysis: (a) Imputation improvement by leveraging hypergraph structure. (b)Imputation improvement by utilizing heterogeneous data.

the original and heterogeneous information.

Chapter 5

MISSING VALUE IMPUTATION WITH DATA MULTI-MODALITY

Multi-modality learning aims to improve model generalization performance by combining information from a variety of data modalities. One frequent strategy is to look for common information that is shared across different learning modalities, while I may also integrate supplemental data to use modality-specific data.

Customers' reviews, as well as the numerical rating score, are supplied for the recommender system. It's not uncommon for some consumers to give a liberal rating to an item, while others are more stringent in their evaluations. The numerical rating value of 3', for example, comes from a forgiving consumer who was dissatisfied with his or her purchasing experience, but the score of 3' is also offered by individuals who have unrealistic expectations for the goods. It is improper to create a recommender in this scenario without removing user bias. Incorporating textual review into the model might be a method to minimize bias. When contrasted to strict consumers with a rating of 3', who may employ fussy terms, tolerant consumers with a rating of 3' would have a different choice of words to describe the item. Due to the data heterogeneity, since each consumer would offer evaluations for items in a different category, another issue known as "Semantic Bias" comes to mind when considering the textual review. For example, the term 'complex' has very different emotional connotations for the product categories 'Computer' and 'Book,' e.g., 'The operating system is complicated for me' vs 'This book has complicated friction,' implying either negative or good feeling. People are unable to build an interpretable missing value imputation technique for the recommender system without taking into account such semantic bias while adding the review into our model.

5.1 IMPUTATION STRATEGY

For this problem with the observed users' contextual review and rating score, the goal is to impute the missing rating score based on his/her long-term rating record together with the most recent short-term review information. Users' rating preference (leniency or strict) is estimated according to his/her similar users when they use the same adjective words and adverb words in their review. There are multiple constraints proposed with respect to three aspects: (1) Maximize the consistency between the imputation results with the observed information ($L_F(\theta)$); (2) each users' rating score is prejudged by his/her contextual review information ($L_R(\theta)$); (3) users who user same adjective words and adverb words in their review are close to each other ($L_U(\theta)$). The overall objective function is formulated as follows:

$$L(\theta) = L_F(\theta) + \alpha L_R(\theta) + \beta L_U(\theta) \tag{5.1}$$

where $L_F(\theta)$, $L_R(\theta)$, and $L_U(\theta)$ correspond to the *imputation consistency*, *cross modality consistency*, and *user similarity*, respectively. The trade-off parameters α and β are non-negative for the purposes of keeping balance of each term, and being convenient for solving the optimization problem. Each term is explained in detail as follows.

Imputation Consistency: In Eq.(5.1), the term $L_F(\theta)$ is designed to require the consistency between the imputation results and the observed information. For the user-item rating score matrix $\mathbf{X} \in R^{m \times n}$, $\mathbf{X}$ is factorized into two low-rank matrices by using non-negative matrix factorization (NMF) (Lee and Seung 2001). The scoring data $\mathbf{X}$ is factorized into two low-rank matrices $\mathbf{U} \in R^{m \times k}$ and $\mathbf{V} \in R^{n \times k}$, where k is the latent grouping number. Matrix $\mathbf{U}$ indicates the row clustering (sample grouping), and matrix $\mathbf{V}$ indicates the column clustering (measurement grouping). The product of these two low-rank matrices $\mathbf{U}(\mathbf{V})^\top$ is treated as the imputation results for original

data $\mathbf{X}$ where the rating score is missing. The impact and extension of the inherent parameter k is studied in (Ding *et al.* 2006). Various type of matrix norm can be used, e.g. $L_{1,2}/L_{2,1}$, L_∞ and etc. I use the F norm for the convenience of calculation.

Cross Modality Consistency: Modality consistency is a significant principle that ensuring the success of missing value imputation in multi-modality learning (Cai *et al.* 2018). The consistency among multiple modalities requires the prediction results for any two modalities should keep consistent as high as possible. Different from the previous methods, where the consistency is usually considered merely among the data in the same modality, I incorporate the contextual information as the supplementary knowledge to explore the certain emotion level of each word and further balance the imputed users' rating score. The users' who are using the same words (adjective and adverb) should be considered similar to each other, and their rating score is also closing to each others'. For each item v, r_avg denotes the average rating score over all the users. The $\mathbf{X}^R \in R^{m\times n}$ is calculated based on the ELMo pre-trained word embedding model (Che *et al.* 2018; Fares *et al.* 2017). Without ambiguity, each item is presented as the average sum of all the adjective and adverbs words that have been observed in its review for each user. $\mathbf{X}^R$ is factorized into low-rank representation $\mathbf{X}^R = \mathbf{U}^R(\mathbf{V}^R)^\top$, that $\mathbf{U}^R$ indicates user clustering-based their rating habit, and $\mathbf{V}^R$ indicates item clustering affected by users' rating habit.

The cross-modality consistency is maintained by obtaining a latent subspace shared by the user rating habit measurement grouping $\mathbf{V}^R$ and item clustering grouping which affected by users' rating habit) $\mathbf{V}$. Based on such subspace learning assumption, I explore the family of subspace learning methods and leverage the Canonical Correlation Analysis (CCA), which aims to maximize the correlation between two

views. The correlation coefficient $\rho_{t,L}$ between two views $\mathbf{V}$ and $\mathbf{V}^{(L)}$ is calculated as:

$$\rho_{t,R} = \frac{\boldsymbol{\omega}^\top \mathbf{V}\mathbf{V}^{R^\top}\boldsymbol{\omega}^R}{\sqrt{(\boldsymbol{\omega}^\top \mathbf{V}\mathbf{V}^\top\boldsymbol{\omega})(\boldsymbol{\omega}^{R^\top}\mathbf{V}^R\mathbf{V}^{R^\top}\boldsymbol{\omega}^R)}} \tag{5.2}$$

since ρ_{ij} is invariant to the scaling of projection vector $\boldsymbol{\omega}$ and $\boldsymbol{\omega}^R$, Eq. (5.3) is equivalent to the following optimization problem:

$$\begin{aligned} &\max_{\boldsymbol{\omega},\boldsymbol{\omega}^R} \boldsymbol{\omega}^\top \mathbf{V}\mathbf{V}^{R^\top}\boldsymbol{\omega}^R \\ &s.t.\ \boldsymbol{\omega}^\top \mathbf{V}\mathbf{V}^\top\boldsymbol{\omega} = 1,\ \boldsymbol{\omega}^{R^\top}\mathbf{V}^R\mathbf{V}^{R^\top}\boldsymbol{\omega}^R = 1 \end{aligned} \tag{5.3}$$

In Eq. (5.1), $L_R(\theta)$ encodes the multi-modality consistency by taking the summation over log reciprocal of the the CCA correlation coefficient ρ_{tL}.

User Similarity: Besides the constraints of cross-modality consistency (which consider the user similarity from the contextual review aspect), the users are also showing their connection by considering their numerical rating score only. The idea it to encode the user grouping information based on graph-structured norm, know as graph regularizer (Cai *et al.* 2011). Following the same instruction in 4.3, $L_U(\theta)$ encodes the construction of nodes (user) and edges (rating-based similarity), where the nodes correspond to the user, and the relationship of the user is revealed by the pairwise similarity according to their rating scores.

5.2 ALGORITHM AND SOLUTION

The efficient multiplicative updating rule for each $\mathbf{U}$ and $\mathbf{V}$ is proposed by decomposing the objective function in Eq.4.4 into:

$$\begin{aligned} \mathcal{O} = &\sum_t^T (Tr(\mathbf{X}\mathbf{X}^\top) - 2Tr(\mathbf{X}\mathbf{V}\mathbf{U}^\top) + Tr(\mathbf{U}\mathbf{V}^\top\mathbf{V}\mathbf{U}^\top)) + \\ &\sum_t^T \alpha_t log\frac{1}{\rho_{t,L}} + \sum_t^T \beta_t Tr(\mathbf{U}^\top\mathbf{L}\mathbf{U}) \end{aligned} \tag{5.4}$$

Algorithm 5 Updating Rules for Multi-Modality Missing Value Strategy

Input:

$\mathbf{X}, \mathbf{L}$

$\mathbf{U}$ and $\mathbf{V}$ are initialized as the NMF factors of $\mathbf{X}$.

$\mathbf{V}^R$ is initialized as the NMF factors of label.

$ObjValue = 10^{-2}$

$\epsilon = 1, Convergence\,Criterion = 10^{-2}$, step= 0

Output:

$\mathbf{U}$, $\mathbf{V}$, $\boldsymbol{\omega}$, $\boldsymbol{\omega}^L$

1: **while** $\epsilon > Convergence\,Criterion$ **do**

2: $\boldsymbol{\omega} \leftarrow \boldsymbol{\omega}$ update by Eq. (5.3)

3: $\boldsymbol{\omega}^R \leftarrow \boldsymbol{\omega}^R$ update by Eq. (5.3)

4: $\mathbf{U}_{i,k} \leftarrow \mathbf{U}_{i,k}$ update in Eq. (5.8)

5: $\mathbf{V}_{j,k} \leftarrow \mathbf{V}_{j,k}$ update in Eq. (5.8)

6: $ObjValue^{step+1} = O(\{\mathbf{U}\}_{t=1}^{T}, \{\mathbf{V}\}_{t=1}^{T}, \boldsymbol{\omega}, \boldsymbol{\omega}^R)$

7: $\epsilon = |\frac{ObjValue^{step+1} - ObjValue^{s}tep}{ObjValue^{s}tep}|$

8: **end while**

where ρ_i equals to:

$$\rho_{t,L} = \boldsymbol{\omega}^\top \mathbf{V}^\top \mathbf{V}^R \boldsymbol{\omega}^R \tag{5.5}$$

Let $\phi_{i,j}$ and $\psi_{k,j}$ be the Lagrange multipliers for the constraints $\mathbf{U}_{i,j} \geq 0$ and $\mathbf{V}_{i,j} \geq 0$ respectively. Following the similar technique used in (Cai *et al.* 2011), the Lagrange function is $\mathscr{L}$ formulated as:

$$\mathscr{L} = \mathcal{O} + \sum_{t}^{T} (Tr(\boldsymbol{\Psi}\mathbf{U}^\top) + Tr(\boldsymbol{\Phi}\mathbf{V}^\top)) \tag{5.6}$$

Table 5.1: Multi-Modality Amazon Customer Rating and Review Data Sets.

ID	Views	|User|	Average Rating	Review
1	**Electronics** & **Video Games**	842	4.55	3541
2	**Patio** & **Tools**	151	4.31	189
3	**Beauty Product** & **Clothing**	275	4.25	408
4	**Art** & **Musical Instruments**	182	4.11	617
5	**Electronics** & **Kindle Store**	756	3.96	1438
6	**Beauty Product** & **Jewelry**	453	4.58	598
7	**Kindle Store** & **Software**	673	3.47	1525

the partial derivatives of $\mathscr{L}$ with respect to $\mathbf{U}$ and $\mathbf{V}$ are:

$$\begin{aligned}\frac{\partial \mathscr{L}}{\partial \mathbf{U}} &= -2\mathbf{X}\mathbf{V} + 2\mathbf{U}\mathbf{V}^\top\mathbf{V} + 2\beta\mathbf{L}\mathbf{U} + \mathbf{\Psi} \\ \frac{\partial \mathscr{L}}{\partial \mathbf{V}} &= -2\mathbf{X}^\top\mathbf{U} + 2\mathbf{V}\mathbf{U}^\top\mathbf{U} + 2\alpha_t\boldsymbol{\omega}^\top\mathbf{V}^\top\mathbf{V}^R\boldsymbol{\omega}^L\mathbf{V}^R\boldsymbol{\omega}^L\boldsymbol{\omega}^\top + \mathbf{\Phi}\end{aligned} \tag{5.7}$$

by setting each partial derivative equals to 0, based on the KKT conditions $\mathbf{U}_{i,j}\boldsymbol{\psi}_{i,j} = 0$ and $\mathbf{V}_{i,j}\boldsymbol{\phi}_{i,j} = 0$, I can get the updating rule for $\mathbf{U}$ and $\mathbf{V}$ respectively as:

$$\begin{aligned}\mathbf{U}_{i,k} &= \mathbf{U}_{k,j}\left(\frac{\mathbf{X}\mathbf{V}}{\mathbf{U}\mathbf{V}^\top\mathbf{V} + \beta_t\mathbf{L}\mathbf{U}}\right)_{i,k} \\ \mathbf{V}_{j,k} &= \left(\frac{\mathbf{X}^\top\mathbf{U}}{\mathbf{V}\mathbf{U}^\top\mathbf{U} + \alpha_t\boldsymbol{\omega}^\top\mathbf{V}^\top\mathbf{V}^R\boldsymbol{\omega}^L\mathbf{V}^R\boldsymbol{\omega}^L\boldsymbol{\omega}^\top}\right)_{j,k}\end{aligned} \tag{5.8}$$

The iterative procedure is described in Algorithm 5 and working as follows. The input of the procedure is the $\mathbf{U}^0$ and $\mathbf{V}^0$, which equal to the NMF factorization factors of user rating score matrix $\mathbf{X}$. The parameter $ObjValue^t$ takes the records of objective function value during the updates, and it is initialized as 10^{-2} to ensure

Table 5.2: Missing Value Imputation Performance Comparison.

Dataset ID	Multi-Modality	Without Modality	GROUSE	IALM	LMaFit	MC-NMF	OR1MP	RMAMR	multiNMF
1	1.891	1.945	1.456	**1.201**	1.854	1.851	1.590	1.344	3.817
2	**1.145**	1.764	1.259	1.415	1.725	1.456	1.464	1.678	2.298
3	1.748	2.231	1.549	**1.198**	2.261	2.041	1.392	1.285	2.301
4	1.456	2.147	**1.135**	1.152	1.546	1.951	1.764	1.783	2.961
5	1.256	1.598	**1.093**	1.413	1.874	2.322	1.270	1.648	2.888
6	1.489	2.315	**1.315**	1.854	2.185	2.485	1.482	1.798	2.253
7	**1.294**	1.474	1.995	**1.185**	1.846	1.749	1.628	1.185	2.403

the compilable of the whole procedure. $\mathbf{V}^R$ is obtained by factorizing the multi-label information $\mathbf{X}^R$ through NMF.

The main computation cost is derived by matrix multiplication. Therefore, omitted the space, the time complexity for Mi-L^2 is $O(mnD)$, that D equals to the maximum clustering number over each view.

5.3 EXPERIMENT

Amazon Multi-Modality Data Set: As shown in Table 5.1, the experiments are all conducted on the Amazon data set (He and McAuley 2016), which includes the reviews (ratings, text, helpfulness votes), product metadata (descriptions, category information, price, brand, and image features), and links (also viewed/also bought graphs) for each online transaction. Each user's review and rating score is collected from two purchase aspects for the purpose of complementing his/her rating habits.

Comparison and Evaluation: As shown in Table 5.2, compared with 6 state-of-the-art methods, including GROUSE (Balzano and Wright 2013), IALM (Lin *et al.* 2010), LMaFit (Wen *et al.* 2012), MC-NMF Xu *et al.* 2012, OR1MP Wang *et al.* 2015b, RMAMR (Ye *et al.* 2015), and (Liu *et al.* 2013). Parameters are initialized as

suggested correspondingly.

Chapter 6

CONCLUSTION

This thesis aims to address the missing value imputation problem when considering the data heterogeneity. Three main topics have been identified as (1) Mining and formulating heterogeneous data; (2) Imputing missing value in multiple scenarios including recommender systems and healthcare-based social media; (3) Imputation with Data Modality. Through the empirical analysis, I have designed the solutions for each scenario and demonstrated the effectiveness of the proposed solutions, compared with state-of-the-art methods. The data heterogeneity has been characterized and formulated in each scenario while addressing the problem of missing value imputation.

For the recommender system scenario, the missing value imputation strategies are classified into two cased (i) offline imputation strategy, and (ii) online imputation strategy. For the offline strategies, the collective matrix completion is adopted under the consideration of multi-view data. The cross-view is decoded in the graph spectral domain and quantifies the matrices' interactive impacts. Experimental comparison with other state-of-the-art methods on ten real-world data sets shows the improved performance of missing value prediction; For the online imputation strategy, the real-time customers' shopping preference is treated as the target value of imputation, and the reinforcement learning-based strategies have been explored to understand the imputation result.

Towards disease-dedicated social networks, I make an effort to formulate patient grouping information based on the hypergraph representation, and leverage additional information such as users' social relationships and clinical data to improve the accuracy of missing value imputation. The proposed iterative algorithms solve the

resulting optimization problems. I also analyze their performance from multiple perspectives. Experimental results show that the missing value imputation performance has been improved by leveraging the auxiliary data from diabetes-dedicated social networks.

The last topic regarding the missing value imputation in the scenario of multi-modality data is explored. The users' bias is revealed in the contrast of their contextual text review and their rating score. The effectiveness of contextual information is addressed to alleviate the data sparsity issue. The missed user rating score is then estimated based on his/her long-term historical rating habit (either leniency or strict) together with his/her short-term contextual review information.

The aforementioned research results for recommender systems have been published and presented in CIKM 2019 and WWW 2021, and the second topic regarding disease-dedicated social networks has been published at IISE 2020. The last topic of missing value imputation with multi-modality data is targeting IJCAI 2022.

6.0.1 FUTURE WORK

Learning from data heterogeneity for missing value imputation is an active study field for multiple applications domains of recommender system, healthcare analysis, natural language understanding, cyber-security analysis and so on. The majority of the publications cited in this thesis are concerned with the usage of recommender systems and social media analysis. Missing value imputation may be used in a variety of domains, including advertisement bidding, 3D picture reconstruction, and so on. State-of-the-art techniques for handling the missing value imputation problem include deep learning models (e.g., generative adversarial network, deep reinforcement learning). Finally, adding more diversified data modalities (e.g., the picture in a recommender system for rating prediction) is required to better explain consumers'

buying intent.

Meanwhile, the ethical issues in healthcare analysis has been widely discussed. 63 percent of adults are uneasy with personal data being used to improve healthcare and are opposed to health care analytics systems taking over functions normally performed by doctors and nurses (McKee 2013). People are questioning what internet privacy actually means. They routinely disclose extensive information about all parts of their life on social media, including embarrassing anecdotes and even incriminating images. Privacy, accountability, and data justice require a great deal of care for the research purpose of any works. The interpretable models can be investigated to evaluate the data fairness and the GAN related models can be further utilized to generate the simulation data for the purpose of privacy protection.

Acuna, E. and C. Rodriguez, "The treatment of missing values and its effect on classifier accuracy", in "Classification, clustering, and data mining applications", pp. 639–647 (Springer, 2004).

Agrawal, S. and N. Goyal, "Analysis of thompson sampling for the multi-armed bandit problem", in "Conference on learning theory", (2012).

Aittokallio, T., "Dealing with missing values in large-scale studies: microarray data imputation and beyond", Briefings in bioinformatics **11**, 2, 253–264 (2010).

Akaho, S., "A kernel method for canonical correlation analysis", arXiv preprint cs/0609071 (2006).

An, L., P. Zhang, E. Adeli, Y. Wang, G. Ma, F. Shi, D. S. Lalush, W. Lin and D. Shen, "Multi-level canonical correlation analysis for standard-dose pet image estimation", IEEE Transactions on Image Processing **25**, 7, 3303–3315 (2016).

Auer, P., N. Cesa-Bianchi and P. Fischer, "Finite-time analysis of the multiarmed bandit problem", Machine Learning (2002).

Auer, P., N. Cesa-Bianchi, Y. Freund and R. E. Schapire, "The nonstochastic multi-armed bandit problem", (2003).

Bai, T., J.-Y. Nie, W. X. Zhao, Y. Zhu, P. Du and J.-R. Wen, "An attribute-aware neural attentive model for next basket recommendation", in "The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval", SIGIR '18 (2018).

Balzano, L. and S. J. Wright, "On grouse and incremental svd", in "Computational Advances in Multi-Sensor Adaptive Processing, 2013 IEEE 5th International Workshop on", pp. 1–4 (IEEE, 2013).

Baraldi, A. N. and C. K. Enders, "An introduction to modern missing data analyses", Journal of school psychology **48**, 1, 5–37 (2010).

Bickel, S. and T. Scheffer, "Multi-view clustering.", in "ICDM", vol. 4, pp. 19–26 (Citeseer, 2004).

Blei, D. M., A. Y. Ng and M. I. Jordan, "Latent dirichlet allocation", (2003).

Blum, A. and T. Mitchell, "Combining labeled and unlabeled data with co-training", in "Proceedings of the eleventh annual conference on Computational learning theory", pp. 92–100 (1998).

Boyd, S. and L. Vandenberghe, *Convex optimization* (Cambridge university press, 2004).

Burges, C. J., R. Ragno and Q. V. Le, "Learning to rank with nonsmooth cost functions", in "Advances in Neural Information Processing Systems", (2007).

Buuren, S. v. and K. Groothuis-Oudshoorn, "mice: Multivariate imputation by chained equations in r", Journal of statistical software pp. 1–68 (2010).

Cai, D., X. He, J. Han and T. S. Huang, "Graph regularized nonnegative matrix factorization for data representation", IEEE Transactions on Pattern Analysis and Machine Intelligence **33**, 8, 1548–1560 (2011).

Cai, D., Q. Mei, J. Han and C. Zhai, "Modeling hidden topics on document manifold", in "Proceeding of the 17th ACM conference on Information and knowledge management (CIKM'08)", pp. 911–920 (2008).

Cai, L., Z. Wang, H. Gao, D. Shen and S. Ji, "Deep adversarial learning for multi-modality missing data completion", in "Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining", pp. 1158–1166 (2018).

Carr, J. C., W. R. Fright and R. K. Beatson, "Surface interpolation with radial basis functions for medical imaging", IEEE transactions on medical imaging **16**, 1, 96–107 (1997).

Chapelle, O. and L. Li, "An empirical evaluation of thompson sampling", in "Advances in Neural Information Processing Systems 24", (2011).

Che, W., Y. Liu, Y. Wang, B. Zheng and T. Liu, "Towards better UD parsing: Deep contextualized word embeddings, ensemble, and treebank concatenation", in "Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies", pp. 55–64 (Association for Computational Linguistics, Brussels, Belgium, 2018), URL http://www.aclweb.org/anthology/K18-2005.

Crawshaw, M., "Multi-task learning with deep neural networks: A survey", arXiv preprint arXiv:2009.09796 (2020).

De Leeuw, E. D., "Reducing missing data in surveys: An overview of methods", Quality and Quantity **35**, 2, 147–160 (2001).

Defferrard, M., X. Bresson and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering", in "Advances in Neural Information Processing Systems", pp. 3844–3852 (2016).

Ding, C., T. Li, W. Peng and H. Park, "Orthogonal nonnegative matrix t-factorizations for clustering", in "Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining", pp. 126–135 (ACM, 2006).

Ding, C. H., T. Li and M. I. Jordan, "Convex and semi-nonnegative matrix factorizations", IEEE transactions on pattern analysis and machine intelligence **32**, 1, 45–55 (2010).

Donders, A. R. T., G. J. Van Der Heijden, T. Stijnen and K. G. Moons, "A gentle introduction to imputation of missing values", Journal of clinical epidemiology **59**, 10, 1087–1091 (2006).

Dong, H., P. Neekhara, C. Wu and Y. Guo, "Unsupervised image-to-image translation with generative adversarial networks", arXiv preprint arXiv:1701.02676 (2017).

Dong, M., B. Zheng, N. Quoc Viet Hung, H. Su and G. Li, "Multiple rumor source detection with graph convolutional networks", in "Proceedings of the 28th ACM International Conference on Information and Knowledge Management", pp. 569–578 (2019).

Enders, C. K., *Applied missing data analysis* (Guilford Press, 2010).

Evgeniou, A. and M. Pontil, "Multi-task feature learning", Advances in neural information processing systems **19**, 41 (2007).

Fares, M., A. Kutuzov, S. Oepen and E. Velldal, "Word vectors, reuse, and replicability: Towards a community repository of large-text resources", in "Proceedings of the 21st Nordic Conference on Computational Linguistics", pp. 271–276 (Association for Computational Linguistics, Gothenburg, Sweden, 2017), URL http://www.aclweb.org/anthology/W17-0237.

Farquhar, J., D. Hardoon, H. Meng, J. S. Shawe-Taylor and S. Szedmak, "Two view learning: Svm-2k, theory and practice", in "Advances in neural information processing systems", pp. 355–362 (2006).

Fox, G. A., S. Negrete-Yankelevich and V. J. Sosa, *Ecological statistics: contemporary theory and application* (Oxford University Press, USA, 2015).

Gao, Y. and G. Church, "Improving molecular cancer class discovery through sparse non-negative matrix factorization", Bioinformatics **21**, 21, 3970–3975 (2005).

García-Laencina, P. J., J.-L. Sancho-Gómez and A. R. Figueiras-Vidal, "Pattern classification with missing data: a review", Neural Computing and Applications **19**, 2, 263–282 (2010).

Geohashes, N., "Geohash", http://geohash.org/site/tips.html (2008).

Golub, G. H. and C. F. Van Loan, *Matrix computations*, vol. 3 (JHU Press, 2012).

Greenland, S. and W. D. Finkle, "A critical look at methods for handling missing covariates in epidemiologic regression analyses", American journal of epidemiology **142**, 12, 1255–1264 (1995).

Gu, Q., C. Ding and J. Han, "On trivial solution and scale transfer problems in graph regularized nmf", in "IJCAI Proceedings-International Joint Conference on Artificial Intelligence", vol. 22, p. 1288 (2011).

Guo, L., H. Yin, Q. Wang, T. Chen, A. Zhou and N. Quoc Viet Hung, "Streaming session-based recommendation", in "Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining", KDD '19 (2019).

Guo, R., X. Zhao, A. Henderson, L. Hong and H. Liu, "Debiasing grid-based product search in e-commerce", in "Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining", (2020).

Haldar, M., P. Ramanathan, T. Sax, M. Abdool, L. Zhang, A. Mansawala, S. Yang, B. Turnbull and J. Liao, "Improving deep learning for airbnb search", KDD '20 (2020).

Hammond, D. K., P. Vandergheynst and R. Gribonval, "Wavelets on graphs via spectral graph theory", Applied and Computational Harmonic Analysis **30**, 2, 129–150 (2011).

Harel, O. and X.-H. Zhou, "Multiple imputation: review of theory, implementation and software", Statistics in medicine **26**, 16, 3057–3077 (2007).

He, J. and R. Lawrence, "A graphbased framework for multi-task multi-view learning", in "ICML", (2011).

He, R. and J. McAuley, "Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering", in "proceedings of the 25th international conference on world wide web", pp. 507–517 (2016).

Hidasi, B., A. Karatzoglou, L. Baltrunas and D. Tikk, "Session-based recommendations with recurrent neural networks", (2015).

Hofmann, T., "Collaborative filtering via gaussian probabilistic latent semantic analysis", in "Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval", pp. 259–266 (ACM, 2003).

Hotelling, H., "Relations between two sets of variates", in "Breakthroughs in statistics", pp. 162–190 (Springer, 1992).

Hu, Y., Q. Da, A. Zeng, Y. Yu and Y. Xu, "Reinforcement learning to rank in e-commerce search engine: Formalization, analysis, and application", in "Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining", KDD '18 (2018).

Huang, J.-T., A. Sharma, S. Sun, L. Xia, D. Zhang, P. Pronin, J. Padmanabhan, G. Ottaviano and L. Yang, "Embedding-based retrieval in facebook search", in "Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining", KDD '20 (2020).

Jaderberg, M., V. Mnih, W. M. Czarnecki, T. Schaul, J. Z. Leibo, D. Silver and K. Kavukcuoglu, "Reinforcement learning with unsupervised auxiliary tasks", arXiv preprint arXiv:1611.05397 (2016).

Janssen, K. J., A. R. T. Donders, F. E. Harrell Jr, Y. Vergouwe, Q. Chen, D. E. Grobbee and K. G. Moons, "Missing covariate data in medical research: to impute is better than to ignore", Journal of clinical epidemiology **63**, 7, 721–727 (2010).

Jolliffe, I., *Principal component analysis* (Wiley Online Library, 2002).

Kakade, S. M. and D. P. Foster, "Multi-view regression via canonical correlation analysis", in "International Conference on Computational Learning Theory", pp. 82–96 (Springer, 2007).

Kipf, T. N. and M. Welling, "Semi-supervised classification with graph convolutional networks", arXiv preprint arXiv:1609.02907 (2016).

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

Lange, T. and J. M. Buhmann, "Fusion of similarity data in clustering", in "Advances in neural information processing systems", pp. 723–730 (2006).

Lee, D. D. and H. S. Seung, "Algorithms for non-negative matrix factorization", in "Advances in neural information processing systems", pp. 556–562 (2001).

Li, R., W. Zhang, H.-I. Suk, L. Wang, J. Li, D. Shen and S. Ji, "Deep learning based imaging data completion for improved brain disease diagnosis", in "International Conference on Medical Image Computing and Computer-Assisted Intervention", pp. 305–312 (Springer, 2014).

Li, S. and P. Kar, "Context-aware bandits", (2017).

Li, S., B. Wang, S. Zhang and W. Chen, "Contextual combinatorial cascading bandits", in "Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48", ICML'16 (JMLR.org, 2016).

Li, Z., H. Zhao, Q. Liu, Z. Huang, T. Mei and E. 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 and Data Mining", KDD '18 (2018).

Lin, Z., M. Chen and Y. Ma, "The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices", arXiv preprint arXiv:1009.5055 (2010).

Little, R. J. and D. B. Rubin, *Statistical analysis with missing data*, vol. 793 (John Wiley & Sons, 2019).

Liu, J., C. Wang, J. Gao and J. Han, "Multi-view clustering via joint nonnegative matrix factorization", in "Proceedings of the 2013 International Conference on Data Mining", pp. 252–260 (SIAM, 2013).

Liu, Q., Y. Zeng, R. Mokhosi and H. Zhang, "Stamp: Short-term attention/memory priority model for session-based recommendation", in "Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining", KDD '18 (2018).

Liu, S., E. Johns and A. J. Davison, "End-to-end multi-task learning with attention", in "Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition", pp. 1871–1880 (2019a).

Liu, X., J. He, S. Duddy and L. O'Sullivan, "Convolution-consistent collective matrix completion", in "Proceedings of the 28th ACM International Conference on Information and Knowledge Management", pp. 2209–2212 (ACM, 2019b).

Loyola, P., C. Liu and Y. Hirate, "Modeling user session and intent with an attention-based encoder-decoder architecture", in "Proceedings of the Eleventh ACM Conference on Recommender Systems", RecSys '17 (2017).

Luengo, J., S. García and F. Herrera, "On the choice of the best imputation methods for missing values considering three groups of classification methods", Knowledge and information systems **32**, 1, 77–108 (2012).

Mathieu, M., C. Couprie and Y. LeCun, "Deep multi-scale video prediction beyond mean square error", arXiv preprint arXiv:1511.05440 (2015).

McAuley, J. and J. Leskovec, "Hidden factors and hidden topics: understanding rating dimensions with review text", in "Proceedings of the 7th ACM conference on Recommender systems", pp. 165–172 (ACM, 2013).

McKee, R., "Ethical issues in using social media for health and health care research", Health Policy **110**, 2-3, 298–301 (2013).

Musil, C. M., C. B. Warner, P. K. Yobas and S. L. Jones, "A comparison of imputation techniques for handling missing data", Western Journal of Nursing Research **24**, 7, 815–829 (2002).

Ngo, T. and Y. Saad, "Scaled gradients on grassmann manifolds for matrix completion", in "Advances in Neural Information Processing Systems", (2012).

Nguyen, H. V. and L. Bai, "Cosine similarity metric learning for face verification", in "Asian conference on computer vision", pp. 709–720 (Springer, 2010).

Nigam, P., Y. Song, V. Mohan, V. Lakshman, W. A. Ding, A. Shingavi, C. H. Teo, H. Gu and B. Yin, "Semantic product search", in "Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining", KDD '19 (2019).

Osman, M. S., A. M. Abu-Mahfouz and P. R. Page, "A survey on data imputation techniques: Water distribution system as a use case", IEEE Access **6**, 63279–63291 (2018).

Pedersen, A. B., E. M. Mikkelsen, D. Cronin-Fenton, N. R. Kristensen, T. M. Pham, L. Pedersen and I. Petersen, "Missing data and multiple imputation in clinical epidemiological research", Clinical epidemiology **9**, 157 (2017).

Pereira, R. C., M. S. Santos, P. P. Rodrigues and P. H. Abreu, "Reviewing autoencoders for missing data imputation: Technical trends, applications and outcomes", Journal of Artificial Intelligence Research **69**, 1255–1285 (2020).

Pobrotyn, P., T. Bartczak, M. Synowiec, R. Białobrzeski and J. Bojar, "Context-aware learning to rank with self-attention", (2020).

Qiu, R., J. Li, Z. Huang and H. YIn, "Rethinking the item order in session-based recommendation with graph neural networks", in "Proceedings of the 28th ACM International Conference on Information and Knowledge Management", CIKM '19 (2019).

Raghunathan, T. E., J. M. Lepkowski, J. Van Hoewyk and P. Solenberger, "A multivariate technique for multiply imputing missing values using a sequence of regression models", Survey methodology **27**, 1, 85–96 (2001).

Raymond, M. R. and D. M. Roberts, "A comparison of methods for treating incomplete data in selection research", Educational and Psychological Measurement **47**, 1, 13–26 (1987).

Salton, G. and C.-S. Yang, "On the specification of term values in automatic indexing", Journal of documentation **29**, 4, 351–372 (1973).

Sanz-Cruzado, J., P. Castells and E. López, "A simple multi-armed nearest-neighbor bandit for interactive recommendation", RecSys '19 (2019).

Sobral, A. and E. Zahzah, "Matrix and tensor completion algorithms for background model initialization: A comparative evaluation", Pattern Recognition Letters (2016).

Strike, K., K. El Emam and N. Madhavji, "Software cost estimation with incomplete data", IEEE Transactions on Software Engineering **27**, 10, 890–908 (2001).

Sutton, R. S. and A. G. Barto, *Reinforcement learning: An introduction* (MIT press, 2018).

Torng, W. and R. B. Altman, "Graph convolutional neural networks for predicting drug-target interactions", Journal of Chemical Information and Modeling **59**, 10, 4131–4149 (2019).

Tsikriktsis, N., "A review of techniques for treating missing data in om survey research", Journal of operations management **24**, 1, 53–62 (2005).

Wang, H., F. Nie, H. Huang and F. Makedon, "Fast nonnegative matrix tri-factorization for large-scale data co-clustering", in "IJCAI Proceedings-International Joint Conference on Artificial Intelligence", vol. 22, p. 1553 (2011).

Wang, Q., C. Zeng, W. Zhou, T. Li, S. S. Iyengar, L. Shwartz and G. Y. Grabarnik, "Online interactive collaborative filtering using multi-armed bandit with dependent arms", IEEE Transactions on Knowledge and Data Engineering (2019).

Wang, Y., R. Chen, J. Ghosh, J. C. Denny, A. Kho, Y. Chen, B. A. Malin and J. Sun, "Rubik: Knowledge guided tensor factorization and completion for health data analytics", in "Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining", pp. 1265–1274 (ACM, 2015a).

Wang, Y., G. Ma, L. An, F. Shi, P. Zhang, D. S. Lalush, X. Wu, Y. Pu, J. Zhou and D. Shen, "Semisupervised tripled dictionary learning for standard-dose pet image prediction using low-dose pet and multimodal mri", IEEE Transactions on Biomedical Engineering **64**, 3, 569–579 (2016).

Wang, Y., L. Wang, Y. Li, D. He, W. Chen and T.-Y. Liu, "A theoretical analysis of ndcg ranking measures", in "Proceedings of the 26th annual conference on learning theory (COLT 2013)", vol. 8, p. 6 (2013).

Wang, Y.-X. and Y.-J. Zhang, "Nonnegative matrix factorization: A comprehensive review", IEEE Transactions on Knowledge and Data Engineering **25**, 6, 1336–1353 (2013).

Wang, Z., M.-J. Lai, Z. Lu, W. Fan, H. Davulcu and J. Ye, "Orthogonal rank-one matrix pursuit for low rank matrix completion", SIAM Journal on Scientific Computing **37**, 1, A488–A514 (2015b).

Watson, H. J., "Update tutorial: Big data analytics: Concepts, technology, and applications", Communications of the Association for Information Systems **44**, 1, 21 (2019).

Wells, B. J., K. M. Chagin, A. S. Nowacki and M. W. Kattan, "Strategies for handling missing data in electronic health record derived data", Egems **1**, 3 (2013).

Wen, Z., W. Yin and Y. Zhang, "Solving a low-rank factorization model for matrix completion by a nonlinear successive over-relaxation algorithm", Mathematical Programming Computation **4**, 4, 333–361 (2012).

Wu, L., D. Hu, L. Hong and H. Liu, "Turning clicks into purchases: Revenue optimization for product search in e-commerce", in "The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval", (2018a).

Wu, L., P. Sun, R. Hong, Y. Fu, X. Wang and M. Wang, "Socialgcn: An efficient graph convolutional network based model for social recommendation", arXiv preprint arXiv:1811.02815 (2018b).

Wu, Q., C. J. Burges, K. M. Svore and J. Gao, "Adapting boosting for information retrieval measures", Information Retrieval (2010).

Xu, T., H. Zhang, X. Huang, S. Zhang and D. N. Metaxas, "Multimodal deep learning for cervical dysplasia diagnosis", in "International conference on medical image computing and computer-assisted intervention", pp. 115–123 (Springer, 2016).

Xu, Y., X. Liu, Y. Shen, J. Liu and J. Gao, "Multi-task learning with sample re-weighting for machine reading comprehension", arXiv preprint arXiv:1809.06963 (2018).

Xu, Y., W. Yin, Z. Wen and Y. Zhang, "An alternating direction algorithm for matrix completion with nonnegative factors", Frontiers of Mathematics in China **7**, 2, 365–384 (2012).

Yan, Y., Z. Liu, M. Zhao, W. Guo, W. P. Yan and Y. Bao, "A practical deep online ranking system in e-commerce recommendation", in "Joint European Conference on Machine Learning and Knowledge Discovery in Databases", pp. 186–201 (Springer, 2018).

Yang, P. and W. Gao, "Information-theoretic multi-view domain adaptation: A theoretical and empirical study", Journal of Artificial Intelligence Research **49**, 501–525 (2014).

Yang, P., Q. Tan and J. He, "Complex heterogeneity learning: A theoretical and empirical study", Pattern Recognition **107**, 107519 (2020).

Yang, Y. and H. Wang, "Multi-view clustering: A survey", Big Data Mining and Analytics **1**, 2, 83–107 (2018).

Ye, X., J. Yang, X. Sun, K. Li, C. Hou and Y. Wang, "Foreground–background separation from video clips via motion-assisted matrix restoration", IEEE Transactions on Circuits and Systems for Video Technology **25**, 11, 1721–1734 (2015).

Ying, R., R. He, K. Chen, P. Eksombatchai, W. L. Hamilton and J. Leskovec, "Graph convolutional neural networks for web-scale recommender systems", in "Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining", pp. 974–983 (2018).

You, J., B. Liu, Z. Ying, V. Pande and J. Leskovec, "Graph convolutional policy network for goal-directed molecular graph generation", in "Advances in neural information processing systems", pp. 6410–6421 (2018).

Young, W., G. Weckman and W. Holland, "A survey of methodologies for the treatment of missing values within datasets: Limitations and benefits", Theoretical Issues in Ergonomics Science **12**, 1, 15–43 (2011).

Yu, F., Q. Liu, S. Wu, L. Wang and T. Tan, "A dynamic recurrent model for next basket recommendation", in "Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval", SIGIR '16 (2016).

Zhang, J. and J. Huan, "Inductive multi-task learning with multiple view data", in "Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining", pp. 543–551 (2012).

Zhang, S. and D. Metaxas, "Large-scale medical image analytics: Recent methodologies, applications and future directions", (2016).

Zhang, S., H. Tong, J. Xu and R. Maciejewski, "Graph convolutional networks: Algorithms, applications and open challenges", in "International Conference on Computational Social Networks", pp. 79–91 (Springer, 2018).

Zhang, S., H. Tong, J. Xu and R. Maciejewski, "Graph convolutional networks: a comprehensive review", Computational Social Networks **6**, 1, 11 (2019a).

Zhang, S., L. Yao, A. Sun and Y. Tay, "Deep learning based recommender system: A survey and new perspectives", ACM Comput. Surv. (2019b).

Zhang, Y., M. Chen, D. Huang, D. Wu and Y. Li, "idoctor: Personalized and professionalized medical recommendations based on hybrid matrix factorization", Future Generation Computer Systems **66**, 30–35 (2017).

Zhang, Y., Y. Yang, T. Li and H. Fujita, "A multitask multiview clustering algorithm in heterogeneous situations based on lle and le", Knowledge-Based Systems **163**, 776–786 (2019c).

Zhang, Y., D.-Y. Yeung and Q. Xu, "Probabilistic multi-task feature selection", Advances in neural information processing systems **23**, 2559–2567 (2010).

Zhao, X., R. Louca, D. Hu and L. Hong, "The difference between a click and a cart-add: Learning interaction-specific embeddings", in "Companion Proceedings of the Web Conference 2020", WWW '20 (2020a).

Zhao, Y., Y.-H. Zhou, M. Ou, H. Xu and N. Li, "Maximizing cumulative user engagement in sequential recommendation: An online optimization perspective", in "Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining", KDD '20 (2020b).

zhongqi, L., "Rec-Tmall", //tianchi.aliyun.com/datalab/dataSet.htm?id=2 (2015).

Zhu, S., Y. Wang and Y. Wu, "Health care fraud detection using nonnegative matrix factorization", in "2011 6th International Conference on Computer Science & Education (ICCSE)", pp. 499–503 (IEEE, 2011).