COMPUTATIONAL DESIGN SCIENCE RESEARCH TO SOLVE REAL-WORLD PROBLEMS WITH HIGH IMPACT

by

Buomsoo Kim

A Dissertation Submitted to the Faculty of the

DEPARTMENT OF MANAGEMENT INFORMATION SYSTEMS

In Partial Fulfillment of the Requirements

For the Degree of

DOCTOR OF PHILOSOPHY

In the Graduate College

THE UNIVERSITY OF ARIZONA

2023

ACKNOWLEDGEMENTS

First and foremost, I would like to thank my dissertation committee members, Drs. Sudha Ram, Yong Ge, and Wei Chen, for their encouragement, inspiration, and guidance. Throughout my time as a Ph.D. student at the University of Arizona, I am exceptionally grateful to my advisor, Dr. Sudha Ram, for her consistent advice and support. I consider myself privileged to have such a knowledgeable and compassionate mentor. I also would like to express my gratitude to my family members and relatives, who supported me during the long journey as a graduate student.

I am indebted to the University of Arizona and all my instructors and peers for the invaluable learning experience I have had during the last five years. I am thankful to my colleagues and faculty in the MIS department, my research collaborators, and other friends for their critical feedback and constructive input.

TABLE OF CONTENTS

TABLE OF CONTENTS 3
LIST OF FIGURES 5
LIST OF TABLES 6
ABSTRACT 7
1. INTRODUCTION 8
2. ESSAY 1: A NOVEL METHOD FOR INTERPRETABLE MACHINE LEARNING USING ROBUST LOCAL EXPLANATIONS 13
2.1. Introduction 13
2.2. Related Work 14
2.3. Proposed Method 19
2.3.1. Overview 19
2.3.2. Synthetic Data Generation 21
2.3.3. Local Model Fitting 25
2.4. Analysis 29
2.4.1. Data 29
2.4.2. Evaluation of Machine Learning Prediction Models 31
2.4.3. Evaluation of Local Explanation Models 32
2.5. Discussion 37
2.5.1. Practical Implications 37
2.5.2. Contributions to the IS Knowledge Base 38
2.5.3. Limitations and Future Work 39
2.6. Conclusion 40
3. ESSAY 2: FAIRNESS-AWARE GRAPH REPRESENTATION LEARNING FOR MULTIPLE PATIENT OUTCOME PREDICTION IN INTENSIVE CARE UNITS 42
3.1. Introduction 42
3.2. Related Work 47
3.2.1. Patient Outcome Prediction in Critical Care 47
3.2.2. Bias and Fairness in AI-Assisted Decision-Making 51
3.3. Methodology 53
3.3.1. Graph Construction and Representation Learning 55
3.3.2. Fair2Vec: Representing Patients with POA Conditions 56
3.3.3. Prediction of PA Conditions 61
3.4. Experimental Results 61
3.4.1. Data 61
3.4.2. Graph Representation Learning and Link Prediction 63
3.4.3. Evaluation 64
3.4.4. Fairness of Predictions 71
3.5. Discussions and Conclusion 72
3.5.1. Research Contributions 72
3.5.2. Practical Implications 74
3.5.3. Limitations and Future Work 74

4. ESSAY 3: GRAPH REPRESENTATION LEARNING FOR RECIPROCAL RECOMMENDATION IN ONLINE MATCHING PLATFORMS: A SOCIAL RELATIONS MODEL APPROACH 76
4.1. Introduction 76
4.2. Related Literature 80
4.2.1. Theoretical Background 80
4.2.2. Reciprocal Recommender Systems 84
4.3. Proposed Method 85
4.3.1. Overview 85
4.3.2. Matrix Factorization-Based Rating Prediction Model 88
4.3.3. Graph Neural Network Architecture for Matching Preference Learning 89
4.3.4. Link Prediction and Individual Dating Suggestions 92
4.4. Experimental Results 95
4.4.1. Background 95
4.4.2. Evaluation 96
4.4.3. Data 98
4.4.4. Offline Experimental Results 100
4.4.5. Ablation Study 103
4.4.6. Training Efficiency Analysis 105
4.4.7. Online Field Test Results 107
4.5. Discussions 107
4.5.1. Managerial Implications 107
4.5.2. Research Contributions 108
4.5.3. Limitations and Future Work *109*
4.6. Conclusion 110
5. CONCLUSIONS 111
6. APPENDIX 112
6.1. Hyperparameter Tuning Results for Machine Learning Prediction Models (Essay 1) 112
6.2. Hyperparameter Tuning Results for Different Local Explanation Models (Essay 1) . 113
6.3. Algorithms for SIF and SF Methods (Essay 2) 114
6.4. Additional Details of the Data Used for Analysis (Essay 2) 115
6.5. Details of Hyperparameter Settings for Evaluation (Essay 2) 116
6.6. List of Common and Uncommon Medical Conditions in Data (Essay 2) 117
6.7. Training Details for MatchSAGE (Essay 3) 118
6.8. Hyperparameter Settings for Different Matching Prediction Models (Essay 3) 119
6.9. Hit Ratio and NDCG for Different Values of *k* (Essay 3) 121
7. REFERENCES 123

LIST OF FIGURES

Figure 2.1. Taxonomy of XAI methods 16
Figure 2.2. Illustration of local explanation models faithful to a black-box model 20
Figure 2.3. Examples demonstrating drawbacks of existing local explanation methods 22
Figure 2.4. Synthetic data generation process of ROLEX 24
Figure 2.5. Local explanations for two hypothetical instances 26
Figure 2.6. Decision procedure for training local explanation models 28
Figure 2.7. Comparison of instances with low and high linear separability scores 34
Figure 2.8. Fidelity-complexity trade-off 36
Figure 3.1. Skewed distribution in critical care data (Pollard et al. 2018) 44
Figure 3.2. Design architecture of the proposed framework 45
Figure 3.3. Relations between four major medical conditions in ICU 50
Figure 3.4. Training procedure of the proposed framework 55
Figure 3.5. Summary of evaluation results for 30 most frequent conditions 67
Figure 3.6. Summary of evaluation results for 30 least frequent conditions 67
Figure 3.7. Summary of evaluation results for all recorded conditions 69
Figure 3.8. Summary of evaluation results for the ranking task 70
Figure 3.9. Summary of evaluation results (fairness) 72
Figure 4.1. Comparison of user-item interactions in conventional and reciprocal recommender systems 76
Figure 4.2. Comparison of rating distribution between accepted and rejected suggestions 77
Figure 4.3. Multi-stage decision-making process of users in online dating services (Adapted from Bruch et al. 2016) 88
Figure 4.4. DNN architecture for matching probability prediction 94
Figure 4.5. Screenshots from matching suggestion pages in the app 98
Figure 4.6. Offline evaluation results (classification) 102
Figure 4.7. Offline evaluation results (ranking) 102
Figure 4.8. Training time analysis results 106
Figure 6.1. Hit Ratio with Different k 121
Figure 6.2. NDCG with Different k 122

LIST OF TABLES

Table 1.1. Summary of the three essays in the dissertation....11
Table 2.1. Description of attributes in the fragility fracture dataset....30
Table 2.2. Evaluation results of machine learning algorithms for prediction tasks....31
Table 2.3. Number of edge cases in different datasets/algorithms....32
Table 2.4. Evaluation results of local explanation models....33
Table 2.5. Ablation study results....36
Table 3.1. Comparison of three different techniques in Fair2Vec....59
Table 4.1. Comparison between extant RRSs and MatchSAGE....79
Table 4.2. Summary of existing work in RRS methodology....85
Table 4.3. Summary of the proposed MatchSAGE framework....87
Table 4.4. Fine-tuning tensor operations used in DNN architecture....94
Table 4.5. Summary statistics of user interaction data....99
Table 4.6. Summary of data from online field test....99
Table 4.7. Summary of ablation study settings....104
Table 4.8. Ablation study results....105
Table 4.9. Summary of node initialization methods used....105
Table 4.10. Field test results....107
Table 6.1. Best hyperparameters found for black-box classifiers....112
Table 6.2. Best hyperparameters found for benchmark local explanation models....114
Table 6.3. Summary of data used for analysis....115
Table 6.4. Summary statistics of the patient-disease bipartite graph....115
Table 6.5. Hyperparameter settings for graph embedding with random walks....116
Table 6.6. Hyperparameter settings for graph embedding with neighborhood aggregation....116
Table 6.7. Hyperparameter settings for feedforward neural networks....117
Table 6.8. List of 30 most common and uncommon medical conditions....117
Table 6.9. Hyperparameter settings for MF methods....119
Table 6.10. Hyperparamter settings for GNN methods....120
Table 6.11. Hyperparameter settings for the DNN model for link prediction....120
Table 6.12. Evaluation results using various tensor operations....120

ABSTRACT

As big data technologies continue to transform our lives, computational design science research (CDSR) is becoming an essential paradigm in the field of information systems (IS). Impact and relevance are central to CDSR, emphasizing the economic and societal impact of IS research. Nonetheless, it remains uncertain how design science research can contribute to the IS body of knowledge. Since design science differs from conventional sciences in its strong emphasis on problem-solving, there have been ongoing discussions about how design science can bring theoretical contributions to the IS field. This dissertation presents three studies that demonstrate how CDSR can contribute to the IS knowledge base while maximizing its broader impact and relevance. The first essay in the dissertation proposes a novel framework to provide individual-level explanations for any black-box machine learning classifier used in healthcare. The second essay develops and implements a fairness-aware graph representation learning for multiple patient outcome prediction in critical care, extending state-of-the-art methods in graph analytics and natural language processing. The third essay proposes a theory-driven graph representation learning framework for reciprocal recommendation in online matching platforms, which show distinct features from conventional user-item recommendation tasks.

1. INTRODUCTION

Commonly characterized by 3**V**'s (large **V**olume, high **V**elocity, and **V**ariety of data) (Laney 2001), big data have shown a vast potential in various application domains such as commerce, public policy, and healthcare (Chen et al. 2012). As a result, big data technologies such as machine learning, distributed data processing, and cloud computing are fundamentally transforming our lives. Accordingly, computational design science research (CDSR) has emerged as one of the novel paradigms in the information systems (IS) research discipline. The key premise of CDSR is impact and relevance for a broader community beyond academia, emphasizing the business and societal impact of research projects. Furthermore, an interdisciplinary approach to problem-solving is emphasized, opening a variety of new research areas such as data representations, computational algorithms, and business analytics (Rai 2016, 2017).

Recently, there have been several exemplary CDSR studies in IS that successfully achieved high impact and relevance. For instance, Martens et al. (2016) demonstrated how massive, fine-grained data on consumer behavior can improve individual-level predictions. Their result presents valuable insights for organizations that it is not only the size but also the granularity of the dataset which matters when using big data in practice. Zhang and Ram (2020) developed a data-driven framework for a comprehensive analysis of triggers and risk factors for asthma. Their framework encompassing machine learning techniques and heterogeneous data provides a novel way to manage the risk of asthma, demonstrating a significant potential to reduce the socio-economic burden of asthma. Whereas Shin et al. (2020) proposed an unstructured data analytics framework using deep learning to enhance social media research.

Nonetheless, design science research has been criticized for a lack of theoretical foundations (Gregor and Hevner 2013). Design science is a practice-oriented discipline with its

roots in the Sciences of the Artificial (Simon 1996). Thus, it deviates from behavioral and natural science paradigms in that its central focus is on creating useful artifacts rather than explaining or describing phenomena of interest (Hevner et al. 2004, March and Smith 1995, Nunamaker et al. 1990). Therefore, there have been constant debates on whether and how design science research can contribute to the IS knowledge base (Gregor and Jones 2007, Hooker 2004, Iivari 2007). Nonetheless, it seems that the jury is still out and the role of theoretical components in design science remains unclear. This ambiguity seems to be escalating in the age of big data, in which the role of theory becomes more obscure and the gap between practice and science widens (Anderson 2008).

In response to such challenges, essays in this dissertation attempt to demonstrate how CDSR can effectively contribute to the IS knowledge base while maximizing broader impact and relevance. CDSR is based on predictive analytics whereas conventional sciences are primarily based on explanatory modeling. Furthermore, predictive analytics differs from explanatory modeling such that the former emphasizes practical utility and predictive power while the latter focuses on explanatory power and causality. This often causes dissonance between empirical analysis and theorizing components in not only the IS discipline but in social sciences in general (Shmueli and Koppius 2011). With growing attention toward predictive analytics, potential ways that big data can contribute to theory and vice versa are increasingly being proposed. For instance, big data can help generate novel theories, compare competing theories, and enhance precision in theories. Conversely, theories can provide focus in analysis and help to assess how constructs are measured (Rai 2016, Shmueli and Koppius 2011).

To this end, the three essays in this dissertation address key issues in contemporary CDSR. The first essay delves into the issue of transparency of artificial intelligence (AI) systems. Despite

their remarkable predictive accuracy, many state-of-the-art computational techniques such as statistical machine learning and deep learning engender black-box models that are not interpretable by human users. Such opaqueness of prediction models can lead to serious issues in practice including safety, ethics, and trust (Ahmad et al. 2018, Doshi-Velez and Kim 2018). Furthermore, it can hamper the researchers' and practitioners' efforts to derive deeper insights from the model by reverse-engineering and iterated experiments (Rai 2020). Thus, in the first essay, I propose design guidelines for explainable artificial intelligence (XAI) systems and demonstrate an explainable prediction method for healthcare based on the guidelines. Specifically, a novel patient-level XAI framework that can be applied to any type of supervised machine learning called robust local explanations (ROLEX) is proposed. The ROLEX framework is validated using real-world big data from a cohort study and two public healthcare datasets.

The second essay attempts to address the issue of fairness in AI-assisted decision-making. When predicting multiple patient outcomes in intensive care units (ICU), practitioners need to be aware of the impact of potentially unfair predictions, which can arise from the highly skewed distribution of data. Such bias in training data, called *prevalence bias* in the essay, is a universal problem in various domains including healthcare. For instance, in the dataset used in this study, it was observed that a handful of medical conditions such as acute respiratory failure and pneumonia occurred in thousands of patients while most conditions occurred less than 100 times. Conventional AI systems trained with biased training data tend to inflate the occurrence of majority instances, potentially providing incompetent decision support for the minority group. However, on the other hand, overemphasizing fairness can lead to suboptimal predictive performance for the majority of instances, resulting in inefficient decision support overall. Hence, a fairness-aware graph representation learning framework that can be used for assessing multiple

disease risks in ICUs is proposed in the essay. Using a large-scale public dataset in critical care, the framework's potential to contribute to healthcare practice is demonstrated.

Another approach highlighted in this dissertation is enhancing methodological rigor and practical relevance by incorporating concepts and theories from other relevant disciplines. This can be critical in design science research since design theories differ drastically from conventional theories in that they are practice-oriented and can be drawn from other disciplines (Gregor and Jones 2007, Hooker 2004). It can also encourage incorporating kernel theories and the development of mid-range theories in CDSR research. Furthermore, as mentioned, the interdisciplinary investigation of a problem of interest is at the heart of CDSR, emphasizing broader business and societal impact. Such a theory-driven approach is demonstrated in the third essay of this dissertation. The graph representation learning framework for reciprocal recommendation in online matching platforms extends and improves upon a state-of-the-art graph neural network (GNN) method. However, it considers the unique characteristics of online matching platforms, which deviate from conventional user-item recommendation services, with methodological improvements based on social science and behavioral theories. As a result, it significantly improves the performance of predicting potential matches in a real-world mobile dating service compared to existing methods.

In the remainder of this proposal, each essay will be delineated with background, related work, and research design. Table 1.1 summarizes the three studies included in the dissertation.

Table 1.1. Summary of the three essays in the dissertation

Essay	Research Objective	Methodologies	Proposed Design Artifact
I	Propose a novel framework to explain black-box artificial intelligence systems for healthcare	- Local, Model-agnostic Explanations - Bayesian Optimization	Robust Local Explanations (ROLEX)

II	Propose a fairness-aware multiple disease risk prediction framework in critical care	- Graph Representation Learning - Collaborative Filtering - Language Modeling	Fairness-aware prediction framework for multiple outcome prediction of individual patients in ICUs (Fair2Vec)
III	Design a theory-driven reciprocal recommender system for online matching services	- Graph Neural Networks - Collaborative Filtering	Graph representation learning framework for reciprocal recommendations (MatchSAGE)

2. ESSAY 1: A Novel Method for Interpretable Machine Learning using Robust Local Explanations

2.1. Introduction

With recent developments in big data technologies and the prevalence of large-scale datasets, the healthcare predictive analytics (HPA) field is witnessing a dramatic surge of interest in the information systems (IS) discipline. Existing IS studies focusing on HPA have proposed novel methods to optimally predict health outcomes such as readmission of patients (Bardhan et al. 2015) and abnormal inhaler usage (Son et al. 2020). Nevertheless, many HPA systems are not put into practice due to a lack of trust (Ahmad et al. 2018, Ribeiro et al. 2016). One of the most important steps towards garnering trust in HPA systems is providing reliable explanations for its predictions. Such explanations play a critical role in supporting clinical decision-making, facilitating user engagement, and enhancing patient safety. Furthermore, it is highly risky to blindly accept and apply the results derived from black-box models. It might lead to undesirable consequences or life-threatening outcomes (Ahmad et al. 2018).

In this study, we propose a novel method to enhance the transparency and accountability of the HPA model by providing human-interpretable explanations of its predictions. Our proposed method called **RO**bust **L**ocal **EX**planations (ROLEX) can be used to generate robust explanations at an instance level for any classification model. In other words, it can be used to generate *model-agnostic, local explanations*. Model-agnosticity refers to the capability of explaining any supervised prediction model (Ribeiro et al. 2016). Existing methods in HPA attempt to explain specific prediction models such as neural networks or tree-based ensembles (e.g., Che et al. 2016; Choi et al. 2016). We claim that such a model-dependent approach to developing explanations is not aligned with recent trends in HPA, in which a wide variety of datasets and diverse types of

prediction models are deployed. Thus, model-agnostic explanation methods enhance the applicability of HPA to a wide range of problems. Secondly, in HPA, it is common to encounter complex, non-linear relationships between variables with heterogenous effects across observations. It is therefore insufficient to explain the model by providing only global, dataset-level explanations such as feature importance estimates. Local explanations refer to instance-level explanations that can be understood by humans (Ribeiro et al. 2016). In the context of HPA, local explanations can help to facilitate personalized care and advance precision medicine by providing patient-level interpretations. Our method is designed to produce more locally faithful explanations than existing local explanation methods through improved sampling and a novel optimization scheme. Finally, we develop a prototype user interface for ROLEX that comprises diverse modes of human-interpretable, patient-level explanations.

We evaluate the proposed method using multiple HPA datasets – a cross-sectional, large-scale dataset from a cohort study and two public healthcare datasets. ROLEX consistently outperforms state-of-the-art explanation methods (Jia et al. 2019; Laugel et al. 2018b; Ribeiro et al. 2016) in terms of local faithfulness, which is defined as the ability to replicate the black-box prediction model in the vicinity of the instance of interest measured by a quantitative metric (Laugel et al. 2018b). Thus, ROLEX shows the potential to enhance the transparency and accountability of various HPA systems.

2.2. Related Work

We use two distinct criteria that are widely accepted in the field, i.e., *the scope of explanations* and *model specificity* (Rai 2020), to classify extant explanation methods. One criterion to classify explanation methods is the *scope of explanations*. Global explanations attempt to explain patterns on a dataset level; this can be achieved by examining the model and the dataset

itself. Coefficient estimates of dependent variables for a regression model and conditional rules developed from a single decision tree (DT) are classic examples of global explanations. In contrast, local explanations focus on a part of the dataset – i.e., certain data instances of interest. Interactions among different features and relationships between input and outcome variables are likely to be highly complex and non-linear in many cases. Nonetheless, the relationship between the outcome(s) and inputs may be explained using a different linear function for each data record, or through instance-level explanations (Rai 2020, Ribeiro et al. 2016). A promising method to design a complex artifact is to find "viable ways of decomposing it into semi-independent components corresponding to its many functional parts" (Simon 1996). In a similar vein, local explanations for intelligent systems can be regarded as reverse designing a complex artifact to understand the phenomenon of interest more deeply.

The second criterion utilized to categorize explanation methods is *model specificity*. Explanation methods can either be considered model-specific or model-agnostic. Model-specific methods apply to only certain types of models. In many cases, model-specific methods pertain to one of the inherent properties of the model. For instance, the attention mechanism (Bahdanau et al. 2015) would be a specific method only applicable to neural network models. In contrast, model-agnostic methods are flexible enough to be applied to different types of supervised learning models.

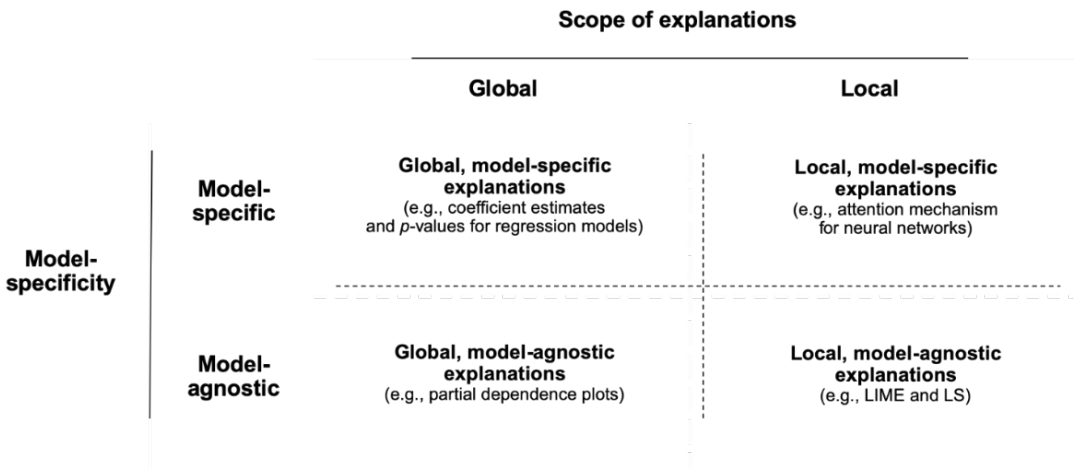


Figure 2.1. Taxonomy of XAI methods

Combining the two criteria results in four types of explanation methods (Figure 2.1). First, most global, model-specific methods take advantage of inherently interpretable models such as logistic regression. Recently, models combining the predictive power of deep neural networks and interpretable models have been proposed (e.g., Yang et al. 2018). Also, attempts to visualize features of deep neural networks on a global level would fall into this category. The attention mechanism for neural networks is one of the most widely used examples of local, model-specific methods. Attentional methods align input and output sequences and provide corresponding weights between the two, enabling explanations for output predictions (Bahdanau et al. 2015). For global, model-agnostic explanations, greedy function approximation algorithms are often used to estimate the partial function expressing the global relationship between variables (Friedman 2001). Whereas local surrogate methods are widely used for local, model-agnostic explanations. Local surrogate methods fit an interpretable model to approximate how the prediction model behaves on a local level. Methods such as perturbation sampling (Ribeiro et al. 2016) and feature attribution (Lundberg and Lee 2017) have been proposed to approximate the global prediction model's behavior at an instance level.

In this study, we focus on *local, model-agnostic* XAI methods to develop an explanation method that is suitable for general HPA systems. Most machine learning algorithms operate based on the principle of inductive bias, making certain assumptions specific to the data (Mitchell 1980). Algorithms to learn patterns from sequential data, e.g., recursive neural networks, assume temporal dependence between inputs, and convolutional neural networks are based on the intuition that objects in image-like data are localized in a few pixels. In the healthcare context, for instance, models such as hidden Markov models and recurrent neural networks can be effective for clinical text data compared to other models. Thus, to maximize the applicability to any HPA problem promptly, a model-agnostic method would be a preferable choice for generating explanations. A primary reason for focusing on local explanations stems from the complexity of problems in healthcare, as described previously. In such cases, high explanatory power on a global scale is insufficient to make practical inferences with actionable implications.

Specifically, we focus on perturbation sampling methods (Jia et al. 2019; Laugel et al. 2018b; Ribeiro et al. 2016) among many local, model-agnostic explanation techniques. In this family of methods, samples are generated around an instance to train local explanation models that can be directly interpreted by humans (e.g., linear regression). Perturbation sampling in the neighborhood of a data instance of interest is used to maximize the *local faithfulness* of local explanation models, defined as the capacity to replicate how the prediction model behaves in the vicinity of the instance (Ribeiro et al. 2016). The local interpretable model-agnostic explanations (LIME) method (Ribeiro et al. 2016) is a widely accepted local, model-agnostic method using perturbation sampling. For each data instance being explained, LIME generates synthetic samples in the entire input feature space and assigns weights to them in proportion to the distance from the

data instance of interest. Finally, a local explanation model – i.e., a penalized regression model – is trained with weighted local samples.

However, it has several shortcomings, including the lack of objective evaluative criteria and diluted local faithfulness of explanations. Accordingly, Laugel et al. (2018b) proposed a new quantitative evaluation criterion for local explanations and improvements in the sampling scheme for training instances to enhance local faithfulness with their local surrogate (LS) framework. LS starts with finding a data point that is close to the DB near the instance being explained, i.e., the "enemy" point. Then, it uniformly samples data points in the neighborhood of the data instance, which is defined as an ϵ-ball around the enemy point. The local embedding aided perturbation (LEAP) (Jia et al. 2019) locally samples training data in the sub-manifold of the neighborhood to improve the quality of data points. It is argued that the improved sampling procedures with emphasis on locality enhance the local faithfulness of ensuing explanations by feeding the explanation model with high-quality training samples.

Nonetheless, these approaches also have limitations. First, they do not consider the fact that instances in the same dataset can drastically differ in terms of their relationship to local DB. Moreover, explanations for instances that are too far from DB can show high fidelity scores even though they are not locally faithful. Secondly, existing methods assume that DB generated by a black-box model tends to be linear in the local scale, i.e., in the vicinity of the instance. However, our close examination of real-world data reveals that DB can be highly non-linear in some cases, leading to several potential issues. First, the training samples can be highly imbalanced. In turn, highly imbalanced samples can lead to biases in the explanation model. Furthermore, using only a linear explanation model could lead to unfaithful interpretations in some cases. Finally, most existing methods lack a human-interpretable user interface, which can be highly critical in real-

world application scenarios. The proposed ROLEX method, which is described in detail in the following section, attempts to address these limitations.

2.3. Proposed Method

2.3.1. Overview

Before explaining our proposed method, we briefly describe general notations and common procedures of *local, model-agnostic* explanation methods. First, a black-box prediction model ($\hat{f}: X \rightarrow Y$), e.g., deep neural networks, is trained in advance. Then, an interpretable local model (s_{x_i}), e.g., logistic regression, is trained to explain the black-box model. Note that a desirable property of the local model is *local faithfulness* to the data instance being explained (x_i), instead of *global faithfulness* to the black-box model. In training the local model, perturbation sampling is one of the most widely used techniques (Jia et al. 2019; Laugel et al. 2018b; Ribeiro et al. 2016). Perturbation sampling methods first generate synthetic samples around the instance x_i ($\{x_j^{(i)} \in R^d, j = 1, 2, \ldots, N\}$). Then, a local explanation model (s_{x_i}) that can be directly interpreted by humans, e.g., linear models and decision rules, are trained using the samples ($x_j^{(i)}$'s) as input features and predicted labels by the black-box model ($\hat{f}(x_j^{(i)})$'s) as a target. Figure 2.2 shows a synthetic example demonstrating the process of generating local, model-agnostic explanations for a binary classifier $\hat{f}$. In the figure, the blue region is where instances are negatively classified by $\hat{f}$, whereas the orange region contains instances that are positively classified; one can see that DB is highly non-linear on a global scale. The local explanation models s_{x_1} and s_{x_2} are locally faithful to the black-box model $\hat{f}$ around x_1 and x_2, respectively. Furthermore, they are both simple models that can be easily interpreted.

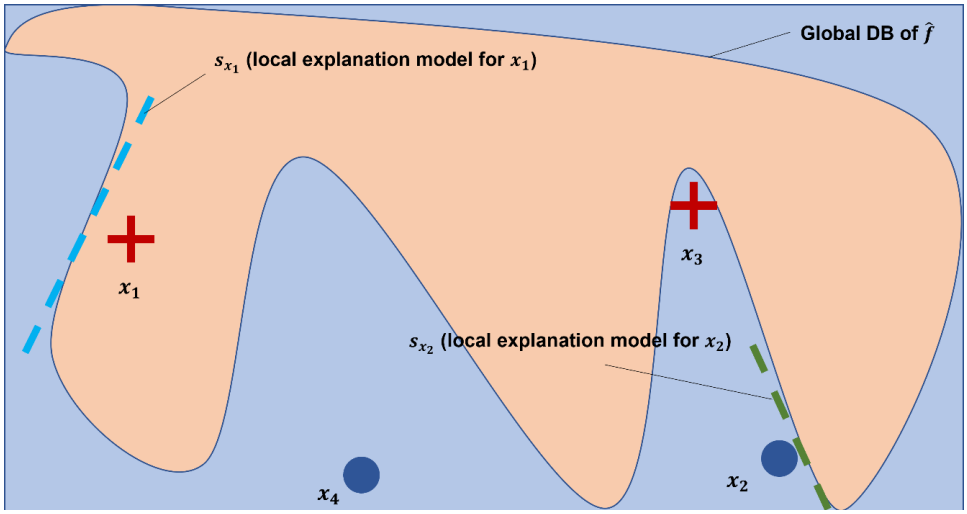


Figure 2.2. Illustration of local explanation models faithful to a black-box model

In the context of HPA, each data instance corresponds to a patient with quantifiable clinical or non-clinical characteristics that are likely to be relevant in predicting future healthcare outcomes, e.g., fragility fracture risks and the likelihood of developing breast cancer. After predicting the fragility risk of patients with $\hat{f}$, we aim to generate independent explanations for the classifier's decision for patient-level prediction, e.g., high/low risks of fragility fracture. Finally, it should be emphasized that the local explanation models ($s_{x_i} \in S_X$) are trained to primarily explain the black-box classifier, not to describe the phenomenon itself. Hence, regardless of the relationship between the black-box classifier and the phenomenon, the explanation model attempts to mimic the classifier. In some cases, the classifier may well approximate the phenomenon of interest – in such cases, the local explanation models can be used to represent both the classifier and the phenomenon. However, the classifier may be inefficient in mimicking the complex relationships between variables, i.e., an under-fitted prediction model. For instance, the classifier may be inappropriately trained due to an insufficient number of training samples or a highly skewed training dataset. Nonetheless, the explanation model can still be valuable in such scenarios. For example, it can be used to point out where and why the classifier is underperforming. Such

insights can be exploited to improve the classifier in the future by providing potential remedies for underfitting.

Hence, we propose the ROLEX method that improves over existing local, model-agnostic explanation methods (Jia et al. 2019; Laugel et al. 2018b; Ribeiro et al. 2016). The ROLEX method has two consecutive phases to generate local explanations - i.e., (1) synthetic data generation and (2) local model fitting.

2.3.2. Synthetic Data Generation

The objective of this step is to generate synthetic data points ($\{x_j^{(i)}, j = 1, 2, \ldots, N_{train}\}$) that reflects DB close to x_i such that the explanation model (s_{x_i}) is locally faithful. To this end, LIME samples around the data instance x_i using a Gaussian perturbation sampling scheme (Ribeiro et al. 2016):

$$P(x_i, \eta) = x_i + \eta, \eta \sim \mathcal{N}(0, 1) \quad (1)$$

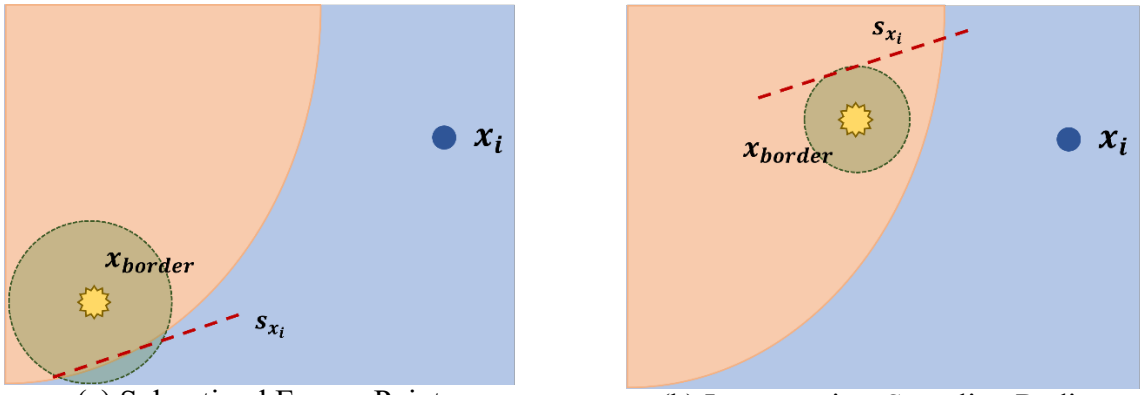
However, it was pointed out that this sampling scheme can include data instances that are far away from x_i, hampering the local faithfulness of s_{x_i} (Laugel et al. 2018b). Therefore, LS first tries to find the nearest "enemy" point ($x_{border}^{(i)}$) with an opposite predicted class, i.e., $\hat{f}(x_i) \neq \hat{f}\left(x_{border}^{(i)}\right)$, using the growing spheres algorithm by iterated sampling while contracting and expanding the radius (Laugel et al. 2018a). Then, it uniformly samples in a ϵ-ball around $x_{border}^{(i)}$, i.e., $\{x \in R^d | distance(x, x_{border}^{(i)}) < \epsilon\}$, where d is the dimensionality of the feature space and $distance(\cdot)$ is a pre-defined distance metric such as Euclidean or Manhattan distances. ϵ is a pre-defined small value, e.g., 0.1 or 0.3 (Laugel et al. 2018b):

$$P(x_i, \eta) = x_{border}^{(i)} + \eta, \eta \sim Unif(\epsilon) \quad (2)$$

LEAP first obtains the local embedding of x_i ($x^{(i)}_{embedding}$) using a subspace learning technique such as the principal component analysis (PCA) to consider the local intrinsic dimensionality of x_i. Then, data points are uniformly sampled in the vicinity of $x^{(i)}_{embedding}$, similar to LS:

$$P(x_i, \eta) = x^{(i)}_{embedding} + \eta, \eta \sim Unif(\epsilon) \tag{3}$$

Afterward, sampled instances are mapped back to the original feature space (Jia et al. 2019).



(a) Suboptimal Enemy Point (b) Inappropriate Sampling Radius

Figure 2.3. Examples demonstrating drawbacks of existing local explanation methods

Nevertheless, the improved sampling schemes of LS and LEAP are not without drawbacks. First, although the growing spheres algorithm for LS is guaranteed to converge, the enemy point ($x^{(i)}_{border}$) discovered by the algorithm is not necessarily an optimal one that can maximize local faithfulness, e.g., $x^{(i)}_{border}$ in Panel (a) of Figure 2.3. Moreover, a different enemy point ($x^{(i)}_{border}$) can be found for the same x_i in every iteration as it is based on stochastic sampling. Similarly, there is no guarantee that the local subspace learned by PCA fully reflects the local DB around x_i, ensuring high fidelity. Such issues are exacerbated when the training data are high-dimensional and complex and $\hat{f}$ is highly non-linear, which is commonly observed in modern healthcare and medical datasets. Secondly, LS and LEAP apply the same sampling radius (ϵ) to all data points being explained. This can be problematic in cases where the distance to the closest DB differs greatly across different x_i's, which is also common in many real-world datasets. For instance, ϵ in

panel (b) of Figure 2.3 is set to a small value so that the samples do not appropriately reflect the local DB.

To tackle the aforementioned issues, we take a distinct approach of directly optimizing the key parameters for the uniform sampling scheme. To be specific, we find $x_{center}^{(i)}$ and ϵ_i while maximizing the local fidelity of s_{x_i} trained with data points obtained with the sampling scheme defined as:

$$P(x_i, \eta) = x_{center}^{(i)} + \eta, \eta \sim Unif(\epsilon_i) \tag{4}$$

Here, $x_{center}^{(i)}$ is a data point in the feature space between x_i and the closest enemy point in the training set ($x_{train\_border}^{(i)}$), i.e., the closest training instance with the opposite predicted class:

$$x_{center}^{(i)} = \beta_i * x_{train\_border}^{(i)} + (1 - \beta_i) * x_i, 0 < \beta_i < 1 \tag{5}$$

Note that unlike $x_{border}^{(i)}$, $x_{train\_border}^{(i)}$ is a stationary point in the subset of the training data ($\{x_j | f(x_j) \neq f(x_i)\}$), which does not require a heuristic algorithm (e.g., growing spheres) to find. We can just compute the distance between x_i and x_j's (in $\{x_j | f(x_j) \neq f(x_i)\}$) and select one having the shortest distance to x_i. Thus, the final optimization objective is as follows:

$$\underset{\beta_i, \epsilon_i}{\operatorname{argmax}} \mathcal{L}\left(s_{x_i} \middle| \left\{x_j^{(i)}, j = 1, 2, \ldots, N_{train}\right\}\right), \tag{6}$$

where $\mathcal{L}\left(s_{x_i} \middle| \left\{x_j^{(i)}, j = 1, 2, \ldots, N_{train}\right\}\right)$ is the local fidelity of the explanation model (s_{x_i}) trained with data points $\left\{x_j^{(i)}, j = 1, 2, \ldots, N_{train}\right\}$ generated with the sampling scheme in Equation 7. Most off-the-shelf optimization schemes, e.g., Bayesian optimization (Snoek et al. 2012), can be used for solving Equation 9. By using a fixed enemy point in the training set ($x_{train\_border}^{(i)}$) instead of applying a stochastic update in every iteration as in LS, we obtain more stable explanations across different iterations, enhancing the robustness of explanations.

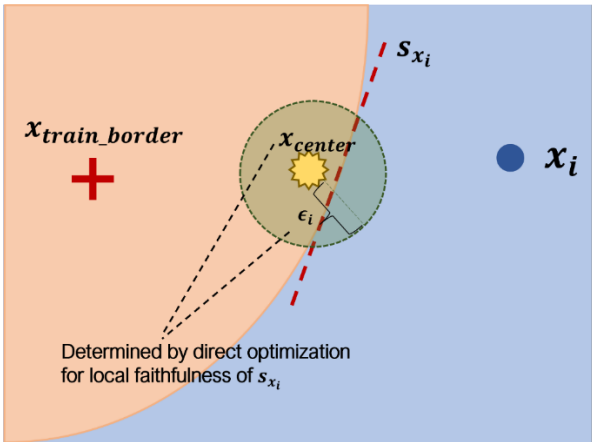


Figure 2.4. Synthetic data generation process of ROLEX

Furthermore, by directly optimizing the objective function rather than relying on heuristics, we can further improve local faithfulness. That is, several cases of suboptimal parameter tuning, which can lead to impaired quality of training samples, can be circumvented by direct optimization. Figure 2.4 describes the data generation process of our local explanation method for instance x_i.

While developing local explanation models, existing methods often assume the local linearity of the DB. However, local DB can be non-linear in real data (e.g., DB around x_3 in Figure 3). This can lead to several issues in training such as an imbalance in the sampled data points and sub-optimal explanation models which assume linearity. First and foremost, since decision boundaries tend to be highly non-linear as mentioned, uniformly sampling in the vicinity of x_i can lead to imbalanced training data, i.e., having much more data points with one specific predicted label than others. This can be especially critical for instances far away from a DB. Hence, we use the synthetic minority over-sampling technique (SMOTE) (Chawla et al. 2002) to further improve the quality of synthetic data points. After uniformly sampling around $x_{center}^{(i)}$, we apply SMOTE to the minority class, reducing the likelihood of s_{x_i} trained in a biased manner towards the majority class.

2.3.3. Local Model Fitting

Using the synthetic data points and their predicted labels by the pre-trained black-box classifier, a local explanation model is trained. As described earlier, the synthetic data points ($\{x_j^{(i)}, j = 1, 2, \ldots, N\}$) are used as input features for an interpretable local model (s_{x_i}), with their predicted labels ($\{\hat{f}(x_j^{(i)}), j = 1, 2, \ldots, N\}$) as a target. Many existing methods use linear models, e.g., penalized linear regression, as a local model, assuming that local DB is approximately linear. However, as mentioned, there could exist cases where local DB is non-linear. In such cases, we argue that non-linear interpretable algorithms such as DT and generalized additive models (GAM) could be effectively used. In determining the non-linearity of local samples, and ultimately which algorithm could be used for local explanations, we propose the *Linear Separability* (*LinSep*) score:

$$LinSep(x_i) = bAcc_{x_j^{(i)} \in V_{x_i}} \left(\hat{f}\left(x_j^{(i)}\right), p_{x_i}(x_j^{(i)}) \right), \tag{7}$$

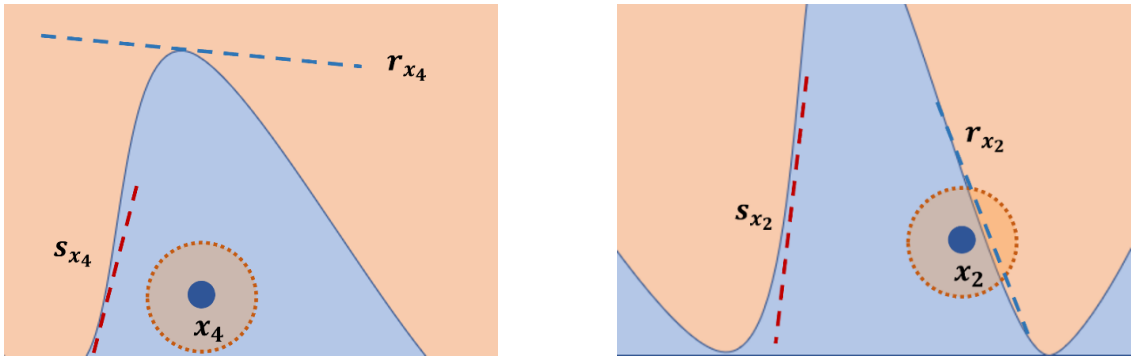
where $bAcc$ is balanced accuracy and V_{x_i} is a neighborhood of x_i, defined as an ϵ-ball around x_i, i.e., $V_{x_i} = \{x_j^{(i)} \in R^d | distance(x_j^{(i)}, x_i) < \epsilon\}$. p_{x_i} is a linear classifier, e.g., a single-layer perceptron, trained with sampled instances in V_{x_i}. Local DB is completely linear, and all samples are linearly separably if the *LinSep* score is 1.0, however, in reality, such a case is uncommon. Thus, we determine that the local DB of an instance is approximately linear if the *LinSep* score is above a certain threshold, e.g., 0.8 in our case, and use a linear explanation model in that case. Otherwise, we use a non-linear explanation model. Nonetheless, it should also be noted that the complexity of explanations and faithfulness has a trade-off relationship in general (Poyiadzi et al. 2021). That is, a non-linear explanation model could show a superior fidelity compared to a linear explanation model, however, it could be too complex to be human-interpretable. For example, a DT classifier with 100 leaf nodes is not easily interpreted by a human. Thus, in many cases, a non-

linear explanation model should be regularized or trimmed to strike a balance between faithfulness and complexity.

After the local model is trained, it is evaluated to assess the local faithfulness with respect to the original classifier. The *Local Fidelity* score is a widely used metric to quantify the local faithfulness of local explanation models:

$$LocalFid(s_{x_i}, x_i) = Acc_{x_j^{(i)} \in V_{x_i}} \left(\hat{f}\left(x_j^{(i)}\right), s_{x_i}(x_j^{(i)}) \right), \tag{8}$$

where Acc is a standard measure of accuracies such as classification accuracy and F-score. To approximate this, test data points are sampled in the neighborhood V_{x_i} and accuracy measures are calculated using them. In short, the *Local Fidelity* score is an approximate measure that quantifies how well the local explanation model (s_{x_i}) mimics the black-box classifier ($\hat{f}$) in the neighborhood of the data instance being explained (x_i).



(a) Local Explanation Models for $\boldsymbol{x_4}$ (b) Local Explanation Models for $\boldsymbol{x_2}$

Figure 2.5. Local explanations for two hypothetical instances

However, the neighborhood of instances with a greater distance from the DB tends to be homogeneous, i.e., with the same predicted label. Thus, test data points sampled in the neighborhood of such instances will tend to be homogeneous or highly biased, similar to the imbalance in synthetic data points in the data generation phase. Such homogeneity in test samples will render most standard predictive performance metrics used for local fidelity computation to

become ineffective. To demonstrate, consider the case of the instance x_4 in Figure 3 (Panel (a) in Figure 2.5). Let's assume that we have trained two local explanation models for the instance - s_{x_4} and r_{x_4}. It is clear that s_{x_4} is more locally faithful to r_{x_4}, the local fidelity scores of the two models computed using samples nearby x_4 (in the orange circle) would be identical if using standard metrics. For example, the accuracy score would be 1 for either model as both models trivially classify all the points within the orange circle as 1. In contrast, in the case of the instance x_2 (Panel (b) in Figure 2.5), which is close to the local DB, the local fidelity score will choose r_{x_2} over s_{x_2} since now we have both labels to be classified. Furthermore, local explanations for instances such as x_4 are likely to be invalid and less informative. Note that although s_{x_4} is relatively more locally faithful than r_{x_4}, it is still considerably distant from x_4. In other words, although it approximates the closest local DB of x_4, it is likely to be not locally faithful enough to capture instance-specific patterns of x_4. When compared to r_{x_2}, which is in the close vicinity of x_2, it is much less localized and thus less informative in terms of providing instance-level interpretations. In such cases, it is advised that decision-makers take caution in interpreting a local explanation since it can be not as locally faithful as that for other instances. For example, we can consider comparing them to other modes of explanation, e.g., global explanations, or explanations for nearby instances having heterogeneous samples.

Therefore, we propose a novel local explanations framework that takes into account such edge cases. Figure 2.6 describes the decision procedure for the proposed LDA framework. First, instances (x_i) are tested whether the samples around them are homogeneous, i.e., having only a single type of predicted label. If they are homogeneous, the instance is excluded from fidelity evaluation, and the ensuing local explanation warrants a special caution in interpreting it (Case 1). We describe the detailed procedure for this step and the resulting evaluation metric, i.e., local DB-

aware (LDA) fidelity score in Algorithm 2.1. The *LDA-Fidelity* score is a better approximation of the local faithfulness of explanation models, compared to the *Local Fidelity* score proposed by Laugel et al. (2018b). If the samples around the instance (x_i) are heterogeneous, test the linearity of local DB using the *LinSep* score. If the *LinSep* score is over a certain threshold, i.e., if the local DB is approximately linear, train a linear explanation model such as logistic regression (Case 2). If the *LinSep* score is below the threshold, train a non-linear local explanation model (e.g., DT or GAM) (Case 3). However, in this case, the complexity of ensuing explanations should be closely monitored to ensure human interpretability.

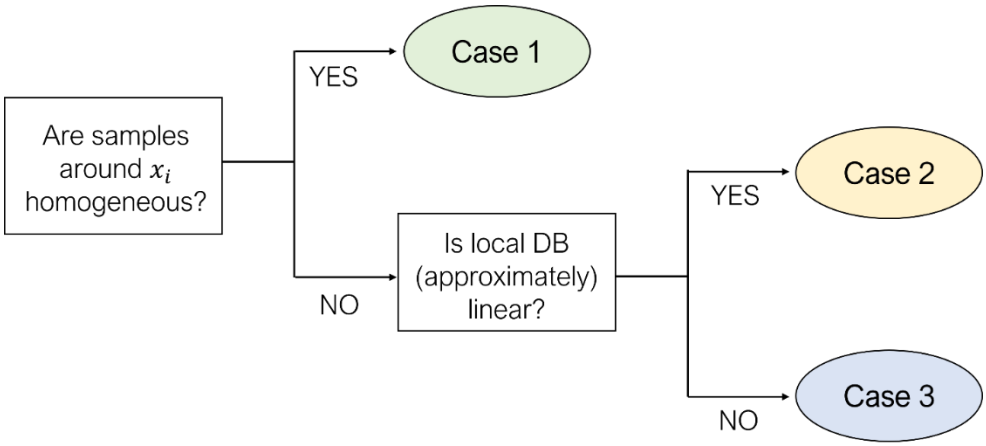


Figure 2.6. Decision procedure for training local explanation models

---

Input: Local explanation models $s(x_i)$, test instances (x_i), black-box classifier $\left(\hat{f}: X \to Y\right)$, $i = 1, 2, \ldots, n$

Initialize: r_{test}, N_{train}, N_{test}, *Acc (any standard evaluation metric for prediction models – e.g., AUC, precision, and recall)*, X_D *(an empty set)*

Algorithm:

1. For each i:
   i. Sample instances $(z \in Z \mid \|x_i - z\|_2 \leq r_{test})_{N_{train}}$
   ii. If $\exists\, f(z_1) \neq f(z_2)\ \forall\, z_1, z_2 \in Z$, for sampled data points, include x_i into a set of decision-sensitive instances X_D.
2. Calculate the fidelity score for each x_i in X_D:
   i. Sample test instances $(z' \in Z' \mid \|x_i - z'\|_2 \leq r_{test})_{N_{test}}$
   ii. Compute the local fidelity score:

$$LocalFid\left(s_{x_i}, x_i\right) = Acc_{z' \in Z}\left(\hat{f}(z'), s_{x_i}(z')\right)$$

3. Calculate the average LDA-Fidelity score for instances in X_D:

$$LDA\_Fid = average\left(LocalFid\left(s_{x_i}, x_i\right)\right), x_i \in X_D$$

Output: Dataset-level decision-sensitive local fidelity score LDA_Fid

Algorithm 2.1. Computing LDA fidelity score

2.4. Analysis

2.4.1. Data

We used three healthcare datasets to assess the applicability of the ROLEX method for various HPA problems. Specifically, we used two widely used public datasets and a large-scale dataset from a cohort study related to fragility fracture. The first two are the Wisconsin breast cancer dataset[1] and the Cleveland heart disease dataset[2], which are included in the University of California Irvine (UCI) Machine Learning Repository. As both datasets are commonly used to assess the predictive performance of a novel HPA model, we use them to assess the local faithfulness of explanations generated by ROLEX and ensure the generalizability of the method. The last dataset is from the Ansung cohort study, a prospective study started in 2001, supported by the National Genome Research Institute of the Korea Centers for Disease Control and Prevention (Kim et al. 2017). Data were collected using a large community-based epidemiological survey of a patient population of Korean men and women (40 to 69 years old). 5,018 participants completed a preliminary examination and survey in 2001. For our study, we selected a subset of participants with valid dual-energy X-ray absorptiometry (DXA) data indicating the presence or absence of a fracture 10 years after starting the cohort study, following a widely used guideline for fracture risk identification (Kanis et al. 2009). As mentioned, we use the cohort dataset to assess the applicability of the ROLEX method in building trustworthy and accountable HPA systems in

[1] https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)
[2] https://archive.ics.uci.edu/ml/datasets/heart+disease

the real world, developing a prototype user interface and local explanations in collaboration with experts.

Fragility fractures are becoming a severe medical and economic issue worldwide (Reginster and Burlet 2006). In the United States, the occurrence of fractures is expected to increase from 1.9 million to 3.2 million between 2018 and 2040 with associated costs climbing up to $95 billion from $57 billion (Lewiecki et al. 2019). With growing concerns over the medical and economic burden of fracture-related diseases, the importance of early diagnosis and intervention before the first fracture is becoming critical (Reginster and Burlet 2006). Once patients with higher fracture risks are identified in advance, effective measures can be taken to prevent them from developing it. For instance, it has been demonstrated that active medication for women with osteoporosis can significantly reduce their fracture risks (Black et al. 2000), while regular workouts and extra care in dietary intake can reduce the risk of fracture in general. To enable efficient and agile prediction of fragility fracture risks, a set of features relevant to fragility fracture prediction was shortlisted by an experienced physician from a collection of over 3,000 indicators. The resulting dataset for analysis comprises 2,227 patients and 29 attributes and 570 of them were reported to have a fragility fracture. The input attributes are classified into five categories, i.e., demographics, lifestyle, physical examination, medical records, and bone quality scores (Table 2.1). As part of data pre-processing, records with null values in outcomes and clinically identified predictors were excluded resulting in 2,074 patients. Finally, we split the datasets into training and test sets in an 80:20 ratio. The training set is used to tune hyperparameters and train the models and the test set is used to evaluate them.

Table 2.1. Description of attributes in the fragility fracture dataset

Category	Description	Examples
Demographics	Self-reported general information about the patient	Gender, age, income, education level

Lifestyle	Self-reported information about the habitual aspects	Alcohol consumption, smoking, exercising
Physical examination	Records measured by professional medical examiners	Height, weight, hormone level, body fat percentage
Medical records	Previous medical records of patients that are relevant to fragility fractures	Previous fracture occurrences, parental fracture occurrences
Bone quality scores	Scores indicating the quality of the bone structure	Trabecular bone score (TBS), bone mineral density (BMD)

2.4.2. Evaluation of Machine Learning Prediction Models

We compare the performance of two interpretable and five black-box machine learning models for binary classification. We included two inherently interpretable models, DT and decision rule lists (DRL), as simple baselines. Furthermore, we included widely-used ensemble models, i.e., random forest (RF) and AdaBoost (ADA), and a deep learning model with multi-layer perceptron (MLP). Finally, we included advanced boosting methods, i.e., extreme gradient boosting (XGB) and categorical boosting (CB), which show an outstanding performance with tabular data in many extant studies and machine learning challenges (Chen and Guestrin 2016).

Table 2.2. Evaluation results of machine learning algorithms for prediction tasks

Dataset	Breast Cancer			Heart Disease			Fragility Fracture		
Metric / Algorithm	AUC	ACC	bACC	AUC	ACC	bACC	AUC	ACC	bACC
DRL	.4643	.3758	.3156	.6035	.5570	.5629	.5383	.6455	.5590
DT	.9340	.9362	.9340	.7560	.7562	.7560	.5575	.6449	.5575
RF	.9728	**.9561**	.9342	.9218	.8524	.8480	.6921	**.7542**	.5396
ADA	.9735	.9210	.9078	**.9267**	.8688	.8669	.6843	.7493	.5001
MLP	.9774	.9473	.8881	.9310	.8852	.8841	.7040	.7493	.5436
XGB	**.9844**	.9122	.9078	.8954	.8524	.8480	.6691	.7493	**.5595**
CB	.9833	**.9561**	**.9407**	.9148	**.9016**	**.8981**	**.7170**	**.7542**	.5231

The result is summarized in Table 2.2. We report AUC, accuracy (ACC), and balanced accuracy (bACC) to take into account the class imbalance prevalent in healthcare data, i.e., more negative class instances than positive instances. Overall, advanced boosting algorithms, i.e., XGB and CB, show improved performances to other models as the datasets are in a tabular format. We report the optimal hyperparameters found with a five-fold cross-validation in Appendix 6.1. In the

following section, we show the evaluation results for local explanations fitted for the five machine learning algorithms trained here.

2.4.3. Evaluation of Local Explanation Models

Before reporting the evaluation results of local explanation models, we show how the proposed LDA framework can be used to assess each instance in data and the overall quality of the dataset. There are largely two types of edge cases with respect to the local DB in the LDA framework, i.e., (1) homogeneous training/test samples due to a large distance from DB, and (2) nonlinearity in a local DB. We show the number of edge (and non-edge) cases in Table 2.3. First, it should be noted that there are a considerable number of edge cases in most scenarios, which might lead to locally unfaithful explanations when especially when using existing methods such as LIME and LEAP. Furthermore, it is apparent that the same instances can be either an edge case or not depending on the algorithm and dataset. For example, most instances (99%) have a linear DB when an MLP classifier is trained on the fragility fracture dataset. In contrast, more than half of the instances (64%) have a nonlinear DB when an XGB classifier is trained on the same data. Thus, users will have to be aware of such different patterns and attentive when interpreting different local explanations even though they are similar instances from the same dataset.

Table 2.3. Number of edge cases in different datasets/algorithms

Algorithm	**Dataset**	**# Instances with heterogeneous samples**	**# Instances with homogeneous samples**	**# Instances with linear DB**	**# Instances with nonlinear DB**
RF	Breast Cancer	64 (56%)	50 (44%)	112 (98%)	2 (2%)
	Heart Disease	30 (49%)	31 (51%)	58 (95%)	3 (5%)
	Fragility Fracture	336 (81%)	79 (19%)	327 (79%)	88 (21%)
ADA	Breast Cancer	81 (71%)	33 (29%)	109 (96%)	5 (4%)
	Heart Disease	15 (25%)	46 (75%)	58 (95%)	3 (5%)
	Fragility Fracture	119 (29%)	296 (71%)	412 (99%)	3 (1%)
MLP	Breast Cancer	89 (78%)	25 (22%)	114 (100%)	0 (0%)
	Heart Disease	18 (30%)	43 (70%)	61 (100%)	0 (0%)
	Fragility Fracture	278 (67%)	137 (33%)	412 (99%)	3 (1%)
XGB	Breast Cancer	72 (63%)	42 (37%)	98 (86%)	16 (14%)

	Heart Disease	42 (69%)	19 (31%)	50 (82%)	11 (18%)
	Fragility Fracture	414 (99%)	1 (1%)	149 (36%)	266 (64%)
CB	Breast Cancer	47 (41%)	67 (59%)	112 (98%)	2 (2%)
	Heart Disease	26 (43%)	35 (57%)	55 (90%)	6 (10%)
	Fragility Fracture	245 (59%)	170 (41%)	393 (95%)	22 (5%)

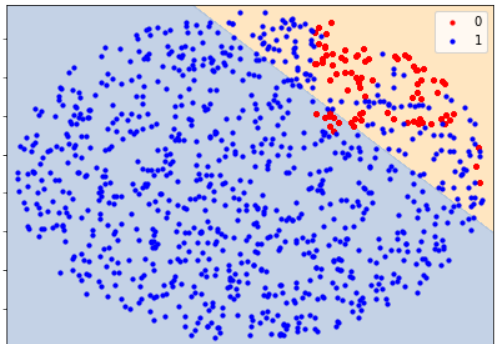
*** Note**: The proportion of each case to the total number of instances (%) is in parentheses

We fit local explanation models to the machine learning models trained in the previous section. We report both the *Local Fidelity* (Laugel et al. 2018b) and *LDA-Fidelity* scores presented in this study. The balanced accuracy score, which is equivalent to the average of recall scores for the positive and negative classes (Brodersen et al. 2010), is used as the measure of accuracy when calculating fidelity scores (Acc) and we set the sampling radius when sampling the test data points (r_{test}) to be proportional to the $max\_distance$ following the heuristic suggested by Laugel et al. (2018b). Therefore, $max\_distance$ is the maximal Euclidean distance between the instance being explained (x_i) and train instances. Finally, we set the testing hyperparameters N_{train}, N_{test} (the number of instances sampled for training and testing s_{x_i}), and r_{test} to 5000, 5000, and $max\_distance$*0.2, respectively. Changing N_{train} or N_{test} higher than a certain threshold value (e.g., 500) did not make a substantial difference.

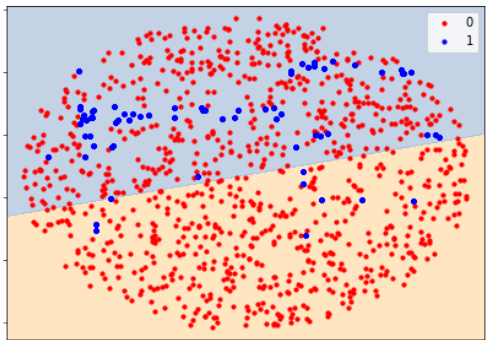
Table 2.4. Evaluation results of local explanation models

Dataset		Breast Cancer		Heart Disease		Fragility Fracture	
Algorithm \ Metric		Fidelity	LDA-Fidelity	Fidelity	LDA-Fidelity	Fidelity	LDA-Fidelity
RF	LIME	.7946	.6342	.8072	.6197	.6987	.6280
	LS	.8253	.6889	.8445	.6881	.6879	.6145
	LEAP	.7214	.5038	.7990	.5097	.5960	.5010
	ROLEX	**.9439**	**.9001**	**.9446**	**.8875**	**.7989**	**.7404**
ADA	LIME	.7296	.6195	.8732	.5815	.8574	.5028
	LS	.7683	.6743	.9892	.9572	.8885	.6575
	LEAP	.6470	.5032	.8804	.5139	.8545	.5009
	ROLEX	**.9339**	**.9070**	**.9895**	**.9573**	**.9711**	**.8994**
MLP	LIME	.7186	.6396	.7622	.5589	.7982	.7015
	LS	.9800	.9745	.9946	.9818	.9662	.9608
	LEAP	.6129	.5042	.8566	.5142	.6654	.5005
	ROLEX	**.9905**	**.9878**	**.9955**	**.9848**	**.9849**	**.9775**
XGB	LIME	.7488	.6023	.7551	.6460	.5620	.5610
	LS	.7571	.6154	.8306	.7546	.6038	.6028

	LEAP	.6875	.5052	.6586	.5041	.5017	.5005
	ROLEX	**.9085**	**.8551**	**.9174**	**.8801**	**.7442**	**.7436**
CB	LIME	.8531	.6437	.8371	.6428	.8016	.6642
	LS	.8862	.7244	.9057	.7847	.8205	.7187
	LEAP	.7960	.5053	.7896	.5063	.7051	.5005
	ROLEX	**.9529**	**.8857**	**.9562**	**.8973**	**.9118**	**.8507**



(a) Neighborhood of an instance having a high linear separability score



(b) Neighborhood of an instance having a low linear separability score

Figure 2.7. Comparison of instances with low and high linear separability scores

Table 2.4 is the summary of evaluation results for local explanation models. It shows the best fidelity scores found for each algorithm after conducting a grid search for optimal hyperparameters. The optimal hyperparameters are reported in Appendix 6.2. Note that ROLEX does not require hyperparameter selection or tuning as it relies on direct optimization of the objective function instead of heuristics. This makes the method more robust and efficient, in addition to providing more locally faithful explanations. However, the level of regularization, e.g., the maximum number of leaf nodes in a decision tree, should be determined in advance. Since this is closely related to human interpretability, which is a critical issue in many real-world cases, we discuss this in detail shortly. Finally, we set the threshold for determining the linear separability of the local neighborhood to 0.8 as instances having the *LinSep* score above 0.8 generally showed local DB that is approximately linear (e.g., panel (a) of Figure 2.7) in comparison to instances

having the score below 0.8 (e.g., panel (b) of Figure 2.7)[3]. However, since this can depend on the dataset used, future studies can experiment with other values.

ROLEX shows consistently higher Local Fidelity and LDA-Fidelity scores compared to established baselines in perturbation sampling-based local explanation methods. LS performed better than LIME and LEAP in most cases, indicating that sampling near the enemy points is more effective than sampling in the entire space and penalizing the samples having a larger distance to the instance.

We also performed an ablation study to determine how each factor contributes to the improved performance of ROLEX. We considered three scenarios – (1) without Bayesian optimization (A), (2) without oversampling (SMOTE) (B), and (3) without non-linear explanations (C). Table 2.5 shows ablation study results on the fragility fracture dataset in terms of average fidelity scores. Overall, removing the oversampling component (B) led to the largest decline in the fidelity score, which means that oversampling using SMOTE contributed the most to improving the performance. In contrast, removing the non-linear explanation component (C) showed a performance decline only when explaining the XGB classifier. A highly likely reason for the finding is that black-box models using other algorithms showed an average *LinSep* score higher than .9, which implies that most instances have linear local DB. In contrast, the black-box model trained with XGB showed an average *LinSep* score of .8, which indicates that there are several instances with non-linear local DB. Hence, using non-linear explanation models for those instances led to performance improvement. In short, training samples for the local explanation models are

[3] These visualizations are created by uniformly sampling 1,000 data points in the vicinity of instances having the *LinSep* score higher than 0.8 (a) and lower than 0.8 (b) in a two-dimensional space. The *x*- and *y*-variables are the patient's BMD and TBS scores, respectively. The red points are samples which are predicted as having a low risk of fracture and the blue points are predicted as having a high risk of fracture by the XGB model.

likely to be imbalanced in many cases, which is supported by a significant contribution of the oversampling scheme. Furthermore, when the local DB is not linear, it is most effective to combine oversampling and non-linear explanation models.

Table 2.5. Ablation study results

Algorithm	ROLEX	ROLEX – A	ROLEX - B	ROLEX - C
RF	.7898	.7603	.6467	.7902
ADA	.9711	.9515	.9035	.9727
MLP	.9849	.9543	.9143	.9866
XGB	.7442	.6846	.6540	.7113
CB	.9118	.8488	.7477	.9097

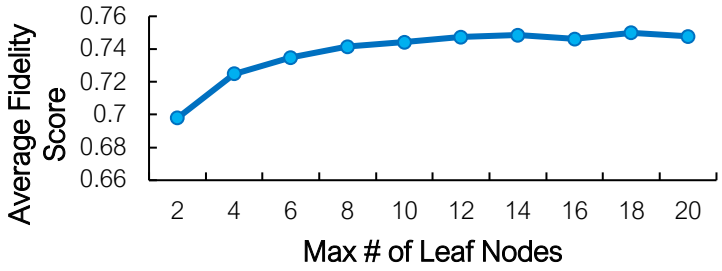


Figure 2.8. Fidelity-complexity trade-off

As shown in the ablation study results, having non-linear explanations can lead to improvement in local faithfulness when local DB is non-linear, e.g., in the case of explaining the XGB classifier. However, as mentioned, this can lead to hampered human interpretability of explanations since faithfulness and complexity have a trade-off relationship in general. Thus, we performed an additional experiment to show how regularization can help balance the two. Specifically, we changed the maximum number of leaf nodes in the decision tree from 2 to 20 and tracked how the average local fidelity score changed (Figure 2.8). As expected, fidelity increases as the complexity of the model increases in general. However, after a certain point – around 10 leaf nodes in this case – the fidelity score gets plateaued and the gain from adding more leaf nodes becomes marginal. Thus, we set the maximum number of leaf nodes to 10 in this study since it provides a moderate level of human interpretability and a high fidelity score.

2.5. Discussion

2.5.1. Practical Implications

Trust strongly influences the usability of IT artifacts, and reliable explanations play an important role in building trust (Gregor and Benbasat 1999). Using our ROLEX method, physicians and patients can gain trust in HPA systems by interpreting complex black-box models and discerning their clinical relevance. Furthermore, researchers and practitioners can derive new insights into patient outcomes by exploring the heterogeneous patterns across local explanation models using multiple approaches. Such insights can contribute to patient safety and well-being.

In general, well-designed XAI methods can enable the development of *relevant, ethical, safe,* and *reliable* AI systems to solve real-world problems. Compared to existing methods in local, model-agnostic explanation (Jia et al. 2019; Laugel et al. 2018b; Ribeiro et al. 2016), our method enhances local faithfulness of explanations, which enables more fine-grained interpretations and applications. In turn, this can improve personalized treatment and care and advance precision medicine, addressing grand challenges in healthcare. Moreover, local explanations generated by ROLEX can improve the overall HPA system by supporting the reverse-engineering process. For example, since local explanations enable individualized interpretations, they can help experts discover patterns beyond average treatment effects. The experts can examine the heterogeneous patterns among various patient groups and identify different predictors that are salient for different groups. Then, machine learning models for different patient groups having more parsimonious and relevant features can be trained.

Furthermore, ROLEX can contribute to real-world problem-solving in light of *ethics and safety* by enhancing the ability to detect and expose the incompleteness of the system. Decisions made by healthcare practitioners can have a considerable effect on the quality of multiple lives and

occasionally determine the life expectancy of patients. Our evaluation framework helps comprehensively navigate the local explanations procedure. That is, it not only helps compare local explainable methods but also informs as to when local explanations can be more (or less) effective. Furthermore, we emphasize human-centric elements of an XAI system, i.e., human-interpretability, which can help practitioners better articulate the incompleteness of the overall system. It has the potential to bolster the *transparency and accountability* of the black-box prediction models in HPA systems. This can reduce conflicts between stakeholders in the healthcare field, e.g., between physicians and patients, most of which arise from information asymmetry (Bloom et al. 2008).

2.5.2. Contributions to the IS Knowledge Base

Lin et al. (2017) discussed how solving healthcare problems with an HPA system is relevant and can contribute to IS discipline, especially in design science research. Extending the discussion, we claim that our ROLEX method provides the ability to explain results from such systems and therefore is relevant for multiple research streams in IS, including *healthcare IS*, *big data and predictive analytics*, and *design science*. Our study sheds light on patient safety and user acceptance, which are highly critical in the healthcare context (Ahmad et al. 2018). It has been established that trust plays a critical role in the user's acceptance of technology in various contexts, and we believe that it plays a more important role in the context of healthcare as users are generally more reluctant to change (Fichman et al. 2011). Furthermore, enhanced transparency can encourage conscientious decision-making on behalf of greater public interest.

Explanations with the help of domain experts and the reverse-designing process can support the discovery of new knowledge regarding a phenomenon of interest (Doshi-Velez and Kim 2018). In the context of big data analytics, this can be one way to exploit the synergistic relationship between big data and theory. Moreover, we discussed that such human-interpretable,

lucid explanations promote information diffusion across diverse stakeholders and possible reduction of conflicts among them. Finally, we pay attention to guidelines for designing XAI systems for real-world problem-solving. We propose that XAI systems are design artifacts with distinct features from conventional AI systems. Based on the features, we suggest guidelines for designing XAI systems in practice that can help researchers examine their XAI methods and set the theoretical foundation of their studies. We also attempt to advance current standards of XAI methods with theoretical discussions and in-depth investigation of the problem-solving process in healthcare.

2.5.3. Limitations and Future Work

Our study has a few limitations and considerations. The optimization process of finding optimal parameters (β_i, ϵ_i) can be slightly more time-consuming especially when the dataset is complex and highly non-linear. Nevertheless, directly optimizing for the local fidelity metric eliminates the need for fine-tuning hyperparameters (e.g., the initial value of the sampling radius) or applying other heuristic algorithms (e.g., PCA for LEAP and growing spheres for LS), thus compensating for the optimal parameter search cost. Furthermore, it considerably improves the faithfulness of explanations measured with both *Local Fidelity* and *LDA-Fidelity* scores. Future work can improve the optimization process, e.g., explore other optimization methods than Bayesian optimization. Moreover, since the evaluation results can differ based on how we set r_{test}, future research can focus on the relationship between learned parameters (e.g., β_i, ϵ_i) and r_{test} and their impact on local faithfulness of explanations.

Another limitation of our method is commonly shared by local explanation models for producing multifaceted interpretation mechanisms. Global linear models have a single set of coefficients for generating interpretations. Whereas each data point has its own set of models and

coefficients in local explanations. This requires the end-user to inspect each data point, possibly inducing mental overload. On the contrary, although it is less common in healthcare, there could be edge cases in which explanations for many instances are homogeneous. That is, global and local explanations become essentially equivalent in a certain region of the feature space. In either case, it might be helpful to assess mid-level explanations, e.g., explanations for patient subgroups. Future work can focus on the systematic exploration of local boundaries to extricate homogeneous subgroups and ways to generate trustworthy and reliable mid-level explanations. Similarly, generating robust and reliable explanations for constantly changing data, i.e., data shifts, can be a nontrivial issue. For instance, if the distribution of new data differs significantly from existing data for which explanations are generated, current explanations might be less valuable. How to utilize local explanation methods to generalize in dynamic environments in which data shifts are prevalent can be an interesting research direction.

Although our primary focus in this study is on explaining HPA systems, we believe that our method can be generalized to other domains having similar characteristics. Future work can investigate the applicability of local explanation methods such as ROLEX across different datasets and domains. Furthermore, while we have focused on explaining black-box binary classifiers in this study, there is ample scope for extending ROLEX to explain multiclass/continuous outcome prediction as well as other facets of machine learning such as unsupervised learning and reinforcement learning.

2.6. Conclusion

In this study, we introduce a theoretical framework for designing and deploying XAI systems in practice and use it to develop a method for generating local explanations for safe and user-centric healthcare predictive analytics. We demonstrate the applicability of our method by

developing a prototype HPA system for predicting and explaining fragility fracture risk. Our study contributes to the IS knowledge base, particularly in *healthcare IS, big data analytics,* and *design science* by introducing a new design artifact for explaining predictions that shed light on trustworthy and user-centric IS.

3. ESSAY 2: Fairness-aware Graph Representation Learning for Multiple Patient Outcome Prediction in Intensive Care Units

3.1. Introduction

Intensive care units (ICUs) are the most resource-intensive and costly wards in large hospitals. Patients admitted to ICUs suffer from critical ailments and often endure life-threatening adverse events during their treatment (Rowan et al. 1993, Unal et al. 2015, Zimmerman et al. 2013). Thus, it has been estimated that the daily cost of an ICU stay is three times higher than a general ward, with a growing number of patients requiring specialized, expensive treatments (Canadian Institute for Health Information 2016). Furthermore, the patients require timely care and intervention – service delays in ICUs have significant adverse effects on patients' health status and treatment outcomes (Chan et al. 2017, Young et al. 2003). However, ICUs are often understaffed and patients receive insufficient attention from physicians and nurses, considerably risking the quality of care and treatment outcome (Numata et al. 2006, Wallace et al. 2012). To make matters worse, intensivists are confronted with massive amounts of data from electronic patient-data management systems (Hyland et al. 2020). Contemporary patient-data management systems contain tremendous information from various sources such as electronic health records, posing challenges in identifying key facts for making clinical decisions. Therefore, it would be valuable to accurately assess patients' health status and predict their prognosis using automated systems, thereby enabling a timely and effective intervention of intensivists and efficient resource management of the hospital.

To this end, prior studies have proposed data-driven frameworks and methods to predict patient outcomes in ICUs, prompted by the increasing availability of big data in critical care (Ghassemi et al. 2015, Sanchez-Pinto et al. 2018). Such studies focus on predicting a handful of

patient outcomes (e.g., mortality, length of stay, and the onset of a major ailment) using patient-level data (e.g., the patient's demographic information and acute/chronic health conditions). However, a more comprehensive assessment of the patient's overall health condition can be valuable in critical care practice. Most chronic and acute medical conditions have considerable complications and comorbidities that occur during the treatment process (Wollschlager and Conrad 1988). For instance, in the dataset used in this study, we found that ICU patients suffer more than seven medical conditions on average after admission (Pollard et al. 2018). Thus, ICU practitioners can benefit more from pre-empting the onset of several medical conditions and understanding co-occurrence patterns between them than solely focusing on the onset of a single condition. Such information can be effectively used for providing better treatments as well as for preventing future complications and adverse events.

In this study, we propose using a graph representation learning approach to exploit complex association patterns between various health-related events in ICU. Graph representation learning is a suitable methodology to model complicated relations between various entities such as patients and medical conditions in our study. However, there remain challenges regarding inherent biases in critical care data such that common medical conditions occur extremely frequently while other conditions are uncommonly observed. For instance, in the dataset used in this study, we observed that a handful of medical conditions such as acute respiratory failure and pneumonia occurred in thousands of patients while most conditions occurred less than 100 times (Figure 3.1). This is a widespread phenomenon in many real-world applications such as e-commerce (Brynjolfsson et al. 2011), natural language (Powers 1998), and social networks (Barabási and Bonabeau 2003). Since most machine learning (ML) algorithms tend to prefer prevalent instances, they tend to

underrepresent less common and rare ones. We name this *prevalence bias*, which potentially leads to unfair decision-making in practice (Mehrabi et al. 2021).

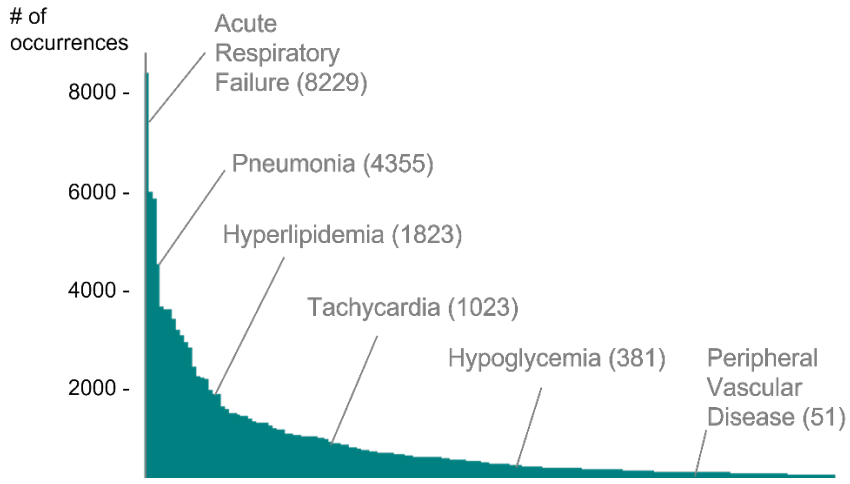


Figure 3.1. Skewed distribution in critical care data (Pollard et al. 2018)

Previous studies have proposed techniques and methods to mitigate prevalence bias in artificial intelligence (AI)-enabled decision support systems such as recommender systems in e-commerce and social network platforms (Abdollahpouri et al. 2017, Li et al. 2021, Zhu et al. 2021). However, there has been scant attention to address this issue in other domains such as healthcare operations. Nonetheless, this can be even more problematic in healthcare since the stakes are high – unfair decisions can lead to grave consequences, risking the patient's life and well-being. Moreover, how fair AI systems can enhance the practical utility in the real world remains unclear. For instance, recent studies suggest evidence for the adverse socio-economic impact of using fair AI and ML algorithms (Fu et al. 2022, Liu et al. 2018). Thus, much remains to be understood about the integration of fairness in real-world AI systems and the impact of deploying such systems in the healthcare operations context (De-Arteaga et al. 2022, McCradden et al. 2020). For instance, in our multiple patient outcome prediction problem in ICUs, conventional ML algorithms can overemphasize prevalent medical conditions, underrepresenting minor ones. Consequently, patients suffering from uncommon or rare diseases can be treated

"unfairly" since the system will show a comparatively low predictive performance for them. In contrast, "fair" algorithms might overcompensate for uncommon conditions at the expense of predictive performance for prevalent medical conditions, which often take a heavy toll on caregivers and hospitals. Therefore, accentuating fairness can be an ineffective strategy from the entire hospital's operations management perspective. We argue that such factors should be fully addressed when designing an AI system for real-world decision support in ICUs.

To fill this research gap, in this study, we propose a novel graph representation learning framework for multiple outcome prediction in ICUs (Figure 3.2). We first represent patient-level diagnosis records as a bipartite graph having two types of nodes, i.e., patients and medical conditions. Thereafter, node sequences are sampled using random walks, in which neighboring nodes are sequentially chosen according to a pre-defined transition scheme (Grover and Leskovec 2016, Perozzi et al. 2014). Using the sampled sequences as inputs, graph embeddings, which are low-dimensional vectors representing each node in the graph, are obtained with a neural network model (Mikolov et al. 2013).

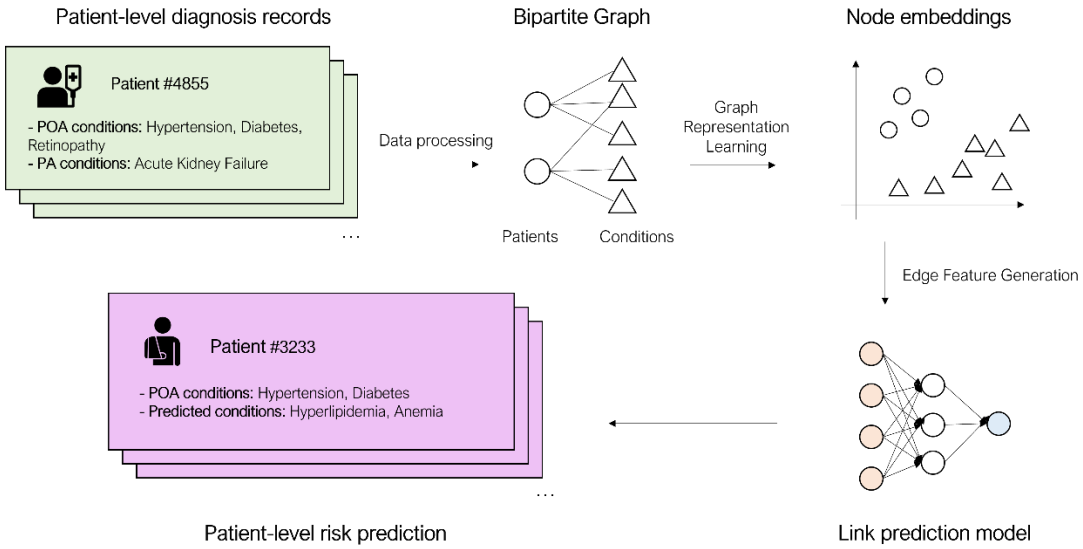


Figure 3.2. Design architecture of the proposed framework

We propose a novel method called Fair2Vec to learn embeddings for patient nodes. Patient nodes can be underrepresented during the random walk and sampling process due to fewer connecting links (Gao et al. 2018). We utilize an insight from sentence embedding methods in natural language processing, in which a sentence is represented as a combination of words (Arora et al. 2017, Wieting et al. 2016). In a similar vein, we represent patient nodes as a combination of node vectors corresponding to medical conditions present at the point of admission (POA). Specifically, we suggest three different representation techniques in Fair2Vec, each of which has a distinct weighting scheme to control the level of prevalence bias and fairness of predictions. This not only helps in overcoming the underrepresentation problem but also reduces the computational burden and realizes timely prediction and decision-making. Furthermore, our approach is inductive in the sense that our framework can be used to make predictions for new patients not included in the initial bipartite graph.

Finally, the edge features are constructed by combining the computed node embeddings, and a supervised ML model is trained to predict the probability of having an edge between a patient and a medical condition in the future, i.e., the likelihood of the patient developing the condition after admission. We extensively evaluate the proposed method with a large-scale public dataset in critical care (Pollard et al. 2018), comparing the performance with state-of-the-art methods in ICU outcome prediction and graph representation learning. Experimental results show that prediction models trained using Fair2Vec significantly outperform strong baselines. Moreover, we show how the accuracy and fairness of predictions change according to the weighting scheme. We find that we can obtain more fair predictions by assigning higher weights to uncommon medical conditions, which results in lower accuracy – i.e., the accuracy-fairness trade-off. In contrast, we can enhance accuracy for major medical conditions such as hypotension and sepsis, which often take a heavy

toll on hospitals, at the expense of fairness of predictions. Different weighting schemes also impact racial disparity in predictive performance, which is an important issue in applying the results in the wild (De-Arteaga et al. 2022, Sunar and Swaminathan 2022). Hence, in addition to a novel method, our study provides guidance in considering various factors (e.g., a trade-off between fairness in patient care and overall cost reduction) when implementing and applying the method in healthcare practice.

3.2. Related Work

3.2.1. Patient Outcome Prediction in Critical Care

Most patients admitted to critical care wards are in dire health conditions, suffering from severe ailments and hospital-associated infections that are often life-threatening (Choi et al. 2022). It has been reported that the mortality rate in ICU is significantly higher than in other wards, exceeding 50% in some cases. Furthermore, approximately 80% of the deaths occurred within fifteen days after admission (Rowan et al. 1993, Unal et al. 2015, Zimmerman et al. 2013). However, ICU physicians and caregivers are often understaffed and overwhelmed with a tremendous amount of information from various patient-data management systems, which can significantly deteriorate the quality of care that the patients are receiving (Duke et al. 2004, Hyland et al. 2020). Thus, accurately predicting ICU patients' prognosis is of vital importance from both medical and management perspectives since it can enable timely intervention and facilitates efficient resource management of the hospital. To this end, data-driven scoring systems such as Assessment and Chronic Health Evaluation (APACHE), Simplified Acute Physiology Score (SAPS), and Mortality Probability Model (MPM0), have been developed (Salluh and Soares 2014). These systems predict the severity of individual patients, e.g., mortality and length of stay, using patient-level data, e.g., physiologic data and chronic health conditions.

By using standardized variables collected by most hospitals and simplified models, such systems enable prompt assessment of a patient's health status and allow facile comparison of quality of care across different units or hospitals. An intensivist can easily compute the expected mortality rate and length of stay of a patient by typing in a few values regarding the health condition of the patient. However, there are several limitations in such scoring systems including the inability to consider unique characteristics of hospitals and patient groups (e.g., the demographic composition of patients) and potential deterioration in predictive performance over time due to external changes (e.g., the emergence of new diseases) (Choi et al. 2022). For instance, average mortality or length of stay will be significantly higher in a hospital that is specialized for senior citizens compared to a hospital that serves a younger population. In addition, they primarily focus on predicting a only couple of dimensions of patient outcomes, i.e., mortality and length of stay, thereby providing limited information to caregivers. Nonetheless, the practitioners might be interested in knowing more specific information regarding the patient's ongoing health status during treatment such as infections or acute conditions that the patient is likely to suffer from during the treatment or side effects from some medications.

Considering the limitations of existing systems, researchers are increasingly proposing novel methods that utilize advanced ML techniques trained with large healthcare data (Ghassemi et al. 2015, Sanchez-Pinto et al. 2018). Several recent studies show that combining state-of-the-art algorithms such as deep learning (Huang et al. 2022, Thorsen-Meyer et al. 2020) and ensemble models (Choi et al. 2022, Safaei et al. 2022). By exploiting richer patient-level data from diverse sources, they significantly improve the predictive performance and applicability of the method. However, there remains room for improvement in terms of the scope of predictions, i.e., the model outcome. As mentioned, most previous studies focused on only predicting patient mortality and

length of stay. Furthermore, although recent studies are increasingly proposing methods to predict health outcomes other than the two, most focus on predicting only a single salient medical condition such as the onset of sepsis (Nemati et al. 2018, Sheetrit et al. 2019), delirium (Green et al. 2019), and hypotension (Ghosh et al. 2016). We acknowledge that such events are of critical importance since they not only are major causes of mortality but also incur a considerable amount of costs to the hospital. Nevertheless, we believe that a more comprehensive assessment of the patient's overall health status can be more valuable for decision support in critical care settings.

Since most chronic and acute medical conditions have considerable complications and comorbidities that occur during the treatment process (Wollschlager and Conrad 1988), intensivists might be interested in better understanding the co-occurrence patterns between different health events. For demonstration, Figure 3.3 visualizes a knowledge graph extracted from Semantic MEDLINE (SemMED), which is a leading biomedical information management application and contains a substantial amount of existing medical knowledge (Rindflesch et al. 2011). We extracted relations between four medical conditions prevalent in ICU (i.e., pneumonia, renal failure, sepsis, and respiratory failure) using SemMED and converted them into a knowledge graph[4]. The four medical conditions are highly associated with each other, showing various relationships (e.g., "complicates", "coexists with", and "causes") between them.

[4] Relations were identified by searching for disease pairs on the SemMED official web application (https://ii.nlm.nih.gov/SemMed/semmed.html)

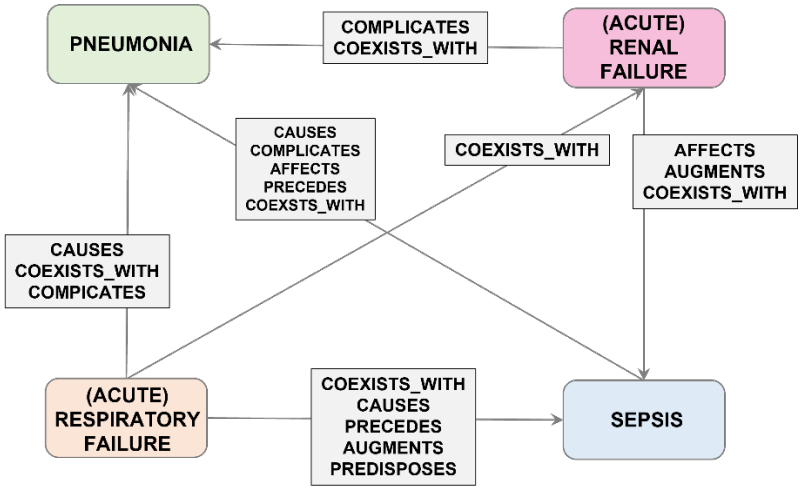


Figure 3.3. Relations between four major medical conditions in ICU

To address this research gap, we propose a novel data-driven framework for multiple outcome prediction in critical care. From the healthcare operations perspective, our framework enables more comprehensive risk assessment in ICU, enhancing patient care and efficient resource management of the hospital. As described earlier, most health-related events (e.g., post-admission (PA) conditions, adverse events, and hospital-acquired infections) that ICU patients suffer from are associated with each other (such as in Figure 3.3). Thus, we adapt and extend established methods in graph representation learning that are widely used to learn complicated co-occurrence patterns between various entities. However, existing methods show limitations in addressing algorithmic challenges such as highly skewed distribution in ICU data and potentially unfair suggestions arising from inherent biases.

Graph representation learning is an ideal technique to analyze interdependencies between multiple types of entities in complex systems (Hamilton et al. 2017b). We believe that it can be effectively applied to outcome prediction in critical care, where there are large fine-grained patient data. Graph representation learning can be used to analyze interactions between various entities in the data (e.g., patients and medical conditions) and improve patient-level predictions. As a result,

it can advance healthcare practice in ICUs by reducing costs and enhancing resource allocation, which are salient problems in critical care nowadays (Hyland et al. 2020).

3.2.2. Bias and Fairness in AI-Assisted Decision-Making

Regardless of its predictive performance, every prediction model suffers from underlying algorithmic biases. There are various types of algorithmic biases that can lead to different unfair decision-making in many applications including logistics, supply chain management, and healthcare (Cohen 2018, De-Arteaga et al. 2022, Sunar and Swaminathan 2022). For instance, biases can arise from the interaction between data and algorithm, algorithm and user, and user and data (Mehrabi et al. 2021). Among them, in this study, we focus on a type of bias arising from uneven and skewed distribution, which we name *prevalence bias*. In many real-world data, it is common to observe a small number of frequent instances being accountable for the vast majority of occurrences. For example, top sellers dominate many electronic marketplaces (Brynjolfsson et al. 2011) and most Hollywood actors have only a handful of links to other actors while a few of them are connected to thousands (Barabási and Bonabeau 2003). In healthcare, a few patients account for a considerable amount of costs during their stay at hospitals (Srinivasan et al. 2018).

We observe similar patterns in critical care data - major medical conditions (e.g., sepsis and hypertension) are commonly observed among many patients whereas minor conditions (e.g., hypernatremia and hypocalcemia) are infrequently observed. Since most ML algorithms are designed to maximize overall accuracy, they tend to overprioritize instances in a majority group (Mehrabi et al. 2021). For instance, AI systems for movie recommender systems tend to assign a higher predicted rating to popular movies compared to less popular movies although a user is likely to demonstrate a similar level of preference for both movies (Zhu et al. 2021). Thus, it can be expected that ML models trained with such data can underrepresent and neglect minor medical conditions and events.

There have been several innovative attempts to address the issue of prevalence bias in real data. For instance, Zhu et al., (2021) proposed a novel post-processing algorithm that can compensate for lower popularity, and Abdollahpouri et al., (2017) introduced a regularization-based framework to enhance the coverage of recommendations for less popular items. Prior work primarily focuses on business applications, in which skewed distributions are widespread and thoroughly studied (Brynjolfsson et al. 2011). Hence, most studies tackle how to reduce biases in algorithms widely used for commercial recommender systems such as the matrix factorization technique (Koren et al. 2009). However, there has been scant attention to designing algorithms that can alleviate prevalence bias in graph-based methods, which are increasingly used in healthcare applications (Su et al. 2020). We believe this is a salient and relevant issue since uneven and skewed distributions are commonly observed in biomedical data and the stakes of potentially unfair decision-making are high in the healthcare domain (McCradden et al. 2020).

Moreover, although fairness is important in promoting values such as social justice and equality, the jury is still out on whether emphasizing fairness in algorithmic decision-making enhances the practical utility of the solution. In addition to potential limitations related to previously proposed notions of fairness (Corbett-Davies and Goel 2018, Kleinberg et al. 2017), there can be the unintended socio-economic impact of deploying fair ML algorithms and AI systems (Fu et al. 2022, Liu et al. 2018, McCradden et al. 2020). Fu et al., (2022) showed that enforcing fairness in algorithmic decision-making can lead to underinvestment in improving the predictive accuracy of algorithms by firms. Such a strategic choice can make every stakeholder worse off, including the protected group by the algorithm. McCradden et al., (2020) pointed out the limitations of current fairness solutions in healthcare predictive analytics. They argue that

healthcare practitioners relying only on operationalized metrics for fairness can face grave ethical problems if real-world consequences are not adequately considered.

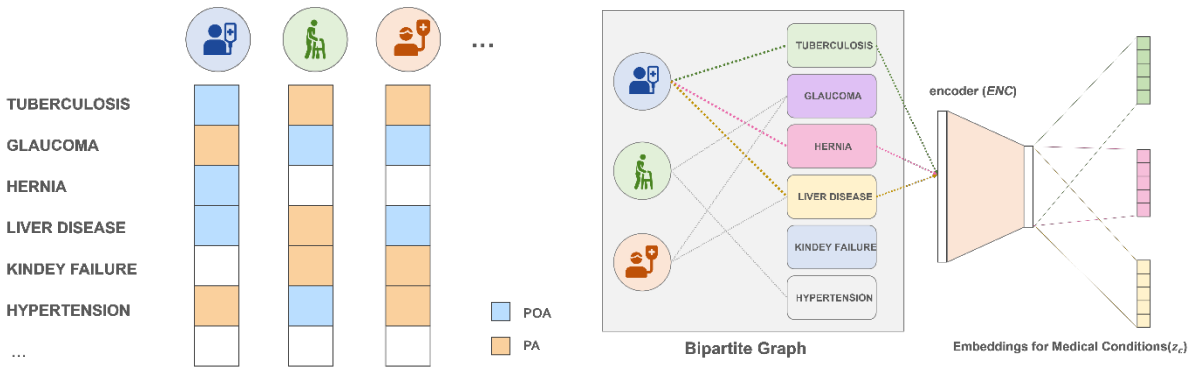
Therefore, we argue that not only how to enhance the fairness of algorithms but also the multifaceted, real-world implications should be thoroughly scrutinized when designing a fair AI system. This could be especially critical in domains with high stakes such as healthcare. For example, to address prevalence bias among various medical conditions with a long-tail distribution, the algorithm might overcompensate for rare medical conditions, excessively penalizing the common conditions. Consequently, the prediction model might suffer from low accuracy for common diseases, which often takes a heavy toll on physicians and caregivers Hence, a fair ML model can lead to the decreased utility of the key stakeholders and impaired managerial efficiency of the hospital. In this study, we propose a novel graph representation learning framework called Fair2Vec, which enables a user to control the level of fairness in the presence of prevalence bias and assess its impact on major/minor instances. Furthermore, we extensively evaluate the framework using widely used accuracy and fairness metrics, and discuss implications on clinical decision-making in critical care practice.

3.3. Methodology

In this section, we describe our proposed framework for outcome prediction in ICU. The overall objective of the framework is to predict the individual patient's health outcomes after admission given the information available shortly after the patient was admitted. In other words, using the data available during the first t_{obs} hours of stay in the ICU after admission, we attempt to assess the probability of multiple health events before discharge (t_{pred}). Specifically, in this study, we focus on predicting the onset of PA conditions, which are diagnosed by ICU physicians during the prediction period. As described earlier, previous studies focus on predicting the risk of

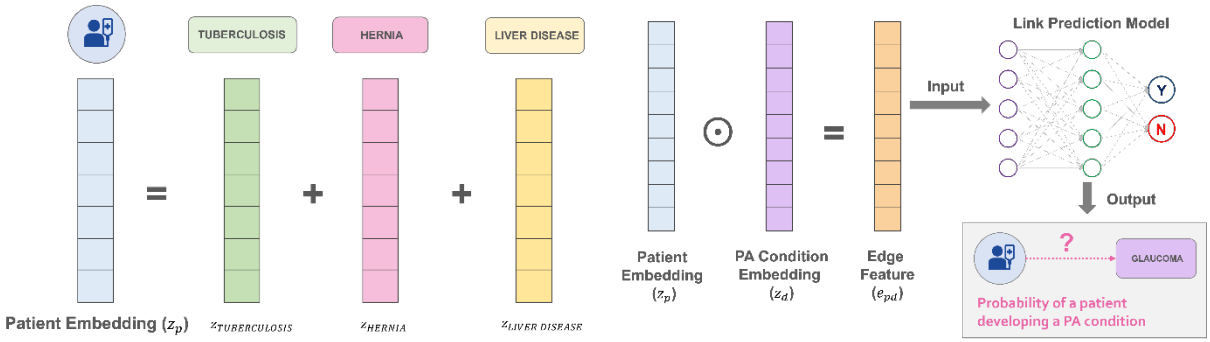
one salient PA condition, e.g., sepsis, delirium, and hypotension, at a time. We take a step forward and attempt to simultaneously assess the occurrence of multiple health conditions since many of the conditions are interrelated to and associated with each other (Wollschlager and Conrad 1988).

We now describe the training procedure, which consists of four consecutive steps that can be modularized and parallelized (Figure 3.4). First, from the ICU database, we identify the POA and PA conditions of each patient based on when the diseases are diagnosed (panel (a) of Figure 3.4). After the POA and PA conditions are identified, they are used to construct a bipartite graph having two types of nodes, i.e., patients and POA/PA conditions, and edges connecting them. Then, the graph embeddings, or vector representations of the nodes, are obtained via graph representation learning (panel (b) of Figure 3.4). However, instead of using the raw results from graph representation learning, we update the patient embeddings using the embeddings for POA conditions. Specifically, we represent each patient as a combination of medical conditions that the patient already had when admitted (panel (c) of Figure 3.4). Then, an ML model for link prediction is trained. The model is used to predict the probability of a patient developing a certain PA condition after admission and make patient-level predictions for unseen patients during the training process (panel (d) of Figure 3.4). In the following subsections, we describe each step in the framework in detail.



(a) POA and PA conditions for each patient are extracted from the ICU database

(b) Node embeddings are obtained from the patient-disease graph



(c) Patient embeddings are computed by combining embeddings of POA conditions

(d) Link prediction model is trained using features combining the two embeddings

Figure 3.4. Training procedure of the proposed framework

3.3.1. Graph Construction and Representation Learning

Before constructing the graph, information regarding whether the individual patient was diagnosed with certain health conditions is extracted from the ICU database. Using the diagnosis records, a bipartite graph having patients and POA/PA conditions as nodes and edges between them is constructed. After a bipartite graph is constructed, a graph representation learning technique is applied to generate embeddings for the nodes. As mentioned, we adapt node embedding methods using random walks with restarts, which have been widely used for various practical applications including healthcare (Yi et al. 2022). Such methods stochastically define the decoder as the probability of traversing consequent nodes. In other words, the similarity between v_i and v_j is defined as the probability of visiting v_j during a random walk starting at v_i:

$$s(v_i, v_j) = p(v_j|v_i) = \frac{e^{\mathbf{z}_i^T \mathbf{z}_j}}{\sum_{v_k \in \mathcal{V}} e^{\mathbf{z}_i^T \mathbf{z}_k}}, \quad (1)$$

Random walk paths are sampled while traversing the nodes according to a certain transition scheme. The simplest sampling scheme would be assigning equal probabilities to the current node's neighbors, i.e., uniform sampling (Perozzi et al. 2014), however more flexible sampling

schemes can be used to better model complex relations between nodes (Grover and Leskovec 2016). Then, a neural network model that projects each node into a latent space is trained while updating the representations by maximizing the probability of identifying neighboring nodes or minimizing the cross-entropy loss defined as:

$$\mathcal{L} = \sum_{(v_k, v_l) \in \mathcal{D}} -\log\big(DEC(z_k, z_l)\big), \tag{2}$$

where $\mathcal{D}$ is the training set containing pairs of nodes sampled during random walks with the node distribution $(v_i, v_j) \sim p(v_j|v_i)$. As a result, we obtain node embeddings corresponding to patients and POA/PA conditions in the graph. Following that, the patient embedding is updated as described in the following subsection.

3.3.2. Fair2Vec: Representing Patients with POA Conditions

In this study, we propose a novel method called Fair2Vec to represent patients in the bipartite graph by updating their node embeddings as a linear combination of POA conditions. Fair2Vec is inspired by natural language processing methods that represent sentences as a linear combination of words (Arora et al. 2017, Wieting et al. 2016). Since words are the basic building blocks of any document, a sentence can be efficiently represented as a unique combination of the words included in the sentence. The simplest method is element-wise, unweighted-averaging (AVG) of word vectors in the sentence (Wieting et al. 2016):

$$z_s = \frac{1}{|S|} \sum_{w \in S} z_w, \tag{3}$$

where z_s is the embedding for sentence *S*, $|S|$ is the number of words in *S,* and z_w is the embedding for word *w*.

It has been demonstrated such a simple method can outperform sophisticated methods including state-of-the-art recurrent neural network architectures that were used to generate

sentence vectors independently (Wieting et al. 2016). However, this does not consider the highly skewed distribution of data, which is also prevalent in natural languages (Powers 1998). Thus, Arora et al., (2017) proposed the Smooth Inverse Frequency (SIF) method which computes sentence embeddings by weighted-averaging word vectors in the sentence:

$$z_s = \frac{1}{|S|}\sum_{w \in S} \frac{a}{a + p(w)} z_w, \tag{4}$$

where $p(w)$ is the estimated frequency of word *w* and *a* is a parameter to control the inverse weighting procedure[5]. The inverse weighting scheme is to penalize very common words that convey little meaning such as *a*, *the*, *where*, and *that* and gives importance to less common words in a corpus that provide a unique semantic feature to each sentence. After calculating the weighted average of word vectors, the first singular vector (u) of the matrix comprising those vectors is removed:

$$z_s := z_s - uu^T z_s \tag{5}$$

This is to remove common factors in most sentences, minimizing redundant information and highlighting distinct information in each sentence. The embedding procedure for SIF is described in detail in Appendix 6.3.

Similarly, we represent a patient node as a combination of POA conditions and use the representation as input to the prediction model. That is, we consider POA conditions that a patient has as *words* and the patient as a *sentence* in natural language. Hence, rather than modifying the random walk scheme or optimization methods, we combine embeddings for POA conditions obtained from standard graph representation learning methods with random walks, e.g., Node2Vec (Grover and Leskovec 2016) and DeepWalk (Perozzi et al. 2014), to generate patient node

[5] It has been reported that setting a to a small value between 0.0001 and 0.001 is optimal for text classification tasks.

embeddings. As a result, patient node embeddings obtained from random walks in the bipartite graph are disregarded and replaced with those from POA condition embeddings.

Due to the weighting scheme penalizing major health conditions, SIF can be useful when we want to compensate for less common conditions. As mentioned, the distribution of ICU data is highly skewed in most cases – uncommon events (e.g., the onset of hypernatremia and hypocalcemia) occur much less frequently than common events (e.g., the onset of acute renal failure and sepsis) and thus have much fewer connecting links in the graph. As a result, they are likely to be underrepresented in the graph representation learning phase, i.e., prevalence bias, which can be counterbalanced by SIF. This could be effective when we want to assign similar importance to collaborative patterns between minor medical conditions to those between major medical conditions.

However, in contrast to the common patterns observed in natural language, common POA conditions such as acute kidney failure and hypertension could be critical in capturing major collaborative patterns since they are important clinical indicators that often precede or coincide with many other conditions. Furthermore, many of them are leading causes of death and intense suffering in ICU, taking a heavy toll on various stakeholders including patients, physicians, and hospitals (Ghosh et al. 2016, Green et al. 2019, Nemati et al. 2018, Sheetrit et al. 2019). In other words, the frequency of words and their significance could have an *inverse* relationship, whereas the significance of a POA condition is *proportional* to its frequency. This can be especially important to capture when we are trying to identify the most common and burdensome diseases from the caretaker's point of view. Therefore, we propose the Smooth Frequency (SF) method to calculate patient embedding (z_p) by combining vector representation of POA conditions (z_c):

$$z_p = \frac{1}{|C|}\sum_{c \in C} \frac{a + p(c)}{a} z_c, \quad (6)$$

where $|C|$ is the number of POA conditions that the patient has and $p(c)$ is the estimated frequency of the condition c. The estimated probabilities $p(c)$ can be computed by normalizing the sampled frequency distribution from random walks or degree distribution in the initial bipartite graph. By inverting the "inverse frequency" calculation again, we assign larger weights to common POA conditions and smaller weights to uncommon ones to effectively capture correlational patterns between health conditions. Finally, similar to SIF, this is followed by removing the first singular vector (u) of the matrix having patient embedding vectors are rows:

$$z_p := z_p - uu^T z_p \quad (7)$$

Appendix 6.3 describes the SF method in detail.

Table 3.1. Comparison of three different techniques in Fair2Vec

	AVG	**SIF**	**SF**
Weighting scheme	Unweighted	Penalizes common instances	Penalizes uncommon instances
Expected effect on prevalence bias (fairness)	Has no effect	Compensates for the minority group	Prioritizes the majority group
Expected effect on predictive performance (accuracy)	Enhances overall accuracy of the model	Enhances accuracy for predicting minor conditions	Enhances accuracy for predicting major conditions

Fair2Vec utilizes three different techniques – i.e., AVG, SIF, and SF – to represent different patients in the graph. As described, different techniques have distinct characteristics, along with disparate effects on the fairness and accuracy of the resulting prediction model (see Table 3.1 for a summary). For instance, since SIF can compensate for the minority group, it can enhance coverage and accuracy for rare medical conditions. Whereas, by prioritizing the majority group, SF can enhance the accuracy of the model in predicting major medical conditions such as hypertension and acute respiratory failure, which are of primary interest to intensivists in most

cases. Thus, Fair2Vec can cater to more diverse application scenarios in outcome prediction by allowing users to choose different schemes depending on the context and end task.

In addition, by disregarding patient node embeddings from random walks, we can circumvent the problem of underrepresenting certain nodes due to the uneven distribution of node connectivity in the bipartite graph. This is a critical problem prevalent in many real-world heterogeneous graphs, e.g., in commercial user-item interaction networks and author-paper academic networks, that should be properly addressed (Dong et al. 2017, Sun et al. 2011). Also, our framework is computationally efficient and scalable since we can use standard node embedding methods (Grover and Leskovec 2016, Perozzi et al. 2014) that scale linearly with the number of nodes in terms of training time ($O(|\mathcal{V}|)$). Besides, applying SF/SIF to obtain patient embeddings is a simple process of linear operations. In contrast, changing the random walk or optimization schemes is likely to increase the computational complexity of the overall algorithm.

Finally, our framework does not require constant re-training and updating embedding vectors when new nodes are added. When we need to make an inference about a new patient who is not included in the original graph, we can simply apply a sentence embedding technique to obtain a corresponding vector representation using the patient's reported POA conditions. This is even more crucial in the context of risk profiling in ICUs since most patients are admitted to the unit unexpectedly and require timely intervention. Decisions must be made quickly after the admission and appropriate measures have to be taken instantaneously to prevent calamitous outcomes. In such situations, it would be inefficient to re-train the model every time a new patient is admitted.

3.3.3. Prediction of PA Conditions

Once the node embeddings of patients and PA conditions are obtained, we can combine them to generate edge features that are used as inputs to the link prediction model. There are several ways to obtain edge features using node features. *Average, Hadamard, weighted-L1,* and *weighted-L2* are four widely used operators to generate edge vectors from corresponding pairs of node vectors (Grover and Leskovec 2016). For node vectors corresponding to two nodes *a* and *b* (z_a, z_b), the edge (e_{ab}) vector representations using these operators are as follows:

$$e_{ab} = \begin{cases} \frac{z_a + z_b}{2} \ (average) \\ z_a \odot z_b \ (Hadamard) \\ |z_a - z_b| \ (weighted - L1) \\ |z_a - z_b|^2 (weighted - L2) \end{cases} \quad (8)$$

When multiple edge features, e.g., those generated by SF and SIF, are used in a single model, they can be combined by concatenating them, which results in a single vector.

After generating edge features, virtually any supervised prediction model such as a decision tree, support vector machine, and neural network can be trained to predict prospective links between patients and PA conditions. In this study, we train a feed-forward neural network to classify pairs of a patient and a PA condition as positive, i.e., with high risk, or negative, i.e., with lower risk. Furthermore, in addition to edge features, other patient-level or disease-level information, e.g., patient demographics, previous medical history, and text descriptions of diagnoses, can be included as input features to the model after pre-processing if available and appropriate.

3.4. Experimental Results

3.4.1. Data

In this study, we used the eICU Collaborative Research Database (Pollard et al. 2018), a

publicly available database from the eICU Telehealth program of Philips Healthcare. The dataset contains de-identified information of 200,859 patient records admitted between 2014 and 2015 in 335 ICU units at 208 hospitals located throughout the US. Following the APACHE-IV protocol (Zimmerman et al. 2006), we set the observation period (t_0) to 24 hours after admission and the prediction period (t_1) after that. That is, we classified the recorded diagnoses into two groups – (1) those reported within 24 hours after admission (POA conditions), and (2) those reported after that (PA conditions). In addition, we only included diagnosis records that are included in the International Classification of Diseases-10 (ICD-10) code system, which is a comprehensive list widely used for medical classification. ICD-10 is developed by the World Health Organization and is mandated by the US government (Manchikanti et al. 2013).

Among 200,859 patient visits, we only considered patient records having at least three POA and PA conditions since such patients are often prioritized by clinicians due to the severity of symptoms. Then, we randomly partitioned the remaining patient visits into training and test sets in a 70/30 ratio. As a result, the training set and the test set contain 16,110 and 6,905 patient diagnosis records, respectively. Overall, there are 752 unique POA conditions and training data includes 719 and test data includes 649 of them. There are slightly more unique PA conditions – 750 and 673 of them in training and testing data, respectively, and 783 in total (Table 6.3 in Appendix 6.4). After the data is processed, we constructed a bipartite graph with the diagnosis records following the graph construction scheme outlined in the previous section. The graph comprises 32,970 patients and 3,181 unique medical conditions, in addition to 228,020 edges between them. The average degree of all nodes in the network is around 14, however, patient nodes show a significantly lower average degree (7.2533) compared to the other nodes (214.2939) (Table 6.4 in Appendix 6.4). This indicates that patient nodes are highly likely to be underrepresented

during the sampling and optimization process due to a small number of connections.

3.4.2. Graph Representation Learning and Link Prediction

We first generated node embeddings for diseases using the Node2Vec algorithm (Grover and Leskovec 2016), which is an established method for general-purpose graph embedding. Node2Vec is widely used for various inductive tasks on a graph including node classification, link prediction, and graph clustering. It follows the standard node embedding procedure of (1) sampling sequences of nodes while traversing the nodes with a given random walk scheme, and (2) unsupervised learning of node-level latent features using a shallow neural network architecture, e.g., Skip-gram (Mikolov et al. 2013). In sampling node sequences, Node2Vec performs biased random walks by combining the elements of breadth-first search and depth-first search, thereby combining global and localized views of the graph. Such a biased transition scheme is reported to result in improved performances compared to uniform sampling in several prediction tasks (Grover and Leskovec 2016, Hamilton et al. 2017b). We describe the optimal hyperparameters used for the embedding procedure in Appendix 6.5.

After the node embeddings are learned, the patient embeddings are updated using the three techniques in Fair2Vec (i.e., AVG, SIF, and SF). Then, edge features for the link prediction task are generated by combining feature vectors of respective node pairs. In doing so, we used the Hadamard operator, which showed overall superior performance to other edge feature generation methods described earlier. Finally, we trained a supervised model to predict the probability of edge construction using generated features. We chose feedforward neural networks with back-propagation, a universal function approximator for tabular features (Goodfellow et al. 2016), for link prediction. Detailed hyperparameter settings that were found after experiments are described in Appendix 6.5. We compare the predictive performances of different models after training in the following section.

3.4.3. Evaluation

We extensively assess the performance of prediction models in our framework by comparing them with strong baselines. We consider a wide variety of baselines; we include state-of-the-art methods in graph representation learning, along with deep learning models using variables for an established severity scoring system for ICU outcome prediction. We list and describe the different methods for benchmarking as follows:

- **Deep learning with tabular features:** Feedforward neural networks trained with input features used for the APACHE-IV scoring system. We present results with two types of deep learning models trained with a different set of variables – (1) variables used to calculate the acute physiology score (APS) and (2) variables used to predict patient outcomes, i.e., patient mortality and length of stay (APACHE). The former is the subset of the latter, which includes all input variables for the APACHE-IV system.
- **Graph embedding with random walks**: Widely used unsupervised methods for various downstream tasks including link prediction and node classification in graphs – Node2Vec (Grover and Leskovec 2016), DeepWalk (Perozzi et al. 2014), and Bipartite Network Embedding (BiNE) (Gao et al. 2018). The first two are generic methods for graph embedding while BiNE is specialized for bipartite graph embedding having different types of nodes. After learning the embeddings, two-layer feedforward neural networks are trained using the edge features.
- **Graph embedding with neighborhood aggregation algorithms**: Graph embedding models based on three widely used neighborhood aggregation algorithms, i.e., Graph Convolutional Network (GCN) (Kipf and Welling 2017), GraphSAGE (Hamilton et al. 2017a), and Graph Attention Network (GAT) (Veličković et al. 2018). The embeddings are updated by

aggregating information from each node's neighbors in a semi-supervised manner. After learning the embeddings, two-layer feedforward neural networks are trained using the edge features.

- **Fair2Vec (AVG, SIF, and SF)** (ours): Prediction models developed using our proposed framework, which generate patient embeddings with AVG, SIF, or SF. After learning the embeddings, two-layer feedforward neural networks are trained using the edge features.

Since the specific requirements differ depending on the context, we rigorously evaluate the models developed using our proposed framework on four plausible scenarios that the ICU outcome prediction system based on the method that could be utilized for real-world clinical decision support:

- **Predicting the occurrence of major conditions**: Although there are more than 700 unique medical conditions in the data, the distribution is highly skewed. That is, a few most prevalent events such as the onset of sepsis and hypotension occur much more frequently compared to others. Moreover, most of them take a heavy toll on patients and the hospital, causing various complications and a number of fatalities, and thus are of primary interest to ICU clinicians (Hall et al. 2013).
- **Predicting the occurrence of uncommon conditions**: Even though a handful of major events are the leading causes of death and intense suffering in the ICU, managing uncommon diseases might be important in some cases. Since the patient profile is highly heterogeneous in many cases, relatively infrequent and less burdensome diseases can be fatal for a specific patient population, calling for additional attention from caregivers. For example, minor bone fractures, which are usually not associated with death, had substantially increased mortality for patients older than 75 years (Bliuc et al. 2009).

- **Predicting the occurrence of all recorded conditions**: Although relatively unusual, healthcare experts might want to predict the risk of most recorded medical conditions in ICUs. In addition, this might provide a more comprehensive and global view of the problem by covering a larger pool of potential conditions, providing insights to practitioners.
- **Identifying conditions that each patient is likely to develop**: Prioritizing medical conditions with a high likelihood of occurrence for each patient is a critical step in providing personalized treatment. This can significantly reduce the burden of diagnosing and caretaking for ICU clinicians.

For the first task, i.e., predicting major diseases, we focused on predicting the 30 most prevalent diseases in the database. They account for over 35% of total disease occurrences in the data and include leading causes of in-hospital deaths such as respiratory failure, sepsis, and pneumonia (Hall et al. 2013). In contrast, for the second task, i.e., predicting uncommon events, we focused on predicting less prevalent medical conditions in the database. We first excluded very rare conditions that occurred less than 200 times, which means that less than 1% of the patients were diagnosed with them. After that, we selected 30 diseases that occurred least frequently after admission. We provide a complete list of the most and least prevalent conditions in Table 6.9 of Appendix 6.6.

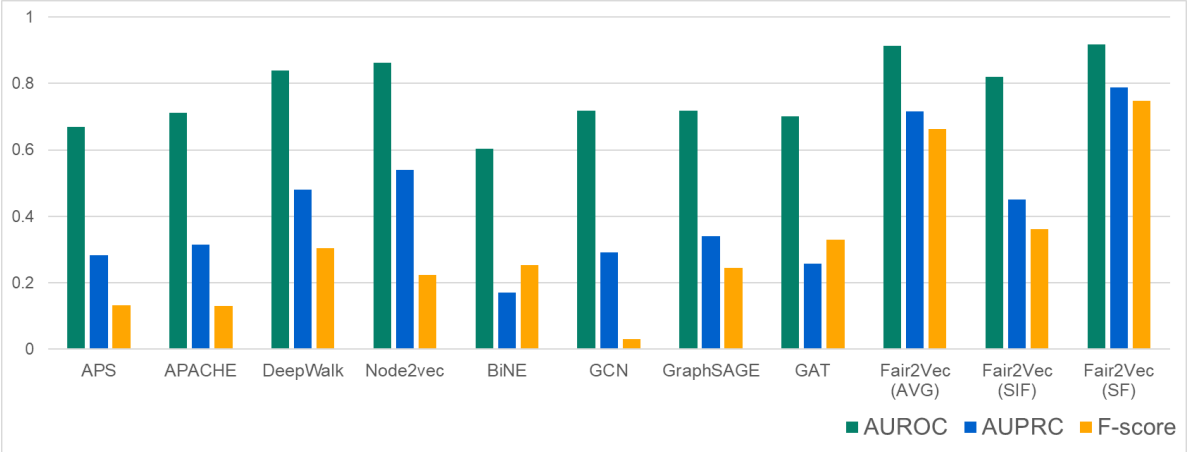


Figure 3.5. Summary of evaluation results for 30 most frequent conditions

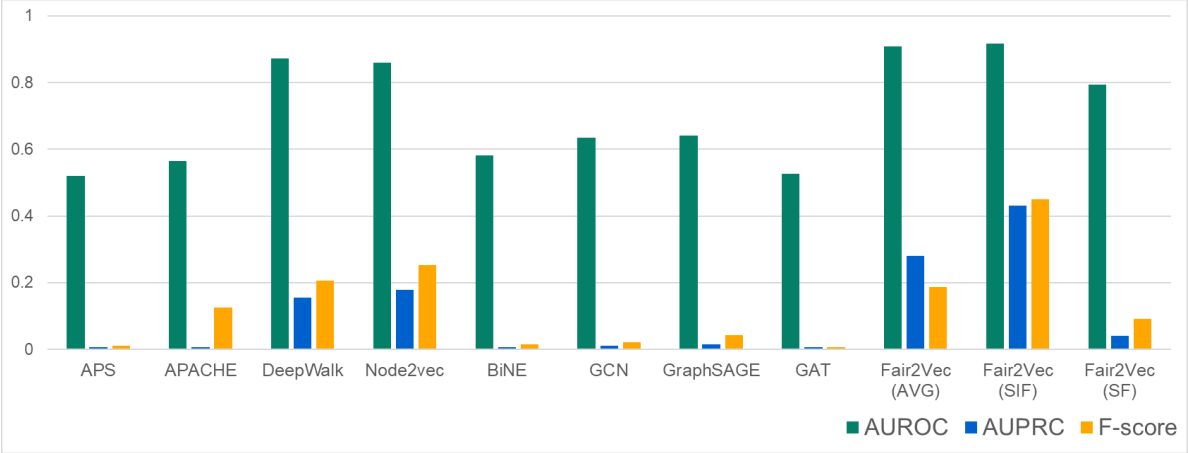


Figure 3.6. Summary of evaluation results for 30 least frequent conditions

In this section, we describe the evaluation results compared to the baselines described earlier. Figure 3.5 and Figure 3.6 summarize the evaluation result of the first and the second tasks, respectively. The first two problems can be defined as a binary classification task, i.e., classifying each patient-PA condition as positive (likely to have an edge between) or negative (unlikely to have an edge between). To this end, we use three metrics that are widely used to evaluate binary classifiers – area under the receiver operating characteristics curve (AUROC), area under the precision-recall curve (AUPRC), and F-score (Powers 2011, Saito and Rehmsmeier 2015).

Overall, graph representation learning methods outperformed standard deep learning models trained with tabular features by a significant margin, demonstrating their ability to effectively model complex relations between patients and medical conditions. Among them, models trained using our proposed framework outperform other methods in graph representation learning. It is noteworthy that Fair2Vec is more effective than bipartite graph embedding (BiNE) and state-of-the-art neighborhood aggregation methods (GraphSAGE, GCN, GAT) in learning and predicting relations between patients and medical conditions.

However, as expected, SF is more effective when predicting prevalent medical conditions compared to SIF whereas SIF is more effective than SF in predicting the occurrence of uncommon ones. A primary reason for this finding would be that since SF penalizes uncommon diseases when representing patients, the link prediction model is unable to identify the occurrence of such conditions. In contrast, since SIF assigns high weights to uncommon conditions, it is better at predicting the occurrence of uncommon PA conditions at the price of worse performance in predicting common PA conditions. Thus, there exists a trade-off in how to combine the embeddings of POA conditions, i.e., penalizing the common or uncommon diseases. Since it has a considerable impact on the prediction performance in different tasks, practitioners should consider the characteristics of the data and their ultimate objective of developing the HPA system when choosing either SIF or SF to represent patient nodes.

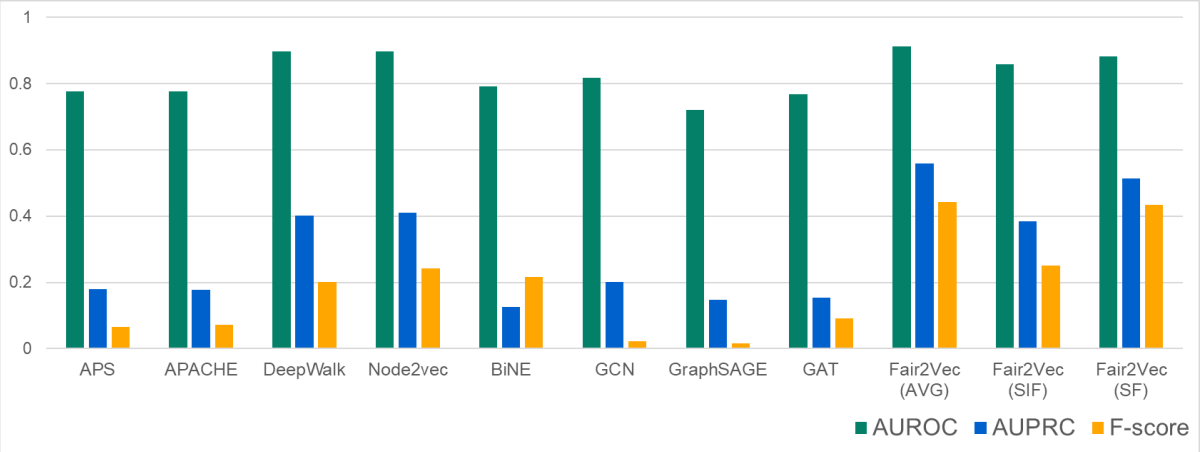


Figure 3.7. Summary of evaluation results for all recorded conditions

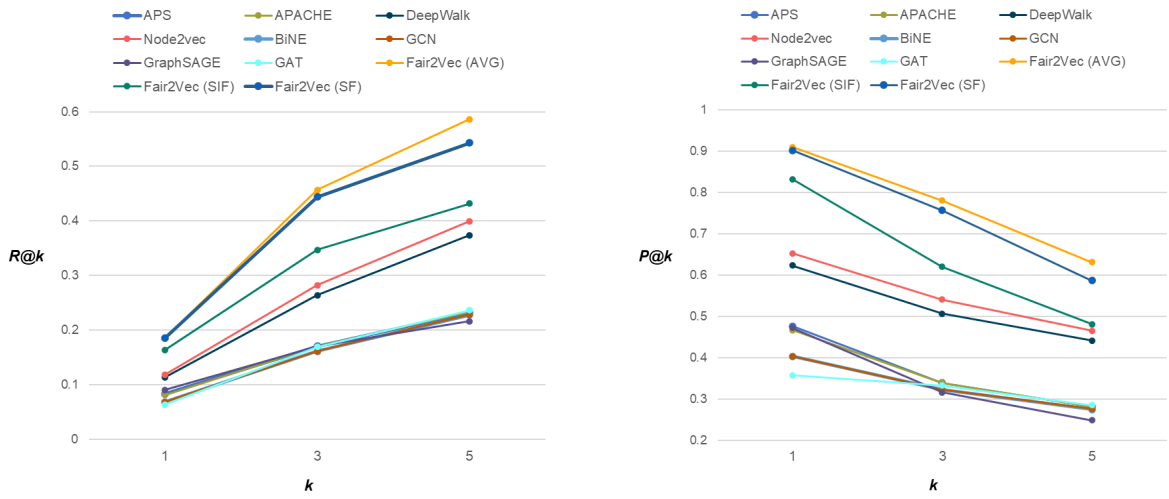
Evaluation results for the third task of predicting all medical conditions are summarized in Figure 3.7. Here, the model combining SF and graph embedding shows overall improved performance over existing methods. In addition, predicting uncommon medical conditions or all recorded conditions is a more difficult task compared to predicting common conditions, which is also noted in prior studies (e.g., Leisman, 2018). Since most patients have not developed uncommon conditions, the prediction model does not have sufficient information to identify them. That is, the model is likely to suffer from a class imbalance problem when predicting rare diseases. Nonetheless, our model demonstrates considerably improved performance compared to existing methods in such tasks as well, showing promising potential.

The last task can be formulated as a ranking problem, i.e., identifying PA conditions that are expected to be developed by each patient with the highest probabilities. Hence, we report two metrics commonly used to evaluate a ranking system, which are precision at k (P@k) and recall at k (R@k). The two are defined as follows:

$$\text{P@}k = \frac{\#\ of\ relevant\ items\ among\ top-k\ items}{k} \tag{13}$$

$$\mathrm{R@}k \;=\; \frac{\#\ of\ relevant\ items\ among\ top-k\ items}{\#\ of\ all\ relevant\ items} \tag{14}$$

Thus, the two metrics can be regarded as individualized precision and recall scores, i.e., the proportion of true positives ("hits") among positively classified cases and the proportion of hits among real positive classes for each individual. These metrics show how well the prediction model performs in terms of identifying relevant items, i.e., medical conditions with higher probabilities, for each instance, i.e., a patient.



(a) *R@k* with different values for k (b) *P@k* with different values for k

Figure 3.8. Summary of evaluation results for the ranking task

We report the evaluation results of the ranking task while setting the number of suggestions (*k*) to 1, 3, and 5 (Figure 3.8). That is, we made top-*k* suggestions to each patient after excluding rare PA conditions that occurred less than 200 times, and calculated *P@k* and *R@k* for each *k*. Overall, we again find that our models which combine Node2Vec and sentence embedding methods significantly outperform the other baselines. Thus, we argue that our framework is effective in not only predicting the occurrence of major/minor medical conditions but also

shortlisting a set of conditions that each patient is highly likely to develop for providing personalized treatment and care.

3.4.4. Fairness of Predictions

In addition to predictive performance, we assessed the fairness of the prediction models. We used two metrics that quantify bias arising from skewed distribution in training data – i.e., the popularity-rank correlation for users (PRU) and popularity-rank correlation for items (PRI) (Zhu et al. 2021). PRU measures user-side popularity bias, which makes a model assign a higher ranking to a popular item compared to an unpopular one, even though a user likes both. To calculate the PRU of a given model, the Spearman's rank correlation coefficient between the popularity and ranking positions of test items is averaged across all users:

$$PRU = -\frac{1}{N}\sum_{u\in\mathcal{U}} SRC\left(pop\left(\tilde{O}_u^+\right), rank\left(\tilde{O}_u^+\right)\right), \tag{15}$$

where $SRC(\cdot,\cdot)$ calculates Spearman's rank correlation, $pop(\tilde{O}_u^+)$ is the popularity of test items (measured in the number of occurrences of all items), and $rank\left(\tilde{O}_u^+\right)$ is the rankings of test items for user u by the prediction model. In contrast, PRI quantifies item-side popularity bias, which leads to a low opportunity for unpopular items to be recommended. For PRI, the Spearman's rank correlation coefficient between item popularity and their average ranking positions over all users is calculated:

$$PRU = -SRC\left(pop(\mathcal{I}), avg\_rank(\mathcal{I})\right), \tag{16}$$

where $pop(\mathcal{I})$ is the popularity scores of all items $\mathcal{I}$ and $avg\_rank(i)$ is the average ranking for item i over the set of matched users. Both PRI and PRU result in values in the range [-1, 1], and a large positive value implies a high level of user/item-side bias. Although the two metrics are closely related and often show similar patterns, they measure different aspects of prevalence bias

(Zhu et al. 2021). Therefore, we report both metrics to provide a complete picture of algorithmic fairness across different models.

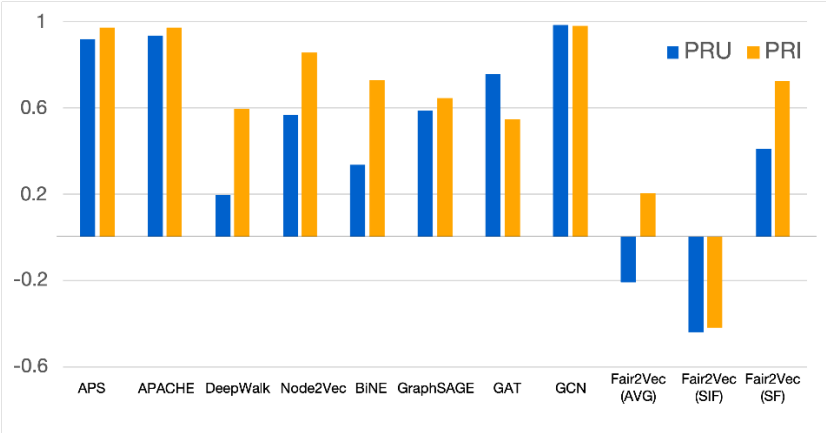


Figure 3.9. Summary of evaluation results (fairness)

Figure 3.9 is the summary of evaluation results in terms of the fairness of predictions. Overall, most existing methods show a high level of bias from both user and item perspectives. For instance, APS, APACHE, and GCN show both PRU and PRI values close to 1, which means that these methods are extremely partial to common medical conditions. Among the three variants of Fair2Vec, SF shows the highest prevalence bias and SIF shows the lowest prevalence bias, as expected. All in all, these results clearly show a trade-off between various objectives. That is, algorithmic fairness can be enhanced by compensating for uncommon medical conditions, however, it can result in lower accuracy for common medical conditions.

3.5. Discussions and Conclusion

3.5.1. Research Contributions

This study contributes to several major research streams in IS including *design science*, *health IS,* and *predictive analytics*. Our study contributes to the IS knowledge base by proposing a novel design artifact that can be used to support patient care in ICU. Specifically, since the key premise

of computational design science research is impact and relevance (Rai 2017), we attempt to tackle a relevant and timely problem and rigorously evaluate the implemented artifact with real-world data. In doing so, we derive implications for healthcare and predictive analytics, enhancing the significance of the study in managerial and business aspects, which is a key requirement of a design artifact (Hevner et al. 2004).

Recent developments and advancements in information technology such as sensors, wearables, and social media have opened up new opportunities to explore previously unresolved complex problems in healthcare. A wide variety of available big data, e.g., electronic health records and spatiotemporal data, and related technologies, e.g., graphics processing units, cloud computing, and parallel data processing, are enabling health IS researchers to explore and evaluate new ideas and prototypes that were inconceivable previously. Although there are recent IS studies utilizing cutting-edge computational techniques for healthcare (e.g., Lin et al. 2017; Zhang and Ram 2020) there have been relatively scant attempts to analyze interdependencies between multiple types of entities in complex systems. The graph analytics framework proposed in this study can be applied to model various relationships between entities in complex systems commonly observed in healthcare and other areas. Our framework can be easily extended to heterogeneous information networks containing more than two entity types. For instance, relationships between patients, mobile devices, and caretakers can be modeled as a tripartite graph using a similar graph representation learning framework.

Finally, our study makes contributions to predictive analytics and business intelligence, which is an emerging field in empirical IS research. With climactic developments in related techniques such as machine learning, statistical modeling, and computational linguistics, recent work in predictive analytics and business intelligence (e.g., Abbasi et al. 2010; Lin et al. 2017;

Zhang and Ram 2020) are changing the landscape of empirical IS research by presenting novel, interdisciplinary solutions to relevant problems. Our study contributes to the literature by combining state-of-the-art methods in emerging fields in data science, i.e., graph representation learning and text analytics, to tackle an important problem with high impact and relevance, i.e., outcome prediction in critical care.

3.5.2. Practical Implications

Our study has practical implications for diverse stakeholders. First and foremost, it can enhance the quality of care and safety of ICU patients. Our predictive modeling framework can be used to accurately predict the prognosis of the patients and used to improve the quality of treatment they are receiving. For instance, patient-centered policies and customized treatment plans can be developed based on the analysis results. Furthermore, it can be used to assist healthcare experts in ICUs, improving their performance and efficiency. Many ICUs clinicians and caregivers experience difficulties in attending to patients due to understaffing and information overload (Hyland et al. 2020). Our framework can be an effective tool in reducing the cognitive and mental burden of clinicians with accurate and timely predictions regarding patients' health conditions. Finally, it can enhance the overall efficiency of the hospital by improving how key resources are allocated and managed in ICUs, which take a heavy toll on many hospitals.

3.5.3. Limitations and Future Work

Our study has a few limitations and related future research directions. First, other various types of information in the ICU database and electronic health records such as drug intake records, allergies, and treatment records can be utilized when available. In doing so, extending the bipartite graph into a heterogeneous information network and applying graph representation learning specialized for heterogeneous graphs (e.g., Dong et al. 2017) may be considered. Similarly, node-level features such as patient demographics and previous medical history may be included if

needed. Although we eschew them in this study to enable agile prediction, such information can be useful and readily available in some cases. Finally, we believe that our framework can be extended and generalized to other graph representation learning tasks, e.g., item recommendations in e-commerce and keyword suggestions in search engines. Future work can investigate how the proposed framework can be adapted for other predictive modeling applications using graph-based approaches.

4. ESSAY 3: Graph Representation Learning for Reciprocal Recommendation in Online Matching Platforms: A Social Relations Model Approach

4.1. Introduction

As consumer choices become increasingly complex and technology continues to advance, recommender systems have become essential components of various online services, offering benefits to both consumers and service providers (Lee et al., 2020; Lee & Hosanagar, 2019). Among various types of recommender systems, reciprocal recommender systems (RRSs) have significantly grown in popularity over the past decade as a vital instrument for connecting users on online matching platforms, serving various purposes such as professional networking and career development (e.g., LinkedIn), matchmaking (e.g., Tinder), knowledge sharing (e.g., Quora), and the gig economy (e.g., Uber Eats and Instacart) (Li et al. 2017, Palomares et al. 2021).

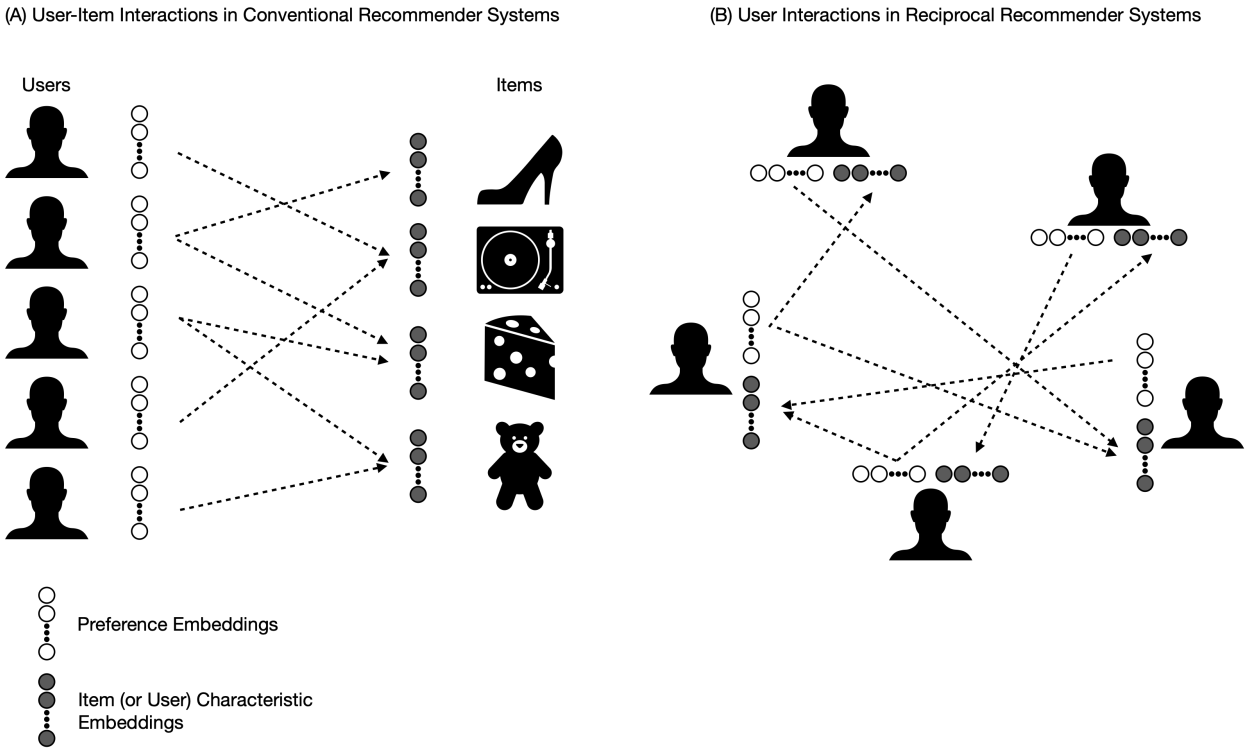


Figure 4.1. Comparison of user-item interactions in conventional and reciprocal recommender systems

RRSs differ from traditional recommender systems in a few ways. First, users play the dual roles of "users" and "items" simultaneously (Figure 4.1). That is, a user serves the role of an item being recommended to other users in addition to his/her original role as a person choosing an item (Palomares et al. 2021). For example, in the context of online dating, individuals make decisions on who they would like to date while being suggested as potential candidates to other users. Although there have been many recent developments in RRSs owing to the advancements in data mining methods, only scant attention has been given to the theoretical underpinnings behind the user's social interactions and the dual roles that users play in online matching platforms. Extant RRS methods either do not consider the dual roles at all (such as in Pizzato et al. 2010) or train independent models for the dual roles and fail to capture their dynamic interactions and common patterns (e.g., Neve & Palomares, 2019).

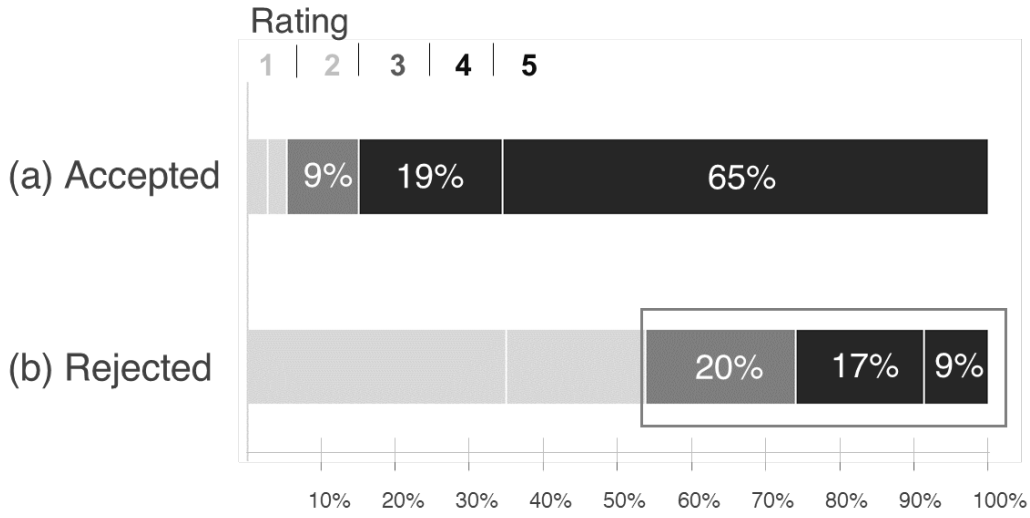


Figure 4.2. Comparison of rating distribution between accepted and rejected suggestions

In addition, users are likely to engage in multiple decision stages, with potentially different rules at each, to effectively eliminate large swaths of available options (Bruch et al. 2016, Jung et al. 2022). For instance, based on our exploratory analysis of data from a real-world online dating app, ratings (i.e., initial screening of potential candidates) and accepting the suggestions (i.e.,

subsequent decision-making) are not perfectly aligned, suggesting different criteria are involved with the two distinct stages of decision making (Figure 4.2). While the vast majority of accepted suggestions have received positive ratings (Figure 4.2 – (a)), almost half of the rejected suggestions received either neutral (i.e., 3) or positive ratings (i.e., 4 or 5) (Figure 4.2 - (b)). That is, rejections do not always imply that users strongly dislike the suggested user. This finding supports the claim that users go through a multi-stage, complex decision-making process when choosing mates in online dating (Bruch et al. 2016).

Existing RRS studies, while providing methodological foundations, lack a comprehensive understanding of the intricate, dyadic interactions between individuals assuming dual roles and participating in multistage decision-making processes within online matching platforms. To address these limitations, we present a novel graph representation learning framework for reciprocal recommendations called MatchSAGE. Drawing on the social relations model (SRM), MatchSAGE captures the distinct, yet interrelated, roles of users by simultaneously learning their dual embeddings (as perceivers and as targets) within a single graph neural network (GNN) architecture. The two embeddings are concurrently inferred using a shared set of parameters that capture distinct but correlated behavioral patterns of the dual roles. Moreover, our approach considers the multi-stage decision-making process of users in online matching platforms by initializing the GNN embeddings with weights from a rating prediction model. In other words, our method learns users' initial screening preferences using rating data through matrix factorization (MF) and employs the results for initializing GNN parameters for matching preference prediction. Additionally, we implement a deep neural network (DNN) model with fine-tuning tensor operations to capture dynamic interactions between different roles and users.

Table 4.1 highlights the advantage of MatchSAGE in capturing RRS-relevant features of online matching platforms, contrasting it with existing methods in the reciprocal recommendation literature. This will be discussed in greater detail in a subsequent section. To assess the applicability of our approach, we conducted both offline and online experiments using data from a real-world mobile dating app. In the offline experiment, our model significantly outperformed state-of-the-art RRSs in both classification and ranking tasks. Additionally, an ablation study demonstrated that incorporating two types of embeddings (i.e., perceiver and target embeddings) into a single GNN architecture considerably improved performance, reinforcing our argument regarding users' different roles based on the context. Furthermore, our method of initializing GNN with rating (or initial screening) preference embeddings yielded superior results compared to alternative initialization techniques, corroborating our claim about the importance of considering multistage decision-making in online matching platforms. Moreover, during the online field experiment, where we deployed our framework in a real-world environment, the results demonstrated its potential for enhancing both firm's revenue and user experience. Notably, our model exhibited the potential to increase the firm's revenue by 21%.

Table 4.1. Comparison between extant RRSs and MatchSAGE

Method	**(1) Impression Management**	**(2) Multi-stage Decision Making**	**(3) Dual Roles of Users**	**(4) Dynamic Interaction Between Different Roles and Users**
Content-based filtering (e.g., Pizzato et al., 2010)	X	X	X	X
Memory-based collaborative filtering (e.g., Xia et al., 2015)	O	X	X	X
Model-based collaborative filtering (e.g., Neve & Palomares, 2019)	O	X	△	X
MatchSAGE	O	O	O	O

*__Note__: “O” indicates the method addresses the feature, “X” indicates the method does not address the feature, and “Δ” indicates the method partially addresses the feature.

The contributions of this study are manifold: (i) we introduce a novel graph representation learning framework for reciprocal recommendation tasks, reflecting complex behavior patterns in online matching platforms, (ii) we ground our framework in behavioral theories that elucidate human decision-making processes and offer practical design guidelines, and (iii) we demonstrate the superior performance of our framework through both offline and online experiments, potentially benefitting both businesses and end users. The remainder of the paper is organized as follows. In the next section, we outline the theoretical background of our study, along with related work in RRSs. Following that, we discuss our proposed framework and the experimental settings for evaluation and present the results from both offline and online experiments. We conclude the paper by highlighting the practical and theoretical implications of our study.

4.2. Related Literature

4.2.1. Theoretical Background

In this section, we describe the theoretical background of this study, along with the unique features of online matching platforms that require special attention in designing RRS. Matching is an important concept extensively studied in various economic contexts such as job formation and employment (Mortensen and Pissarides 1994), allocation of bank loans (Den Haan et al. 2003), and seller-buyer matching (Kiyotaki and Wright 1993). Thus, previous work in economics provides a solid foundation in the overall understanding of generic models that shape representative types of matching markets such as labor markets (Petrongolo and Pissarides 2001). Nevertheless, they can be limited in understanding and interpreting people’s behaviors in online matching platforms, which differ from those in classical matching markets. For instance, in online

matching platforms, the search size for each individual is significantly larger than in offline settings. This leads to various issues such as choice overload (Jung et al. 2022) and a high level of impression management (Ellison et al. 2006), which complicate the searching and matching processes as a result.

Thus, we rely on behavioral theories to facilitate our understanding of people's social interactions in online matching platforms. SRM suggests three essential components which collaborate in forming an interpersonal perception commonly followed by social behaviors: *perceivers* (also called *receivers*), *targets* (also called *senders*), and *relationships* between them (Back and Kenny 2010). Perceivers have innate characteristics and tendencies (e.g., perceptual biases and leniency), which strongly affect their perception of targets, i.e., the perceiver effect. At the same time, the targets' characteristics and behaviors (e.g., meta-perception and self-expressive behaviors) influence the perceivers' perception, i.e., the target effect. While there are general tendencies that the perceiver and target effect influence the development of interpersonal perception, we may observe some variances of these tendencies when a particular perceiver and target are paired together, also known as the relationship effect. Therefore, to accurately analyze user interaction patterns in online matching platforms and obtain insights from them, the three main effects (i.e., perceiver, target, and relationship effects) should be considered simultaneously. SRM has been extensively used to investigate various types of dyadic relationships, e.g., between family members (Eichelsheim et al. 2009), households (Koster and Leckie 2014), and humans and pets (Kwan et al. 2008). Following this stream of literature, we adopt SRM to understand interpersonal interactions in online matching platforms and to inform the design of our proposed framework.

However, there are several unique characteristics of online matching platforms that should be considered when adopting SRM for the development of an RRS (Liao et al. 2018). First, from the perceiver's perspective, the size of the search space in online matching platforms is massive compared to that of offline social networks (Bapna et al. 2016). Hence, in general, users go through a complex choice process when making a decision (Bruch et al. 2016). To elaborate, in offline social networks, the number of people an individual can interact with is highly bounded by geographical and social factors. In contrast, users in online matching platforms are not hindered by such obstacles, allowing them to interact with an unprecedented number and variety of individuals. For example, in offline dating, possible mates are restricted to a small pool of people in a limited social circle, whereas, in online dating, people can connect to virtually anyone on the platform.

To mitigate the inundation of possible choices, online platforms often provide several functions and tools that help individuals narrow down possible options and effectively make a final decision (Bruch et al. 2016, Jung et al. 2022). For instance, on LinkedIn.com, recruiters can reduce the number of candidate applicants using a tag function that allows recruiters to classify candidates into different subgroups (e.g., "for interviews", "waitlist", and "reject"). Similarly, many online dating sites let users browse and rate other users' profiles. Then, users can see those whom they have highly rated (i.e., narrowed-down options) and choose to send messages to a subset of them (i.e., potential matches). As a result, the decision-making process of users in online matching platforms considerably differs from that in offline social networks.

On the other hand, from the target's perspective, there are an enormous number of competitors craving the perceiver's attention. Thus, impression management, or self-presentation, is becoming increasingly commonplace in online matching platforms (Roulin and Levashina 2016).

For example, in online dating, users often add touched-up photos of themselves or idealized self-descriptions in addition to basic profile information (e.g., gender, age, and height). However, such polished self-representation of targets may often impede perceivers from accurately evaluating the targets. That is, users' self-expression in online matching platforms tends to be more malleable than offline relations in that people have better control of both unintentional cues (e.g., non-verbal linguistic cues or facial expressions) and intentional cues (e.g., personal information, description, etc.) of their true selves. As a result, users can shape romanticized portraits of themselves with the help of technology (Ellison et al. 2006).

Lastly, in online matching platforms, a user simultaneously plays the dual roles of a perceiver and a target (Jung et al. 2022, Neve and Palomares 2019, Palomares et al. 2021). For example, in an online professional networking platform, the same user can often be both a recruiter and a prospective job seeker. In contrast, it is unlikely for a single individual to play dual roles concurrently in an offline setting due to institutional, social, and technical restrictions. When examining a user playing dual roles, extra care must be taken because distinct sets of criteria can be associated with different roles (Tong et al. 2020). For example, as a recruiter, a user may emphasize work ethics and diligence, but the same person can prioritize work/life balance over other factors when looking for a job. Therefore, it is crucial to understand the dual nature of people's behavior in online matching platforms.

In summary, it is essential to take into account these unique characteristics of online matching platforms before developing RRSs. Nevertheless, to our knowledge, extant approaches in reciprocal recommendation have shed little light on such characteristics. In the following section, we review previous RRS literature, identify gaps for improvement, and suggest a new approach that can close the gap from our unique perspective.

4.2.2. Reciprocal Recommender Systems

RRSs, also known as people-to-people recommender systems, aim to discover hidden connections between users and connect each other in online matching platforms (Palomares et al. 2021). In conventional recommendation problems, users make decisions on items, not vice versa. In contrast, in RRSs, choices are bilateral in that a user chooses other users, and the other users choose the user as well (i.e., users are equivalent to items in traditional recommender systems). Prior to describing our method, we review existing work in RRS methodology (see Table 4.2 for a summary). Initial attempts to design RRSs have primarily relied on content-based filtering, i.e., analyzing user-level data such as profile information and self-descriptions. For instance, Pizzato et al. (2010) proposed an algorithm to predict preferences and compatibility between users using their profile attributes. However, as mentioned above, people's self-expressions in online matching platforms are often unreliable because they tend to present idealized portraits of themselves instead of their true selves (Ellison et al. 2006). Therefore, relying on users' self-descriptive data exclusively may lead to suboptimally trained models, resulting in poorly performing recommender systems in practice.

To mitigate such pitfalls, collaborative filtering techniques, which exploit historical user interaction records (Koren et al. 2009), have been extensively adopted for RRS. For instance, Xia et al., (2015) proposed the Reciprocal Collaborative Filtering method, which uses graph-based similarity functions to assess user preferences. Recently, researchers are increasingly applying machine learning-based methods to implement a large-scale RRS in practice. Machine learning methods are highly capable of capturing diverse relations between entities based on the behavioral patterns of individuals in online social networks (Hamilton et al. 2017c, Koren et al. 2009). Furthermore, since they do not rely on user-level data, they are less susceptible to the aforementioned issue of self-embellishment of user profiles. Thus, in this study, we focus on

machine learning-based methods, which show state-of-the-art results in reciprocal recommendations - e.g., probabilistic matrix factorization (Kutty et al. 2012, Neve and Palomares 2019) and GNNs (Chang et al. 2022, Luo et al. 2020).

Table 4.2. Summary of existing work in RRS methodology

Category	**Methodology**	**Representative work**
Content-based filtering	Compatibility scores between users based on profile information	Pizzato et al. 2010
Collaborative filtering	Memory-based filtering (similarity functions based on user interaction records)	Akehurst 2011, Kleinerman et al. 2018, Xia et al. 2015
	Machine learning-based prediction models	Kutty et al. 2012, Neve and Palomares 2019

Nevertheless, most extant methods have neglected the unique features of online matching platforms that were described earlier. For instance, most extant methods combine user (i.e., perceiver) and item (i.e., target) embeddings after separately learning them from distinct machine-learning models (Kleinerman et al. 2018, Luo et al. 2020, Neve and Palomares 2019). From the SRM perspective, since the features of perceivers and targets are learned independently, they are limited in terms of reflecting the reciprocal nature of RRS. On top of that, most of them have failed to consider the complex choice process of users in online matching platforms (Bruch et al. 2016). In contrast, our proposed framework in this study captures the unique aspects of online matching platforms from the SRM and decision theory perspectives (Back and Kenny 2010, Bruch et al. 2016). We describe the proposed MatchSAGE framework in detail in the following section.

4.3. Proposed Method

4.3.1. Overview

In this section, we describe the proposed MatchSAGE framework. Although we present our framework in the context of online dating, a prototypical RSS context, it can be generalized to various contexts such as job-matching portals, knowledge-sharing platforms, and the gig economy.

MatchSAGE consists of three major components – (1) MF-based rating prediction model, (2) GNN-based matching preference prediction model, and (3) DNN architecture for matching probability calculation. We describe each component in detail in subsequent subsections.

Our framework is distinct from extant RRS methods in a few ways. First, our framework learns two different graph embeddings of a user (i.e., dual embeddings) in a unified GNN framework; one for representing a user's role as a perceiver and the other as a target, allowing us to distinguish the personal characteristics reflected in playing the two distinct roles. Particularly, our method jointly updates the dual embeddings of a user during training to learn not only distinct but also correlated patterns of the two distinct roles as a perceiver and as a target. In addition, when predicting the likelihood of matching between two users, we utilize a novel DNN architecture to identify the dynamic, multifaceted interaction between them. In DNN, instead of simply pairing the user embeddings to calculate the probability of the perceiver accepting the target, we propose using fine-tuning layers to capture complex interactions between the perceiver and the target. Consequently, this enables effective modeling of interpersonal perception between various users, i.e., the relationship effect.

To reflect the multi-stage decision process of users, we design a sequential pipeline of machine learning. As stated earlier, online matching platforms provide users with tools to help them mitigate the inundation of possible options and narrow down choice options (e.g., tagging in LinkedIn.com). In our experimental setting, a user first rates others so that later they can make a final choice among those who are highly rated. This indicates that rating and matching records represent different stages of people's decision-making. Some previous studies have incorporated these data into methods such as multi-task learning (Ma et al. 2018). They, however, have failed to capture the sequential nature of the data, unlike our proposed method. In our method, the initial

filtering process of users is captured by a matrix factorization method trained based on rating data (i.e., ratings between users). Note that compared to matching based on complex, bilateral decision-making processes of two individuals, rating is a simpler, unidirectional behavior and, thus, we have used a relatively simple approach in recommender systems, matrix factorization.

After that, the results (i.e., user embeddings) of the rating prediction model are used to initialize the node embedding weights of our GNN model for matching preference prediction. The assumption that underpins our design choice of not learning the two tasks (i.e., rating and matching prediction) concurrently is that, as claimed in prior literature, the decision criteria used in the two tasks can differ (Bruch et al., 2016). Therefore, the objective function optimizing the prediction accuracy of the two tasks simultaneously may result in a suboptimal convergence in the learning process of the model. To empirically validate our assumption, we compare the performance of our sequential approach to that of the model trained based on a multitask approach. Table 4.3 summarizes the key components of our proposed framework.

Table 4.3. Summary of the proposed MatchSAGE framework

<table>
<tr><th colspan="2">Theoretical Foundation</th><th>Unique Characteristics in Online Matching Platforms</th><th>Proposed Framework (MatchSAGE)</th></tr>
<tr><td rowspan="3">SRM Components (Back and Kenny 2010)</td><td>Perceiver</td><td rowspan="2">A single user plays dual roles within the same service</td><td rowspan="2">(1) Dual embeddings for each user</td></tr>
<tr><td>Target</td></tr>
<tr><td>Relationship</td><td>Perceiver and target users dynamically interact, determining interpersonal perception</td><td>(2) DNN architecture to capture the complex interaction patterns between dual embeddings</td></tr>
<tr><td colspan="2">Multistage decision-making process (Bruch et al. 2016)</td><td>Users engage in multiple decision stages with potentially different rules at each</td><td>(3) Initialization of node embeddings weights (for matching prediction) using the outcomes of a rating prediction model</td></tr>
</table>

4.3.2. Matrix Factorization-Based Rating Prediction Model

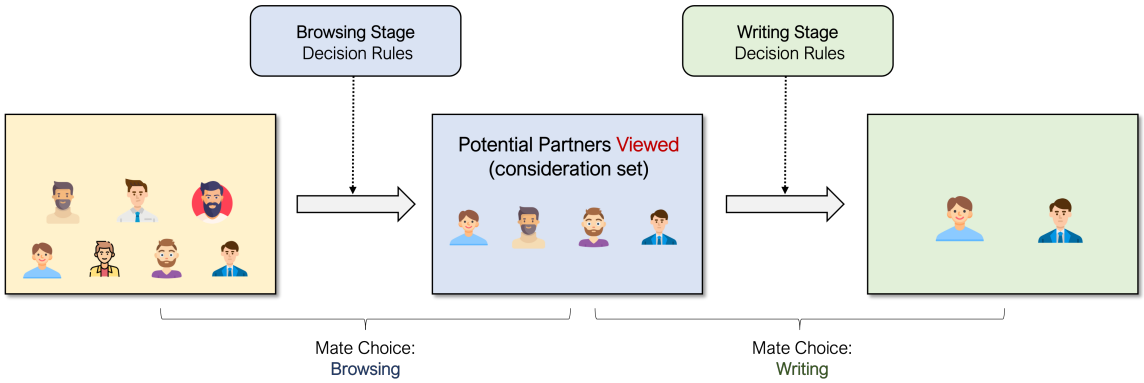


Figure 4.3. Multi-stage decision-making process of users in online dating services (Adapted from Bruch et al. 2016)

As described earlier, the search space is generally much larger in online matching platforms compared to offline, often resulting in a complex, multi-step choice process for a perceiver (such as in Figure 4.3). A user generally shortlists potential partners after a quick browse and takes a closer look at each of them before making a final decision. In each stage, different decision rules are applied. For instance, in the browsing stage, the user can select a subset of users based on general preference using simple criteria such as height, income, and geolocation. In the final decision stage, the user may take a more comprehensive set of criteria (e.g., perceived personalities) into consideration, conducting a more comprehensive and deliberate examination (Bruch et al. 2016).

Therefore, we need to clearly distinguish the two different decision-making stages rather than combine them in a single model. To this end, we propose a sequential learning process based on MF and GNN models. We first develop a rating prediction model to consider the users' browsing (i.e., initial screening) process. In general, MF algorithms such as singular value decomposition (SVD) showed a solid performance in rating prediction tasks despite their simplicity and ease of training (Koren et al. 2009, Paterek 2007). We used the general matrix

factorization (GMF) algorithm, which is a combination of matrix factorization and neural networks widely used for rating prediction in various applications (He et al. 2017).

GMF attempts to predict the rating given to item *i* by user *u* ($\hat{r}_{\mathbf{ui}}$) by learning weights of the output layer (**h**) and user/item latent factors:

$$\hat{r}_{\mathbf{ui}} = \mathbf{h}^{\boldsymbol{T}}(\mathbf{p_u} \odot \mathbf{q_i}), \quad (1)$$

where $\mathbf{p_u}$ and $\mathbf{q_i}$ are the latent factor vectors for user *u* and item *i*, respectively, which represent the characteristics of users/items in a latent vector space. The operator ⊙ denotes the element-wise (Hadamard) product of two vectors. As a regression task with continuous outcomes, loss functions such as mean squared error and mean absolute error can be used to optimize. After the model is trained, we extract the latent factor vectors ($\mathbf{p_u}$, $\mathbf{q_i}$), which capture the underlying characteristics of users/items (i.e., perceivers/targets) that led to the observed ratings. We use them as inputs to the matching prediction model as described in the following section.

4.3.3. Graph Neural Network Architecture for Matching Preference Learning

For matching preference learning, we propose a novel graph representation learning approach that extends cutting-edge GNN architectures. GNNs generate embeddings for each node by aggregating information from its nearby nodes, i.e., embeddings of the neighboring nodes. In online matching platforms, a graph-based approach can be more effective than extant approaches using user profile data (e.g., content-based methods) because it is less prone to the bias of self-reported data that might be excessively embellished. In addition, it can be more effective in analyzing complex interaction patterns between users compared to other collaborative filtering-based machine learning methods (Hamilton et al. 2017c). As described earlier, users often demonstrate idealized self-representation in online dating services (Ellison et al. 2006) while perceivers may have developed some strategies to reveal targets' true selves. However, solely

based on the target's self-reported profile, it is difficult to uncover how a perceiver makes judgments on a target. On the other hand, a target's previous interaction patterns with other users (e.g., ratings given to the target by other users) can be more useful in identifying the target effect on a perceiver's judgments.

Prior to training GNNs, the user embeddings ($h_{v_i}^0$) are initialized. Extant GNN-based approaches initialize the embeddings based on node characteristics such as user profile information (e.g., demographics, text descriptions) (Hamilton et al. 2017c), unsupervised embeddings (Duong and Hoang 2019), or random tensors (Abboud et al. 2021). In contrast, we adopt a sequential learning approach, as stated earlier, where the user embeddings are informed by the outcomes of the rating prediction model:

$$h_{v_i}^0 = x_{v_i}\ (i\ =\ 1, 2, 3, \ldots, |\mathcal{V}|),$$
$$h_u^{perceiver} = \mathbf{p_u}, \qquad h_u^{target} = \mathbf{q_u}, \tag{2}$$

where x_{v_i} is the node attribute for the node v_i; $|\mathcal{V}|$ is the number of nodes in the graph; $h_u^{perceiver}$ and h_u^{target} are initial node embeddings of user u, which are updated at each iteration of training; and $\mathbf{p_u}$ and $\mathbf{q_u}$ are user and item latent factors derived from the rating prediction model for the user, respectively.

By doing so, we can leverage users' stated preferences learned from previous ratings of other suggested users and seamlessly relay them to the later stage of the decision-making process (i.e., accept or reject). In short, the sequential learning pipeline incorporating embeddings from the rating prediction model in the matching preference model can effectively simulate the multi-stage decision-making process of users in online matching platforms. In the analysis section, we compare the prediction performances of matching prediction models using a variety of node initialization methods including ours.

Furthermore, unlike extant GNN methods, our model trains two embeddings for each user to capture how one plays perceiver ($h_u^{perceiver}$) and target (h_u^{target}) roles differently. Then, during every iteration of training, embeddings of the node's neighbors ($\mathcal{N}(v_i)$) are aggregated:

$$h_{\mathcal{N}(v_i)}^k \leftarrow AGGREGATE_k(\{h_u^{k-1}, \forall u \in \mathcal{N}(v_i)\}), \quad (3)$$

where $AGGREGATE_k(\cdot)$ is the aggregator function at iteration k and h_u^{k-1} is the embedding for node u from the last iteration $(k-1)$. Any order-invariant function, such as element-wise maximum or sum, can be used as the aggregator function. This new embedding is combined with the node's previous embedding and fed into a feed-forward neural network:

$$h_{v_i}^k \leftarrow \sigma(\mathrm{W}^{\mathrm{k}} \cdot COMBINE(h_{v_i}^{k-1}, h_{\mathcal{N}(v_i)}^k)), \quad (4)$$

where $\sigma(\cdot)$ is a non-linear activation function, such as rectified linear unit or sigmoid function, and $COMBINE(\cdot)$ is a function that combines the aggregated neighborhood information and previous node embedding. The resulting vectors after the final iteration K ($h_{v_i}^K$) are node representations (z_{v_i}) computed by the model:

$$z_{v_i} = h_{v_i}^K \quad (5)$$

Note that the above process was applied separately to update $h_{v_i}^{perceiver}$ and $h_{v_i}^{target}$, and ultimately to produce $z_{v_i}^{perceiver}$ and $z_{v_i}^{target}$.

Neighborhood aggregation methods differ in terms of how they define the aggregator ($AGGREGATE_k(\cdot)$) and combination ($COMBINE(\cdot)$) functions. For instance, graph convolutional networks (GCN) perform a simple average operation of neighboring nodes' previous embeddings (Kipf and Welling 2017), whereas graph attention networks (GAT) compute a weighted average of neighbor embeddings, using the attention mechanism to determine the weights (Veličković et al. 2018). For our method, we adopted and enhanced the GraphSAGE framework (Hamilton et al.

2017a) among various neighborhood aggregation approaches for a couple of reasons. First, it provides a variety of general aggregation functions, such as element-wise mean and max-pooling, which have demonstrated improved performance in many tasks. It is also a well-established method successfully used for many recommendation frameworks in practice (Wang et al. 2019, Ying et al. 2018, Zhang and Chen 2020).

Our GNN-based matching preference prediction model was trained in the recommendation context by optimizing a novel pairwise loss function, i.e., a max-margin loss that was customized to take into account two different embeddings of each user:

$$\sum_{(u,i,j)\in D_m} \max\left(-z_u^{perceiver} \cdot z_i^{target} + z_u^{perceiver} \cdot z_j^{target} + \Delta, 0\right) \tag{6}$$

$$D_m = \{(u,i,j) | (u,i) \in R^p, (u,j) \in R^n\}$$

where R^p includes positive interactions, i.e., user u accepted user i, and R^n includes negative interactions, i.e., user u rejected user j, and Δ is the hyperparameter that is set to a small positive value (Ying et al. 2018). The primary goal is to learn the user and item embeddings such that the similarity between positive interactions ($z_u^{perceiver} \cdot z_i^{target}$) are larger than that of negative pairs ($z_u^{perceiver} \cdot z_j^{target}$). Further details of the training procedure are described in Appendix 6.7.

4.3.4. Link Prediction and Individual Dating Suggestions

After the GNN model is trained, we calculate a user's probability of accepting a suggestion (i.e., a potential match) using the learned embeddings (z_i's). This can also be regarded as a directed link prediction task between two user nodes. The simplest approach to estimate the user i's potential preference towards user j would be computing a similarity measure such as the cosine similarity between the perceiver embedding of i ($z_i^{perceiver}$) and the target embedding of j (z_j^{target}):

$$cos(i,j) = \frac{z_i^{perceiver} \cdot z_j^{target}}{\| z_i^{perceiver} \| \| z_j^{target} \|} \quad (7)$$

This approach has apparent advantages, including simplicity and straightforward computation. However, it might be limited in modeling dynamic, non-linear interaction patterns between many users. Thus, we propose a DNN architecture with customized fine-tuning layers, which can encode and incorporate such dynamic patterns in predicting matching probabilities between users (Figure 4.4). In designing the fine-tuning layers, we have tested several well-established operators for link prediction using node embeddings (Grover and Leskovec 2016).

Table 4.4 describes the tensor operations used in the fine-tuning layers that combine embeddings of two users. These tensor operations are applied to each pair of inputs, which are $z_i^{perceiver}$ (perceiver embedding of user *i*) and z_j^{target} (target embedding of user *j*) and $z_j^{perceiver}$ (perceiver embedding of user *j*) and z_i^{target} (target embedding of user *i*) in this case. Note that the tensor operations yield a one-dimensional vector output by combining two embedding vectors:

$$\begin{aligned} z_1' &= f\left(z_i^{perceiver}, z_j^{target}\right) \\ z_2' &= f\left(z_j^{perceiver}, z_i^{target}\right), \end{aligned} \quad (8)$$

where f is one of the tensor operations of choice in Table 4, and z_1' and z_2' are outputs from the fine-tuning layers. Following that, the outputs (z_1' and z_2') are used as inputs to separate dense layers, whose outputs are concatenated at the next layer. Finally, the last dense layer takes the concatenated vector outputs and yields the final output ($\hat{y}_{ij}$), which is the probability of user *i* accepting user *j*. Hence, the network is trained using the binary cross-entropy loss as follows:

$$\sum_{(i,j)\in R^+ \cup R^-} y_{ij} log(\hat{y}_{ij}) + (1 - y_{ij}) log(1 - \hat{y}_{ij}), \quad (9)$$

where i and j are users in the training set, and $\hat{y}_{ij}$ is the predicted probability of user i accepting user j (i.e., j is suggested to i). y_{ij} is equal to 1 if user i accepted user j, and 0 otherwise.

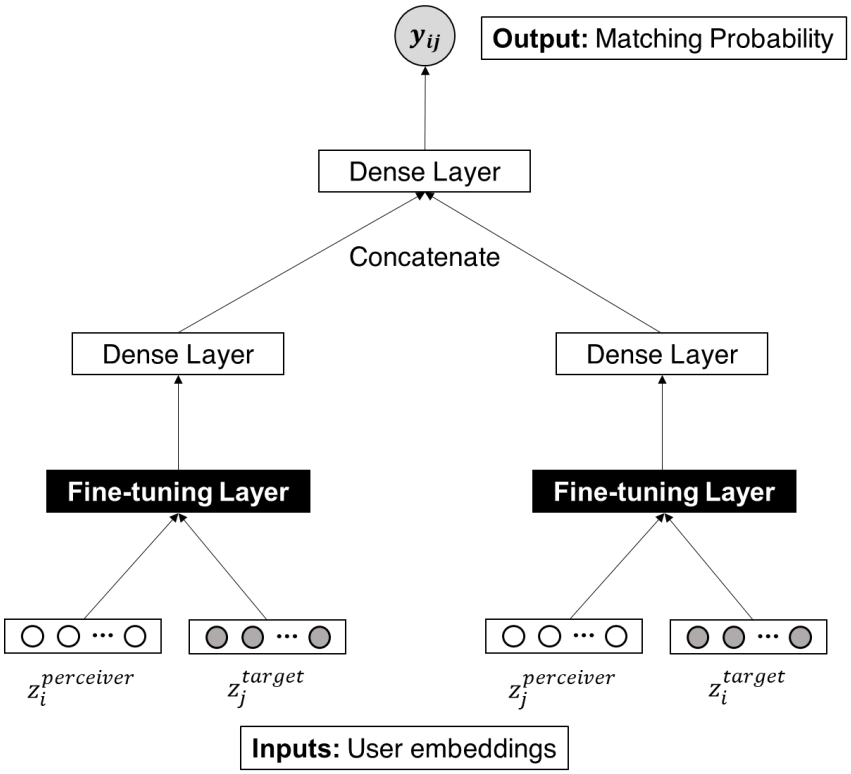


Figure 4.4. DNN architecture for matching probability prediction

Table 4.4. Fine-tuning tensor operations used in DNN architecture

Operation	Definition	Description
Average	$(v_i + v_j)/2$	Element-wise average of two tensors
Hadamard	$v_i * v_j$	Element-wise product of two tensors
L1	$\lvert v_i - v_j \rvert$	Absolute value of element-wise substraction of two tensors
L2	$(v_i - v_j)^2$	Square of element-wise substraction of two tensors
Concatenate	$[v_i; v_j]$	Concatenation of two tensors into one tensor

Using these fine-tuning tensor operations enables us to model non-linear, complex interaction patterns between various users in the platform. Furthermore, by incorporating the preferences of both users ($i \rightarrow j$ and $j \rightarrow i$), we can consider the reciprocal nature of the relationships as well. Combining $z_i^{perceiver}$ (perceiver embedding of user i) and z_j^{target} (target

embedding of user *j*) assesses the preference of user *i* towards user *j*. In contrast, $z_j^{perceiver}$ (perceiver embedding of user *j*) and z_i^{target} (target embedding of user *i*) assess user *j*'s preferences towards user *i*. Among the list in Table 4, we found that the Hadamard tensor operation showed the best performance.

After training, we used the DNN model to calculate the probability of successful matching between users on the platform. For each user, we shortlist the top-*k* users based on the calculated matching probabilities by the link prediction model, i.e., *k* users who are most likely to be accepted by the user according to the model. After excluding users who were already suggested, we were able to make personalized dating suggestions for individual users.

4.4. Experimental Results

4.4.1. Background

We collaborated with a South Korean mobile app service company that specializes in online dating. The app has been serving users since 2017 and is one of the leading dating apps in Korea as of 2023. Similar to other heterosexual dating apps, the app suggests users to other users of the opposite gender. For making dating suggestions, the company has hired professional human matchmakers, who are trained to make personalized matching suggestions, offer consulting services, and provide emotional support to the users. However, as the user base has grown considerably, the need for reducing the burden of matchmakers has increased rapidly. Currently, the matchmakers are overwhelmed by a huge number of users on both sides, taking excessive time and effort to provide customized suggestions to each user, not to mention the high training costs of newly hired matchmakers. Thus, the company has been actively seeking to reduce the burden by automating and streamlining the matchmaking process, i.e., using algorithms to shortlist a pool of users that are likely to be accepted by the user of interest. Hence, we implemented and trained

MatchSAGE using real-world data provided by the app to assess whether it can be used to improve the business by improving matching suggestions and curtailing the burden of the matchmakers.

4.4.2. Evaluation

The reciprocal recommendation problem in our study can be formulated as a binary classification task, in which each interaction (i.e., dating suggestion) is classified as *accepted* or *rejected*. Instead, it can be regarded as a personalized ranking task, in which users who are more likely to be preferred than others are identified. We adopt both perspectives and, hence, include multiple evaluation measures for binary classifiers and ranking algorithms.

First, we report the area under the receiver-operating curve (AUROC) and the area under the precision-recall curve (AUPRC) for the binary classification task. For the ranking task, we calculate the hit ratio ($HR@k$), a recall-based metric computed as:

$$HR@k = \frac{\text{Number of Hits among Top-k Users}}{\text{Number of Ground Truth Users}}, \tag{10}$$

where "hits" refer to relevant users in the recommended list (i.e., top-k users) and ground truth users refer to all users liked by a focal user. The hit ratio, however, does not consider the order (i.e., rankings) of recommendations. Therefore, we also consider normalized discounted cumulative gain ($NDCG@k$) defined as:

$$NDCG@k = Z_k \sum_{i=1}^{k} \frac{2^{r_i} - 1}{log_2(i+1)}, \tag{11}$$

where Z_K is a normalizing constant and r_i equals 1 if the user at rank i is relevant (i.e., accepted), and 0 otherwise. The $NDCG@k$ metric assigns higher weights to top-ranked results and incrementally smaller weights to lower ranks. All of the above metrics range from 0 to 1.

In addition to the offline experiment with historical data, we conducted a field test to investigate the impact of MatchSAGE on users' behavior and, consequently, on the sales and

revenue of the company by deploying in the app. The baseline suggestions were made by the company's existing recommendation method. To elaborate, the company developed a rule-based system to automate the process of providing daily match suggestions to users. Its rules were developed based on in-depth interviews with the human matchmakers hired by the company. While the rule-based system can be effective in capturing user preferences, its development is time-consuming, requiring manual efforts combined with expert domain knowledge. In addition, it is not versatile in changing environments such that if matching patterns of users change, then its performance might degrade.

We randomly selected active users in the app and offered each user five baseline suggestions made by the company's rule-based system (henceforth, RS) and another set of five suggestions from the top-5 suggestions made by MatchSAGE (henceforth, MS). In total, each user receives 10 recommendations (5 RSs and 5 MSs). Then, we compared the average acceptance rate and the average rating of suggestions made by the rule-based system (RS) and MatchSAGE (MS). We admit that we were not able to include other methods in the field test due to resource constraints, and we leave it as one of the future directions.[6]

We illustrate the details of the user interface and the online field test procedure (Figure 4.5). A user is provided with 10 matching suggestions, i.e., 10 user profiles in our settings. The app shows a photo and brief demographic information of each profile in the suggestions list (Figure 4.5 - (a)). To see more information about each suggestion, a user can click one of the photos in the suggestion list, which links to a new page with detailed information about the suggestion (Figure 4.5 - (b)), including additional profile pictures, self-description, etc. Among the 10 suggestions,

[6] From a practical perspective, it is sufficient to demonstrate a superior performance compared to an existing system for a company to adopt the proposed method. Therefore, we chose to compare to the current rule-based system, which did not require extensive investment from the company.

we tracked their acceptance rate to compare the overall performance of RS and MS. Note that if there was no response from a user regarding certain suggestions, then we considered them rejected. For instance, assume a user accepted #4 and #7, and rejected #1, #2, and #3 out of the 10 suggestions. The rest of the suggestions (i.e., #4, #5, #8, #9, and #10) will be also considered rejected by the user. In addition, we recorded the ratings given by users to these suggestions as a measure of user satisfaction on a scale of 1 to 5.

Figure 4.5. Screenshots from matching suggestion pages in the app

4.4.3. Data

For the offline experiment, we used user interaction data from the app, which includes acceptance, rejection, and rating records, between December 1, 2021, and February 15, 2022. We trained the models using data before February 1, 2022, and evaluated them with data after that. We included users having at least one interaction in both training and testing periods, resulting in

23,291 unique users, 4,987,498 matching interactions, and 2,352,962 rating interactions during the training period (Table 4.5).

Table 4.5. Summary statistics of user interaction data

Statistic	Training Data	Test Data	Rating Data
Number of edges (accepted)	834,278	26,304	-
Number of edges (rejected)	4,153,220	138,826	-
Number of edges (total)	4,987,498	165,130	2,352,962

On average, accepted suggestions show a significantly higher average rating compared to rejected ratings ($\mu_{accept} = 4.405, \mu_{reject} = 2.376, t = 990.552, p < .0001$). Nonetheless, actual choices (i.e., accept or reject) and preferences (i.e., rating) of the users do not always align with each other. For example, more than 40% of rejected suggestions received a neutral or positive rating (i.e., a rating of 3, 4, or 5), implying that users employ different criteria for the two preference types. We hypothesize that many of rejected suggestions had "deal-breakers", but other attributes of the suggested user compensated for those criteria, resulting in rejection despite a positive rating. This discrepancy between choices and ratings suggests that users may go through a multi-stage, complex decision-making process when choosing partners in online dating (Bruch et al. 2016).

For the field testing, we randomly recruited 282 active users in the app and provided matching suggestions according to the protocol described in the previous section. Then, we tracked whether the user accepted and rated the suggested user. As a result, we obtained 2,820 matching suggestion records and 1,522 rating records. Table 4.6 summarizes the data from the online field test.

Table 4.6. Summary of data from online field test

	RS	MS	Total
Number of suggestion records	1,410	1,410	2,820
Number of rating records	743	779	1,522

4.4.4. Offline Experimental Results

We compare the predictive performance of MatchSAGE to a wide range of well-established methods in recommender systems and graph representation learning. We briefly describe these baselines below:

- **Content-based Filtering**: This model resembles classical recommender methods that utilize user profile information to calculate compatibility scores between users (Pizzato et al. 2010a). However, two major improvements are made to the existing methods – (i) using a deep learning model (two-layer feedforward network) instead of iterative calculation of compatibility scores and (ii) including unstructured data (i.e., self-description) as input features. The text data is converted into user-level features using word representation learning (Mikolov et al. 2013) and sentence embedding methods (Arora et al. 2017).
- **Matrix Factorization**: Existing studies have employed matrix factorization-based latent factor models for reciprocal recommendation (e.g., Kutty et al., 2012; Neve & Palomares, 2019). Two popular variants – i.e., SVD (Koren et al. 2009) and neural matrix factorization (NeuMF) (He et al. 2017)- that represent users and items in a latent vector space were implemented in this study. The latter is a neural network extension of SVD that has been reported to outperform most other matrix factorization-based methods (He et al. 2017). These models were trained using stochastic gradient descent and binary cross-entropy loss.
- **Shallow node embedding methods:** In shallow node embedding methods, the encoder function is an "embedding lookup" (Hamilton et al., 2017a). As a result, most of them do not support an optimization function customized for the target task and inductive capability for unseen nodes. We include two widely used random walk-based node embedding methods for

generic purposes – DeepWalk (Perozzi et al. 2014) and Node2Vec (Grover and Leskovec 2016).

- **Existing GNN methods**: GNN architectures have been primarily used for designing general user-item recommender systems. For instance, Neural Graph Collaborative Filtering (NGCF) is a GNN-based collaborative filtering framework exploiting high-order connectivity in graphs (Wang et al. 2019). PinSAGE is a web-scale recommender system building upon the GraphSAGE architecture (Hamilton et al. 2017a), which has shown superior performance to many deep learning approaches (Ying et al. 2018). There are also some variations of the GNN architecture using different aggregation and combinator functions, including graph convolutional networks (GCN) (He et al. 2020) and graph attention networks (GAT) (Veličković et al. 2018). We added the two approaches, along with PinSAGE, to our baseline list.
- **MatchSAGE**: our matching prediction model trained using the procedure outlined in the previous section. After the GNN-based matching preference model, for which we used rating embeddings for initialization, we trained a DNN model for matching probability prediction. For the DNN model, we used the Hadamard tensor operation, which showed the best performance overall. We also report the performance of the models using other tensor operations in Appendix 6.8.

Offline evaluation results are summarized in Figures 4.6 and 4.7. The details of hyperparameter tuning are described in Appendix 6.8. First, we found that matrix factorization methods (i.e., SVD and NeuMF) generally outperformed the content-based method while state-of-the-art graph-based methods such as GAT showed improved performance over existing benchmarks. Notably, MatchSAGE significantly outperformed all other methods, achieving the

best classification metrics (i.e., ROAUC and PRAUC) and ranking metrics (i.e., HR@5 and NDCG@5).[7]

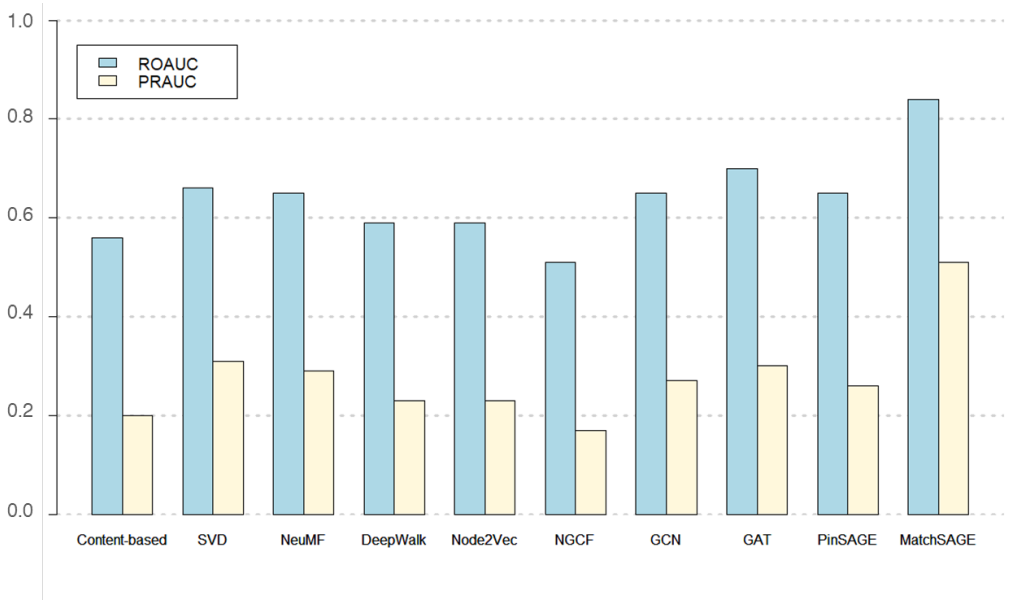


Figure 4.6. Offline evaluation results (classification)

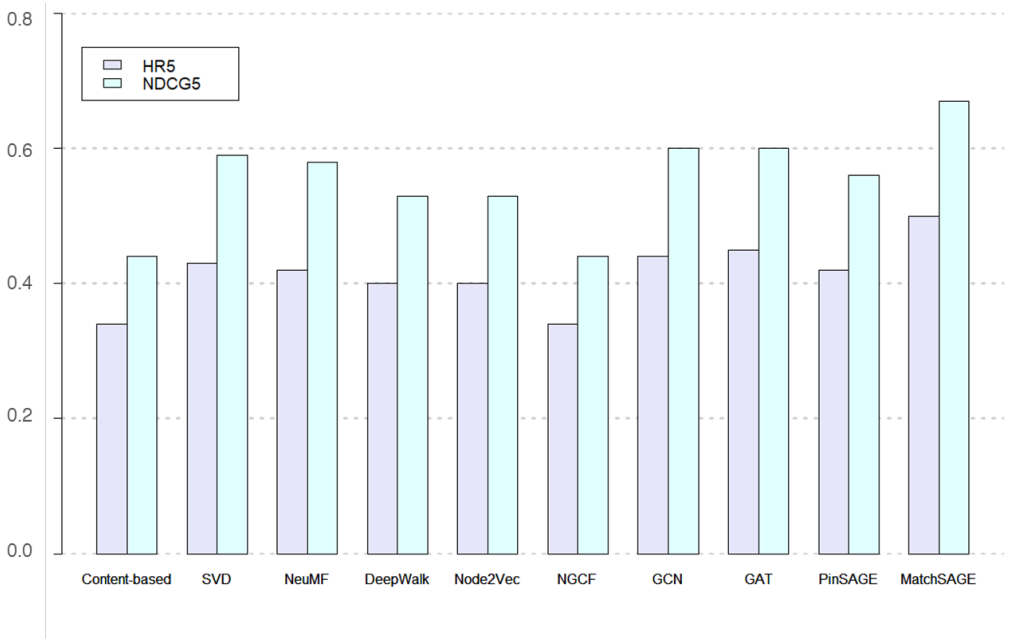


Figure 4.7. Offline evaluation results (ranking)

[7] We set the number of ranked users (k) when calculating HR@k and NDCG@k to five since the app currently suggests 3-5 users to a target user at a time. However, we show how the two metrics change with different values of k in Appendix 6.9.

The offline evaluation result demonstrates the effectiveness of MatchSAGE's methodological improvements, which effectively capture the users' behavioral patterns in online matching platforms. MatchSAGE has several characteristics that render it particularly suitable to manage reciprocal recommendation tasks in online matching platforms. To summarize, first, MatchSAGE takes advantage of large social network data to learn users' behavioral patterns (Xue et al. 2005). Also, it is grounded on GNNs to extend the use of network data to an *n-degree* of separation, overcoming the limitation of classic collaborative filtering approaches (e.g., SVD and NeuMF) which only use the ego network of a user to infer his/her behavior patterns (Wang et al. 2020). In addition, our method, along with other GNN methods, supports the inductive capacity to provide predictions for previously unseen nodes. Moreover, MatchSAGE takes account of users' unique behavior in online matching platforms. That is, the use of a network combined with a loss function tailored to capture users' acceptance/rejection of recommendations prevents MatchSAGE from overfitting to users' impression management behavior (e.g., embellished profile data). Note that while shallow embedding methods also use network data, they do not consider the polarity of a network that has positive and negative edges (i.e., accept and reject in our setting). In addition, MatchSAGE captures the multi-stage decision-making process of users in online matching platforms by sequentially combining rating and match prediction models. Lastly, MatchSAGE takes account of the dual role of users in online matching platforms by learning and producing two separate latent embedding vectors for a user.

4.4.5. Ablation Study

To determine which key component in MatchSAGE played a critical role in improving the performance, we conducted an ablation study as summarized in Table 4.7. Three baselines are considered in this ablation study. The baseline is identical to the vanilla PinSAGE method (Hamilton et al. 2017a, Ying et al. 2018). The other three lack one of the components of our

proposed method – i.e., (i) the sequential learning pipeline that reflects users' multi-stage decision-making (A), (ii) expanded embeddings that take account of the dual role of users (B), and (iii) the DNN architecture with fine-tuning layers to model complex relationships between perceiver and targets (C).

Table 4.7. Summary of ablation study settings

Model	Sequential learning pipeline	Expanded user embedding	DNN model for link prediction
Baseline (PinSAGE)			
A		V	V
B	V		V
C	V	V	
MatchSAGE	V	V	V

Instead of sequential learning, model A is trained using a multi-task loss function that minimizes the max-margin loss and mean-squared loss between predicted and actual ratings as follows:

$$\sum_{(u,i,j)\in D_m} \max\left(-z_u^{perceiver} \cdot z_i^{target} + z_u^{perceiver} \cdot z_j^{target} + \Delta, 0\right) + \sum_{(p,q)\in D_r} (r_{pq} - \hat{r}_{pq})^2 \quad (12)$$

$$D_m = \{(u,i,j)|(u,i) \in R^p, (u,j) \in R^n\}, D_r = \{(p,q)|(p,q) \in R^r\},$$

where R^r is the pair of users (p, q) in which p has rated q during the training period. Whereas, model B is trained using a similarly designed max-margin loss but without the two types of embeddings:

$$\sum_{(u,i,j)\in D_m} \max\left(-z_u \cdot z_i + z_u \cdot z_j + \Delta, 0\right) \quad (13)$$

$$D_m = \{(u,i,j)|(u,i) \in R^p, (u,j) \in R^n\},$$

where z_i's are embeddings of individual users. Finally, model 3 replaces the DNN link prediction model with the cosine similarity function (described in Equation 7).

The ablation study results show that all three components have contributed significantly to the performance improvement of MatchSAGE (Table 4.8). That is, models B, C, and D showed considerably improved results over the baseline (i.e., PinSAGE). However, dual embeddings played a more salient role in improving performance as model B yielded a more dramatic improvement than others. This result indicates that considering the different roles of users (i.e., perceivers and targets) is crucial in designing RRS.

Table 4.8. Ablation study results

Model	ROAUC	PRAUC	HR@5	NDCG@5
PinSAGE	.6551	.2620	.4206	.5559
Model B	.8035	.4634	.4717	.6280
Model C	.7747	.4031	.4767	.6395
Model D	.7929	.4942	.5036	.6781
MatchSAGE	**.8361**	**.5211**	**.5128**	**.6901**

4.4.6. Training Efficiency Analysis

Next, we analyzed the time needed to train a GNN model with different node initialization schemes including ours. As discussed earlier, there are several approaches to node initialization including node-level characteristics (e.g., demographics, text descriptions) (Hamilton et al. 2017c), learning-based methods (e.g., Node2Vec and DeepWalk) (Duong and Hoang 2019), and random tensors (Abboud et al. 2021). Thus, we compared various node initialization methods described in Table 4.9.

Table 4.9. Summary of node initialization methods used

Category	Method	Description
Random initialization	Constant	Initialize embeddings with a fixed value (e.g., 1)
	Uniform	Initialize embeddings by randomly generated numbers from a uniform distribution (Glorot and Bengio 2010)
	Normal	Initialize embeddings by randomly generated numbers from a normal distribution (He et al. 2015)
Node-level features	Content	Initialize embeddings with user profile information

Learning-based	Node2Vec	Initialize embeddings by pre-trained node vectors from the Node2Vec model
	MatchSAGE (ours)	Initialize embeddings by pre-trained node vectors from the rating prediction model

We trained each GNN model up to 500 epochs and tracked the calculated loss at each epoch (Figure 4.8). Overall, we find that content-based (Content) and learning-based approaches (Node2Vec and MatchSAGE) are generally more efficient – they take less time to converge compared to randomly initialized models. Especially, our method reaches the plateau in about 200 epochs and the converged loss value is lower than other models. These results indicate that MatchSAGE shows superior performance in terms of not only predictive accuracy but also the efficiency of training, which can be critical in deploying an RRS in a dynamic environment such as online matching platforms (Kim et al. 2012).

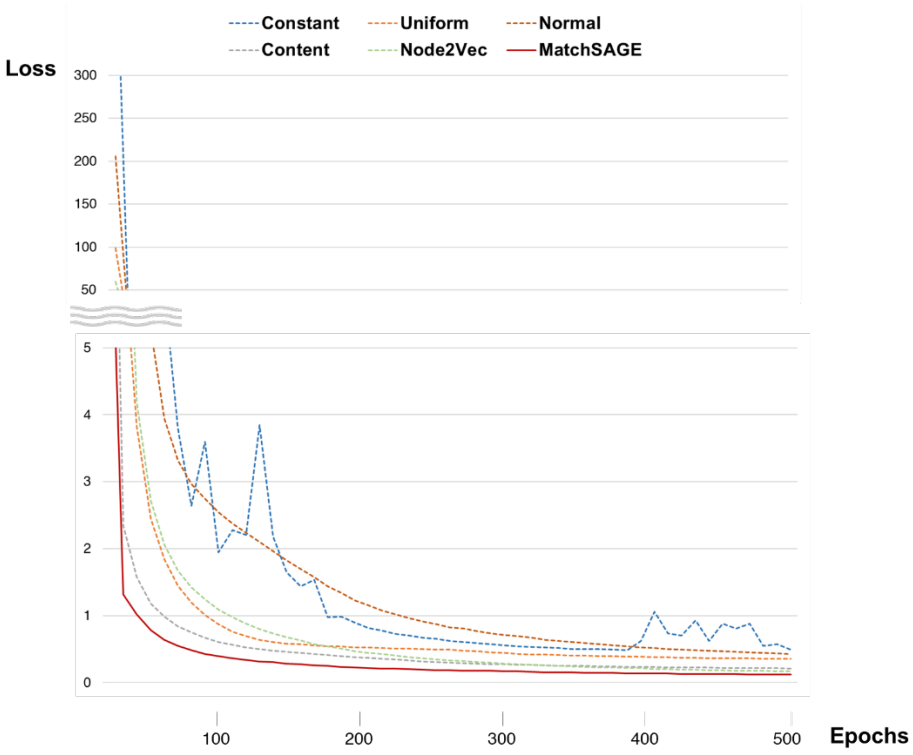


Figure 4.8. Training time analysis results

4.4.7. Online Field Test Results

We conducted an online field test, following the protocol described earlier. Table 4.10 summarizes the online field test results showing the average acceptance rates of the different suggestion methods, i.e., the existing rule-based system (RS) developed using the human experts' knowledge base and our MatchSage method (MS). The findings show that personalized recommendations using MS considerably improved the quality of suggestions, outperforming the RS method reflecting how human experts make recommendations. Specifically, MS shows a significantly higher average acceptance rate (**RS**: M = .3000, SD = .4584, **MS**: M = .3631, SD = .4810, p<.001) and average rating (**RS**: M = 3.0026, SD = .1.6035, **MS**: M = 3.5211, SD = 1.4973, p<.001) compared to RS. This corresponds to a 21.04% and 17.27% improvement in terms of average acceptance rate and average rating, respectively. To summarize, we find that our proposed framework can be used to provide automated and personalized suggestions that can significantly enhance the acceptance rate among users.

Table 4.10. Field test results

	Average Acceptance Rate	**Average Rating**
RS	.3000 (.4584)	3.0026 (1.6035)
MS	.3631 (.4810)	3.5211 (1.4973)
p-value	<.001	<.001
# Observations	2,820	1,522
Performance Improvement (%)	+21.04%	+17.27%

***Note**: standard errors are in paratheses

4.5. Discussions

4.5.1. Managerial Implications

From the managerial perspective, we proposed a data-driven framework that can potentially boost the revenue of online matching platforms (e.g., professional networking/career development, matchmaking and dating, knowledge sharing, and the gig economy) while reducing the cost of

human labor. Online field testing in the online dating app revealed that the personalized suggestions made by the proposed MatchSAGE method have the potential to enhance the average acceptance rate by up to 21% (see Table 4.10 for details). Considering the average number of matches per user in the app is currently 1.15 per week, it can be increased to 1.40 matches per week, indicating that around $7.5 additional revenue per user can be generated without additional costs.[8]

In addition, a higher level of customer satisfaction can be derived by implementing an RRS system based on our framework. That is, our results have shown that MatchSAGE ($\mu = 3.5211, \sigma = 1.4973$) can make suggestions that lead to higher rating scores compared to the existing system ($\mu = 3.0026, \sigma = 1.6035$) (Table 4.10). The higher average rating implies that the users are in general more satisfied with the suggestions, which can lead to higher customer loyalty and retention rates. Moreover, the automation of matching suggestions will reduce the burden of human matchmakers and result in more efficient utilization of company resources. Human matchmakers can put more focus on managing post-matching customer satisfaction than on making suggestions. For instance, they can provide emotional support and further dating advice to users, enhancing their overall user experience and satisfaction in the longer term.

4.5.2. Research Contributions

Our study also contributes to the information systems knowledge base. First, the proposed MatchSAGE framework in this study can be used for various applications of recommender systems in online matching platforms. Although both recommender systems (e.g., Bouayad et al., 2020; Ghose et al., 2019; Lee et al., 2020) and online matching platforms (e.g., Bapna et al., 2016; Jung et al., 2019, 2022) have been extensively studied in previous IS literature, there has been little

[8] The expected revenue from a successful matching in the app is around $30.

attempt to study their intersection from a unique perspective. Our study contributes to the knowledge base by providing a new angle in investigating the two salient topics in IS with a data-driven approach.

Furthermore, the proposed method has theoretical implications in that it adapts and extends SRM, a social science theory extensively used in the psychology and organization science literature, to guide the design of an information artifact important in practice, i.e., RRSs. Specifically, we interpreted the components of SRM in an online matching platform context, where people present distinct patterns of behavior compared to those in offline settings. Thus, our study contributes to IS research by not only proposing a decision-supporting artifact with practical relevance (Hevner et al. 2004, Padmanabhan et al. 2022) but also enhancing the synergy between big data and theory (Rai 2016). Our approach of attending to SRM for understanding the perception of interpersonal relationships in online matching platforms may lead to interesting implications for future studies.

Lastly, from the methodological perspective, our study highlights the value of graph representation learning, which has received relatively thin attention in the field of RRSs. Particularly, we extended existing graph representation learning methods based on SRM-guided components: sequential learning combining rating and matching prediction tasks and dual embeddings. Our results showed the unique design of MatchSAGE significantly outperformed state-of-the-art graph representation learning methods such as PinSAGE, GCN, and GAT.

4.5.3. Limitations and Future Work

Our research can be extended in multiple directions in future studies. First, we expect that the performance of the overall framework can be further improved by utilizing human inputs. By reducing the burden of menial matchmaking, we can encourage human experts to collaborate with

the machine, coming up with creative and novel solutions. Combining human and machine intelligence can help design a system that demonstrates superior intelligence (Jain et al. 2021). For instance, humans can both learn from the machine and teach the machine where they are failing or have uncertain predictions (Johns et al. 2015). Second, we may run another field test with multiple baselines including those used in the offline experiment. In this study, due to resource limitations, we could only deploy MatchSAGE in the app and compared it against the recommender system already running in the app. However, to further validate the value of adopting MatchSAGE in a real-world setting, it would be ideal to make a comparison to other baselines such as NeuMF and PinSAGE. Finally, our framework can be adopted and extended to other settings in online matching platforms. For instance, some mobile dating apps provide other functionalities to connect users (e.g., chats, self-promotion, etc.). Future work can flexibly modify the sequential learning pipeline and GNN architecture proposed in this study are flexible to accommodate such differences.

4.6. Conclusion

In this study, we proposed a novel graph representation learning framework for reciprocal recommendations. The proposed MatchSAGE framework extends graph representation learning and recommender systems, incorporating people's unique behavioral patterns exhibited in online matching platforms from the SRM perspective. MatchSAGE outperformed cutting-edge approaches in offline experiments using large-scale data from an online dating application. In addition, an online field test revealed that the application of MatchSAGE in a real-world setting could lead to not only increased revenue but also improved customer satisfaction. Our study contributes to the body of knowledge in information systems by designing and implementing a novel design artifact for real-world problem-solving from a theoretical perspective.

5. CONCLUSIONS

CDSR is evolving into a crucial paradigm in the IS field with recent climactic advances in data-related technologies. However, it is still unclear how CDSR may advance our understanding of IS and its impact on our lives. This dissertation offers three essays that show how CDSR can add value to the body of IS knowledge while maximizing its scope and applicability. The first essay suggests a novel framework to offer individual-level justifications for any black-box machine learning classifier utilized in healthcare. The second essay extends cutting-edge approaches in graph analytics and natural language processing to create a fairness-aware graph representation learning for multiple patient outcome prediction in critical care. The third paper offers a framework for theory-driven learning of graph representations for reciprocal recommendations in online matching platforms, which considerably differ from traditional user-item recommendation tasks and require special attention to user behaviors.

To summarize, this dissertation focuses on proposing methods to enhance the rigor and relevance of CDSR research. The information artifacts designed and implemented in the three essays tackle significant challenges in practical domains including healthcare and mobile applications. These studies make extensive contributions to business analytics, health IS, and design science research and provide implications for practitioners in the field.

6. APPENDIX

6.1. Hyperparameter Tuning Results for Machine Learning Prediction Models (Essay 1)

We performed five-fold cross-validation to find optimal hyperparameters for black-box classifiers for each dataset. The following is the description of each hyperparameter and the best hyperparameters found for each classifier (Table 6.1).

- **Maximum depth (max depth):** The maximum depth of the tree. This is applicable to tree-based algorithms, i.e., RF, XGB, and CB. Possible values are {10, 20, 30}.
- **Number of estimators (# estimators):** The maximum number of estimators that are constructed. This is applicable to ensemble-based algorithms, i.e., RF, ADA, XGB, and CB. Possible values are {100, 200, 300}.
- **Learning rate:** For boosting algorithms (ADA, XGB, and CB), this is the weight applied to the classifier at every iteration. Whereas for MLP, this refers to the size of the step in updating the model parameters during stochastic gradient descent (SGD). Possible values are {.01, .001, .0001}.
- **Number of epochs: (# epochs):** The maximum number of times each data instance is used to update the model parameters during SGD. This is applicable to MLP. Possible values are {100, 500, 1000}.

Table 6.1. Best hyperparameters found for black-box classifiers

Algorithm	Dataset	Max depth	# Estimators	Learning rate	# Epochs
RF	Breast Cancer	10	100	-	-
	Heart Disease	20	100	-	-
	Fragility Fracture	20	300	-	-
ADA	Breast Cancer	-	300	.01	-
	Heart Disease	-	300	.01	-
	Fragility Fracture	-	300	.01	-

MLP	Breat Cancer	-	-	.001	1000
	Heart Disease	-	-	.0001	1000
	Fragility Fracture	-	-	.0001	500
XGB	Breast Cancer	10	300	.01	-
	Heart Disease	10	100	.01	-
	Fragility Fracture	10	300	.01	-
CB	Breast Cancer	10	200	.01	-
	Heart Disease	10	100	.001	-
	Fragility Fracture	10	300	.01	-

6.2. Hyperparameter Tuning Results for Different Local Explanation Models (Essay 1)

We tried different values for hyperparameters of local explanation models and chose ones that show the best result in terms of *Local Fidelity* scores. The following is the description of each hyperparameter and the best hyperparameters found for each explainer (Table 6.2).

- **Kernel width** (w): A hyperparameter to control penalization weights for the synthetic data points in LIME (Ribeiro et al. 2016). The exponential kernel for each data point $x_j^{(i)}$ is defined as $\exp\left(-\frac{distance\left(x_i, x_j^{(i)}\right)}{w\sqrt{dimension(X)}}\right)$, where $dimension(X)$ refers to the dimensionality of the feature space, i.e., the number of features. Possible values are {.25, .75, 1}.
- **Initial radius** (η): The initial radius to sample in the L2-hypersphere around the instance being explained (x_i) in the growing spheres algorithm (Laugel et al. 2018a). Possible values are {.25, .5, .75}.
- **Sampling radius** (ϵ): The radius to uniformly sample synthetic data points in the vicinity of $x_{border}^{(i)}$ (in LS) or $x_{embedding}^{(i)}$ (in LEAP). As mentioned, unlike in ROLEX, this value is applied to all data instances when using LS or LEAP. Possible values are {.3, .5, .7}.
- **Number of neighbors** (N_{nei}): the number of nearest neighbors to consider when performing subspace learning (e.g., PCA) in LEAP. Possible values are {30, 40, 50}.

Table 6.2. Best hyperparameters found for benchmark local explanation models

Dataset		Breast Cancer				Heart Disease				Fragility Fracture			
Hyperpara-meter / Algorithm		w	η	ϵ	N_{nei}	w	η	ϵ	N_{nei}	w	η	ϵ	N_{nei}
RF	LIME	.25	-	-	-	1	-	-	-	1	-	-	-
	LS	-	.25	.3	-	-	.75	.3	-	-	.5	.7	-
	LEAP	-	-	.5	40	-	-	.3	30	-	-	.7	40
ADA	LIME	.25	-	-	-	1	-	-	-	1	-	-	-
	LS	-	.25	.3	-	-	.25	.3	-	-	.25	.5	-
	LEAP	-	-	.3	50	-	-	.3	40	-	-	.5	50
MLP	LIME	1	-	-	-	.1	-	-	-	1	-	-	-
	LS	-	.75	.5	-	-	.75	.5	-	-	.25	.3	-
	LEAP	-	-	.3	50	-	-	.3	50	-	-	.7	50
XGB	LIME	1	-	-	-	1	-	-	-	1	-	-	-
	LS	-	.25	.3	-	-	.5	.7	-	-	.25	.3	-
	LEAP	-	-	.3	40	-	-	.3	40	-	-	.3	50
CB	LIME	.75	-	-	-	.75	-	-	-	1	-	-	-
	LS	-	.5	.3	-	-	.25	.3	-	-	.25	.5	-
	LEAP	-	-	.3	50	-	-	.3	40	-	-	.5	50

6.3. Algorithms for SIF and SF Methods (Essay 2)

Algorithm 1 describes the embedding procedure of the SIF method and Algorithm 2 describes the SF method in detail.

Algorithm 6.1. Smooth inverse frequency (SIF) embedding method

Input: Word embeddings $\{z_\omega : \omega \in W\}$, a set of sentences S, parameter a, and estimated probabilities $\{p(\omega) : \omega \in W\}$ of the words.
Output: Sentence embeddings $\{z_s : s \in S\}$
for all sentence s in S **do**:
  $z_s \leftarrow \frac{1}{|s|}\sum_{\omega \in W} \frac{a + p(\omega)}{a} z_\omega$,
end for
Form a matrix X_S whose columns are $\{z_s : s \in S\}$, and let u be its first singular vector
for all sentence s in S **do**:
  $z_s \leftarrow z_s - uu^T z_s$
end for

Algorithm 6.2. Smooth frequency (SF) embedding method

Input: Embeddings of POA conditions $\{z_c : c \in C\}$, a set of patients $\mathcal{P}$, parameter a, and estimated probabilities $\{p(c) : c \in C\}$ of POA conditions
Output: Patient embeddings $\{z_p : p \in \mathcal{P}\}$
for all patient p in $\mathcal{P}$ **do**:

2: $z_p \leftarrow \frac{1}{|C|}\sum_{c \in C} \frac{a + p(c)}{a} z_c$,
3: **end for**
4: Form a matrix $X_{\mathcal{P}}$ whose columns are $\{z_p : p \in \mathcal{P}\}$, and let u be its first singular vector
5: **for all** patient p in $\mathcal{P}$ **do**:
6: $z_p \leftarrow z_p - uu^T z_p$
7: **end for**

6.4. Additional Details of the Data Used for Analysis (Essay 2)

Table 6.3 is the summary of the data used for analysis and Table 6.4 provides the summary statistics of the created bipartite graph having patients and diseases as nodes

Table 6.3. Summary of data used for analysis

	Training data	**Test data**	**Total**
Timeframe	24 hours after admission	After the training period until the patient is discharged	-
Number of patient visits	16,110	6,905	23,015
Number of unique POA conditions	719	649	752
Number of unique PA conditions	750	673	783

Table 6.4. Summary statistics of the patient-disease bipartite graph

	All nodes	**Patient nodes**	**POA/PA nodes**
Number of nodes	23,794	23,015	779
Average degree	14.0316	7.2533	214.2939

6.5. Details of Hyperparameter Settings for Evaluation (Essay 2)

Prior to prediction, graph embedding methods require training a model that encodes nodes into vector representations (encoders). Table A1 presents hyperparameter settings for the graph embedding methods with random walks and Table A2 presents hyperparameter settings for the neighborhood aggregation methods.

Table 6.5. Hyperparameter settings for graph embedding with random walks

Hyperparameter	**Setting**	**Description**
Number of walks	5	Number of random walks per node
Length of walks	30	Maximum number of traversed nodes in a random walk
Return parameter	1	Parameter controlling the likelihood of immediately revisiting a node in the walk. Setting it to a higher value will make the random walker less likely to visit an already visited node (only applies to Node2Vec).
In-out parameter	2	Parameter controlling how the search is performed between "inward" and "outward" nodes. If the parameter is large than 1, the walk will approximate the breadth-first search behavior, while if smaller than 1, it will approximate the depth-first search behavior (only applies to Node2Vec).
Model architecture	Skip-gram	Neural network architecture that learns vector representation from random walks
Dimensionality	50	Dimensionality of the latent space learned by the neural network model
Window size	5	Maximum number of nodes between the current and predicted node in sampled random walks

Table 6.6. Hyperparameter settings for graph embedding with neighborhood aggregation

Hyperparameter	**Setting**	**Description**
Hidden size	50	Size of hidden layers
Number of hidden layers	2	Number of hidden layers in the neural network
Number of heads	6	Number of heads in the attention mechanism (only applies to the graph attention network model)
Dimensionality	50	Dimensionality of the latent space learned by the neighborhood aggregation model
Loss function	Hinge loss	Loss function that minimizes the distance between positive pairs of nodes and maximizes the distance between negative pairs of nodes. The distance is computed as the dot product of the two node vectors.

After the embeddings are obtained, a supervised machine learning model (i.e., a

feedforward neural network in our case) that is used for link prediction is trained. Table A3 presents hyperparameter settings for the feedforward neural network.

Table 6.7. Hyperparameter settings for feedforward neural networks

Hyperparameter	**Setting**	**Description**
Number of hidden layers	1	Number of hidden layers in the network, i.e., the depth of the network
Hidden size	30	Dimensionality of the hidden layer, i.e., the width of the network
Activation function	Rectified linear unit (ReLU)	Nonlinear activation function applied to the hidden layer
Optimizer	Adam	The type of optimizer used for parameter updating by gradient descent
Learning rate	0.001	The step size for updating the model parameters
Number of epochs	10	Number of training epochs
Loss function	Cross entropy loss	Binary cross entropy loss widely used for binary classification tasks

6.6. List of Common and Uncommon Medical Conditions in Data (Essay 2)

Table B1 presents a list of the 30 most common and uncommon medical conditions in the data used in this study, along with their associated ICD-10 codes.

Table 6.8. List of 30 most common and uncommon medical conditions

30 Most Common Conditions		**30 Most Uncommon Conditions**	
Medical Condition	**ICD-10 code**	**Medical Condition**	**ICD-10 code**
Acute respiratory failure	J96.00	Volume depletion, unspecified	E86.9
Acute renal failure	N17.9	Hematuria, unspecified	R31.9
Essential (primary) hypertension	I10	Phlebitis and thrombophlebitis of unspecified site	I80.9
Pneumonia, unspecified organism	J18.9	ST elevation (STEMI) myocardial infarction involving other coronary artery of anterior wall	I21.09
Paroxysmal atrial fibrillation	I48.0	Atrioventricular block, complete	I44.2
Sepsis, unspecified organism	A41.9	Malignant neoplasm of prostate	C61
Altered mental status, unspecified	R41.82	Alkalosis	E87.3
Hyperglycemia, unspecified	R73.9	Hepatic failure, unspecified	K72.9
Heart failure, unspecified	I50.9	Ischemic cardiomyopathy	I25.5
Respiratory failure, unspecified with	J96.91	Unspecified adrenocortical	E27.40

hypoxia		insufficiency	
Hypotension, unspecified	I95.9	Parkinson's disease	G20
Chronic obstructive pulmonary disease (COPD)	J44.9	Epilepsy, unspecified, not intractable, with status epilepticus	G40.901
Acidosis	E87.2	Sleep apnea, unspecified	G47.30
Severe sepsis with septic shock	R65.21	Esophageal varices with bleeding	I85.01
Elevated white blood cell count, unspecified	D72.829	Other forms of acute ischemic heart disease	I24.8
Chronic kidney disease, unspecified	N18.9	Interstitial pulmonary disease, unspecified	J84.9
Unspecified protein-calorie malnutrition	E46	Decubitus ulcer, stage unspecified Head	L89.90
Hyperlipidemia, Unspecified	E78.5	Cardiac arrhythmia, unspecified	I49.09
Gastrointestinal hemorrhage, unspecified	K92.2	Generalized idiopathic epilepsy and epileptic syndromes	G40.3
Hypokalemia	E87.6	Pyothorax without fistula	J86.9
Hypothyroidism, Unspecified	E03.9	Perforation of intestine (nontraumatic)	K63.1
Systemic Inflammatory Response Syndrome of non-infectious origin without organ failure	R65.2	Acute and subacute infective endocarditis	I33.0
Hyperosmolality and hypernatremia	E87.0	Subarachnoid hemorrhage, unspecified	I60.9
Cystitis, unspecified	N30.9	Peritoneal abscess	K65.1
Hypovolemia	E86.1	Calculus of kidney	N20.0
Thrombocytopenia, unspecified	D69.6	Other specified types of non-Hodgkin lymphoma, unspecified site	C85.80
Cardiac arrest, cause unspecified	I46.9	Multiple fractures of ribs	S22.4
Unspecified convulsions	R56.9	Osteomyelitis, unspecified	M86.9
Hypomagnesemia	E83.42	Acute cholecystitis	K81.0
Pneumonitis due to inhalation of food and vomit	J69.0	Acute on chronic diastolic (congestive) heart failure	I50.33

6.7. Training Details for MatchSAGE (Essay 3)

MatchSAGE is trained with backward propagation using the hinge loss function. The forward propagation phase adapts that of GraphSAGE (Hamilton et al. 2017a), but accommodates learning two types of embeddings (active and passive). Algorithm 6.3 describes the pseudocode for the forward propagation, which yields two types of node embeddings (i.e., $z_v^{perceiver}$ and z_v^{target} for all $v \in \mathcal{V}$). The embeddings are initialized using user and item input features

($\mathbb{x}_v^{user}, \mathbb{x}_v^{item}$), which are outputs from the rating prediction model in our case. These embeddings are updated in the back-propagation phase by stochastic gradient descent.

Algorithm 6.3. MatchSAGE forward propagation algorithm

Input: Graph $\mathbb{G}(\mathcal{V}, \mathcal{E})$; user input feature $\{\mathbb{x}_v^{user}, \forall v \in \mathcal{V}\}$; item input feature $\{\mathbb{x}_v^{item}, \forall v \in \mathcal{V}\}$; depth K; weight matrices $\{\mathbb{W}^k, \forall k \in [1, K]\}$; non-linearity σ ; differentiable aggregator functions $\{AGGREGATE_k, \forall k \in [1, K]\}$; neighborhood function $\mathcal{N}: v \rightarrow 2^{\mathcal{V}}$

Output: Vector representations $z_v^{perceiver}, z_v^{target}$ for all $v \in \mathcal{V}$

$\mathbf{h}_v^0 \leftarrow \mathbb{x}_v^{item}, \mathbf{g}_v^0 \leftarrow \mathbb{x}_v^{user}, \forall v \in \mathcal{V}$;

for $k = 1 \cdots K$ **do**:

  for $v \in \mathcal{V}$ **do:**

    $\mathbf{g}_{\mathcal{N}(v)}^k \leftarrow AGGREGATE_k\left(\mathbf{g}_u^{k-1}, \forall u \in \mathcal{N}(v)\right)$;

    $\mathbf{h}_{\mathcal{N}(v)}^k \leftarrow AGGREGATE_k(\{\mathbf{h}_u^{k-1}, \forall u \in \mathcal{N}(v)\})$;

    $\mathbf{g}_v^k \leftarrow \sigma(\mathbb{W}^k \cdot COMBINE(\mathbf{g}_v^{k-1}, \mathbf{g}_{\mathcal{N}(v)}^k))$

    $\mathbf{h}_v^k \leftarrow \sigma(\mathbb{W}^k \cdot COMBINE(\mathbf{h}_v^{k-1}, \mathbf{h}_{\mathcal{N}(v)}^k))$

  end

  $\mathbf{g}_v^k \leftarrow \text{NORMALIZE}(\mathbf{g}_v^k), \forall v \in \mathcal{V}$

  $\mathbf{h}_v^k \leftarrow NORMALIZE(\mathbf{h}_v^k), \forall v \in \mathcal{V}$

end

$z_v^{perceiver} \leftarrow \mathbf{g}_v^k, z_v^{target} \leftarrow \mathbf{h}_v^k, \forall v \in \mathcal{V}$

6.8. Hyperparameter Settings for Different Matching Prediction Models (Essay 3)

We describe in detail hyperparameter settings for training different models. Table E1 describes hyperparameter settings for MF methods (i.e., SVD and NeuMF) and Table E2 shows those for GNN methods (i.e., PinSAGE and MatchSAGE).

Table 6.9. Hyperparameter settings for MF methods

Hyperparameter	**Value**
Learning rate	1e-3
Dimension of hidden layer	50

Dimension of node embedding	50

Table 6.10. Hyperparamter settings for GNN methods

Hyperparameter	**Value**
Dimension of node feature	50
Number of hidden layers	1
Dimension of hidden layer	100
Dimension of node embedding	50
Non-linearity	Rectified Linear Unit
Learning rate	1e-2
Optimizer	Adam

It should be noted that there are two steps in training – node representation learning and prediction using node embeddings – for GNN methods. After node embeddings are obtained, we trained a DNN model to make predictions on node pairs, i.e., link prediction. Table E3 summarizes hyperparameter settings for the DNN model.

Table 6.11. Hyperparameter settings for the DNN model for link prediction

Hyperparameter	**Value**
Dropout rate	.5
Tensor operation	Hadamard
Number of hidden layers	1
Dimension of hidden layer	100
Dimension of node embedding	100
Non-linearity	Rectified Linear Unit
Learning rate	1e-3
Optimizer	Adam
Loss function	Cross-entropy loss

Finally, we compare the performance of the DNN models using different tensor operations. As mentioned, the Hadamard operations showed the best performance overall. Evaluation results using other tensor operations are summarized in Table E4.

Table 6.12. Evaluation results using various tensor operations

Tensor Operation	**ROAUC**	**PRAUC**	**HR@5**	**NDCG@5**
Average	.4831	.1549	.3430	.4466
Hadamard	**.8361**	**.5211**	**.5128**	**.6901**

L1	.5833	.2099	.4089	.5428
L2	.4874	.1480	.3414	.4394
Concatenate	.5783	.2095	.3624	.4721

6.9. Hit Ratio and NDCG for Different Values of *k* (Essay 3)

We reported HR@5 and NDCG@5 (i.e., $k = 5$) in the analysis section as the app usually suggests five users to a target user at a time. However, we also show how the two metrics change with different values of k from 1 to 10 in Figure 6.1 and Figure 6.2. We find that MatchSAGE consistently outperforms all baselines in both metrics.

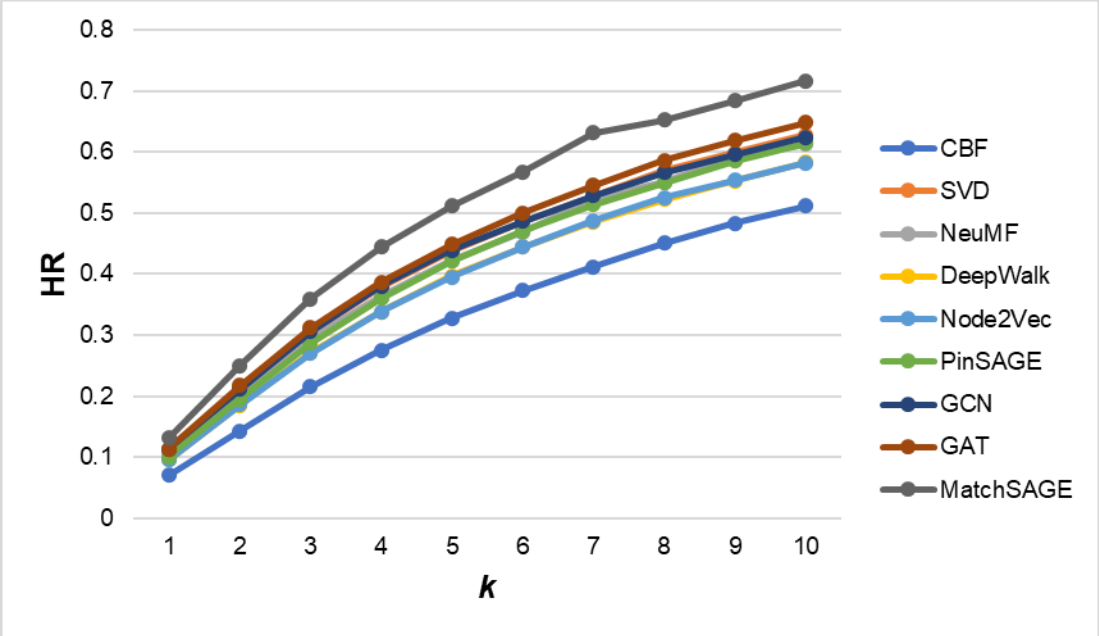


Figure 6.1. Hit Ratio with Different *k*

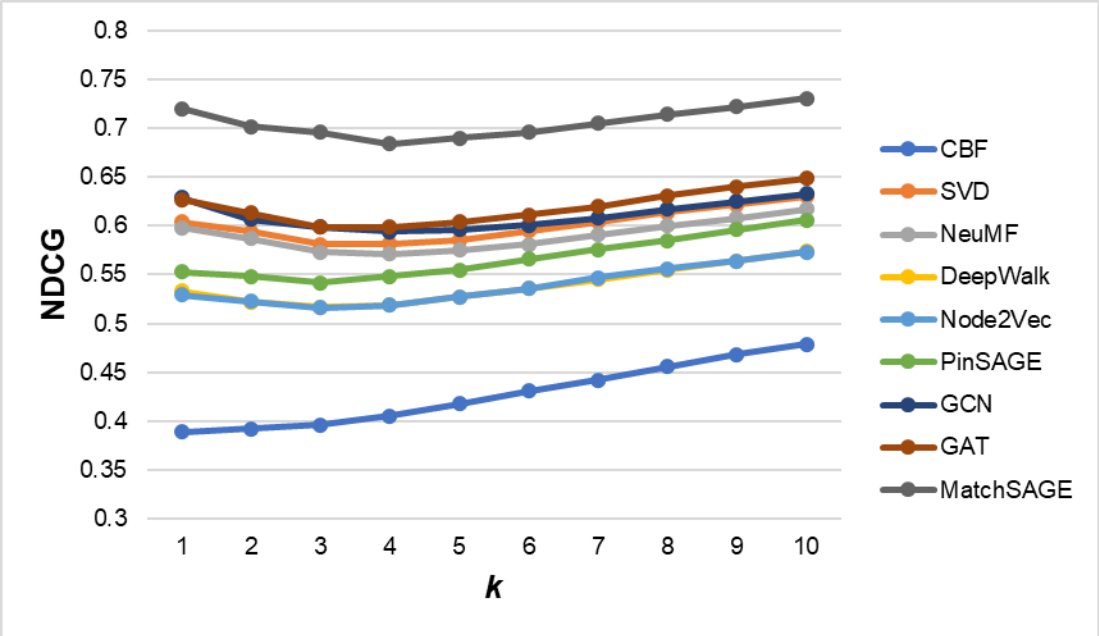


Figure 6.2. NDCG with Different *k*

7. REFERENCES

Abbasi A, Zhang Z, Zimbra D, Chen H, Nunamaker JF (2010) Detecting Fake Websites: The Contribution of Statistical Learning Theory. *MIS Quarterly* 34(3):435–461.

Abboud R, Ceylan İİ, Grohe M, Lukasiewicz T (2021) The Surprising Power of Graph Neural Networks with Random Node Initialization. (June 4) http://arxiv.org/abs/2010.01179.

Abdollahpouri H, Burke R, Mobasher B (2017) Controlling Popularity Bias in Learning-to-Rank Recommendation. *Proceedings of the Eleventh ACM Conference on Recommender Systems*. (ACM, Como Italy), 42–46.

Ahmad MA, Teredesai A, Eckert C (2018) Interpretable Machine Learning in Healthcare. *2018 IEEE International Conference on Healthcare Informatics (ICHI)*. (IEEE, New York, NY), 447–447.

Akehurst J (2011) CCR - A Content-Collaborative Reciprocal Recommender for Online Dating. *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence*.

Anderson C (2008) The End of Theory: The Data Deluge Makes the Scientific Method Obsolete. *Wired* https://www.wired.com/2008/06/pb-theory/.

Arora S, Liang Y, Ma T (2017) A Simple But Tough-to-beat Baseline for Sentence Embeddings. *Proceedings of the 5th International Conference on Learning Representations*. (Toulon, France).

Back MD, Kenny DA (2010) The Social Relations Model: How to Understand Dyadic Processes. *Social and Personality Psychology Compass* 4(10):855–870.

Bahdanau D, Cho K, Bengio Y (2015) Neural Machine Translation by Jointly Learning to Align and Translate. *International Conference on Learning Representations*.

Bapna R, Ramaprasad J, Shmueli G, Umyarov A (2016) One-Way Mirrors in Online Dating: A Randomized Field Experiment. *Management Science* 62(11):3100–3122.

Barabási AL, Bonabeau E (2003) Scale-free networks. *Scientific american* 288(5):60–69.

Bardhan I, Oh J ha (Cath), Zheng Z (Eric), Kirksey K (2015) Predictive Analytics for Readmission of Patients with Congestive Heart Failure. *Information Systems Research* 26(1):19–39.

Black DM, Thompson DE, Bauer DC, Ensrud K, Musliner T, Hochberg MC, Nevitt MC, Suryawanshi S, Cummings SR (2000) Fracture Risk Reduction with Alendronate in Women with Osteoporosis: The Fracture Intervention Trial. *The Journal of Clinical Endocrinology & Metabolism* 85(11):4118–4124.

Bliuc D, Nguyen ND, Milch VE, Nguyen TV, Eisman JA, Center JR (2009) Mortality Risk Associated With Low-Trauma Osteoporotic Fracture and Subsequent Fracture in Men and Women. *JAMA* 301(5):513–521.

Bloom G, Standing H, Lloyd R (2008) Markets, Information Asymmetry and Health Care: Towards New Social Contracts. *Social Science & Medicine* 66(10):2076–2087.

Bouayad L, Padmanabhan B, Chari K (2020) Can Recommender Systems Reduce Healthcare Costs? The Role of Time Pressure and Cost Transparency in Prescription Choice. *MISQ* 44(4):1859–1903.

Brodersen KH, Ong CS, Stephan KE, Buhmann JM (2010) The Balanced Accuracy and Its Posterior Distribution. *Proceedings of the 20th International Conference on Pattern Recognition*. (IEEE, Istanbul, Turkey), 3121–3124.

Bruch E, Feinberg F, Lee KY (2016) Extracting multistage screening rules from online dating activity data. *Proc Natl Acad Sci USA* 113(38):10530–10535.

Brynjolfsson E, Hu Y (Jeffrey), Simester D (2011) Goodbye Pareto Principle, Hello Long Tail: The Effect of Search Costs on the Concentration of Product Sales. *Management Science* 57(8):1373–1386.

Canadian Institute for Health Information (2016) *Care in Canadian ICUs* (Canadian Institute for Health Information).

Chan CW, Farias VF, Escobar GJ (2017) The Impact of Delays on Service Times in the Intensive Care Unit. *Management Science* 63(7):2049–2072.

Chang Y, Shu L, Du E, Chen C, Zhang Z, Zheng Z, Huang Y, Xing X (2022) GraphRR: A multiplex Graph based Reciprocal friend Recommender system with applications on online gaming service. *Knowledge-Based Systems* 251:109187.

Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP (2002) SMOTE: Synthetic Minority Over-sampling Technique. *jair* 16:321–357.

Che Z, Purushotham S, Khemani R, Liu Y (2016) Interpretable Deep Models for ICU Outcome Prediction. *AMIA Annu Symp Proc* 2016:371–380.

Chen H, Chiang RHL, Storey VC (2012) Business Intelligence and Analytics: From Big Data To Big Impact. *MIS Quarterly* 36(4).

Chen T, Guestrin C (2016) XGBoost: A Scalable Tree Boosting System. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '16*. (ACM Press, San Francisco, California, USA), 785–794.

Choi E, Bahadori MT, Sun J, Kulas J, Schuetz A, Stewart W (2016) RETAIN: An Interpretable Predictive Model for Healthcare using Reverse Time Attention Mechanism. *Advances in Neural Information Processing Systems 29*. (Curran Associates, Inc., Barcelona, Spain), 3504–3512.

Choi MH, Kim D, Choi EJ, Jung YJ, Choi YJ, Cho JH, Jeong SH (2022) Mortality prediction of patients in intensive care units using machine learning algorithms based on electronic health records. *Sci Rep* 12(1):7180.

Cohen MC (2018) Big Data and Service Operations. *Prod Oper Manag* 27(9):1709–1723.

Corbett-Davies S, Goel S (2018) The Measure and Mismeasure of Fairness: A Critical Review of Fair Machine Learning. (August 14) http://arxiv.org/abs/1808.00023.

De-Arteaga M, Feuerriegel S, Saar-Tsechansky M (2022) Algorithmic fairness in business analytics: Directions for research and practice. *Production and Operations Management* 31(10):3749–3770.

Den Haan WJ, Ramey G, Watson J (2003) Liquidity flows and fragility of business enterprises. *Journal of Monetary Economics* 50(6):1215–1241.

Dong Y, Chawla NV, Swami A (2017) metapath2vec: Scalable Representation Learning for Heterogeneous Networks. *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. (ACM, Halifax NS Canada), 135–144.

Doshi-Velez F, Kim B (2018) Considerations for Evaluation and Generalization in Interpretable Machine Learning. *Explainable and Interpretable Models in Computer Vision and Machine Learning*. (Springer).

Duke G, Green J, Briedis J (2004) Survival of critically ill medical patients is time-critical. *Critical Care and Resuscitation* 6:261–7.

Duong CT, Hoang TD (2019) On Node Features for Graph Neural Networks.

Eichelsheim VI, Deković M, Buist KL, Cook WL, Johnson D (2009) The Social Relations Model in Family Studies: A Systematic Review. *Journal of Marriage and Family* 71(4):1052–1069.

Ellison N, Heino R, Gibbs J (2006) Managing Impressions Online: Self-Presentation Processes in the Online Dating Environment. *J Comp Mediated Comm* 11(2):415–441.

Fichman RG, Kohli R, Krishnan R eds. (2011) Editorial Overview - The Role of Information Systems in Healthcare: Current Research and Future Trends. *Information Systems Research* 22(3):419–428.

Friedman JH (2001) Greedy Function Approximation: A Gradient Boosting Approach. *The Annals of Statistics* 29(5):1189–1232.

Fu R, Aseri M, Singh PV, Srinivasan K (2022) "Un"Fair Machine Learning Algorithms. *Management Science* 68(6):4173–4195.

Gao M, Chen L, He X, Zhou A (2018) BiNE: Bipartite Network Embedding. *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. (ACM, Ann Arbor MI USA), 715–724.

Ghassemi M, Celi LA, Stone DJ (2015) State of the art review: the data revolution in critical care. *Crit Care* 19(1):118.

Ghose A, Kwon HE, Lee D, Oh W (2019) Seizing the Commuting Moment: Contextual Targeting Based on Mobile Transportation Apps. *Information Systems Research* 30(1):154–174.

Ghosh S, Feng M, Nguyen H, Li J (2016) Hypotension Risk Prediction via Sequential Contrast Patterns of ICU Blood Pressure. *IEEE Journal of Biomedical and Health Informatics* 20(5):1416–1426.

Glorot X, Bengio Y (2010) Understanding the difficulty of training deep feedforward neural networks. *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*. (JMLR Workshop and Conference Proceedings), 249–256.

Goodfellow I, Bengio Y, Courville A (2016) *Deep learning* (MIT Press).

Green C, Bonavia W, Toh C, Tiruvoipati R (2019) Prediction of ICU Delirium: Validation of Current Delirium Predictive Models in Routine Clinical Practice. *Critical Care Medicine* 47(3):428–435.

Gregor S, Benbasat I (1999) Explanations from Intelligent Systems: Theoretical Foundations and Implications for Practice. *MIS Quarterly* 23(4):497.

Gregor S, Hevner AR (2013) Positioning and Presenting Design Science Research for Maximum Impact. *MIS Quarterly* 37(2):337–355.

Gregor S, Jones D (2007) The anatomy of a design theory. *Journal of the Association of Information Systems*:312–335.

Grover A, Leskovec J (2016) node2vec: Scalable Feature Learning for Networks. *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*. (San Francisco, California, USA), 855–864.

Hall MJ, Levant S, DeFrances CJ (2013) Trends in Inpatient Hospital Deaths: National Hospital Discharge Survey, 2000–2010. *National Center for Health Statistics Data Brief*.

Hamilton WL, Ying R, Leskovec J (2017a) Inductive Representation Learning on Large Graphs. *31st Conference on Neural Information Processing Systems (NIPS 2017)*. (Long Beach, CA, US).

Hamilton WL, Ying R, Leskovec J (2017b) Representation Learning on Graphs: Methods and Applications. *IEEE Data Engineering Bulletin*.

Hamilton WL, Ying R, Leskovec J (2017c) Representation Learning on Graphs: Methods and Applications. *IEEE Data Engineering Bulletin*.

He K, Zhang X, Ren S, Sun J (2015) Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. *2015 IEEE International Conference on Computer Vision (ICCV)*. (IEEE, Santiago, Chile), 1026–1034.

He X, Deng K, Wang X, Li Y, Zhang Y, Wang M (2020) LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation. *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*. (ACM), 639–648.

He X, Liao L, Zhang H, Nie L, Hu X, Chua TS (2017) Neural Collaborative Filtering. *Proceedings of the 26th International Conference on World Wide Web*. (International World Wide Web Conferences Steering Committee), 173–182.

Hevner AR, March ST, Park J, Ram S (2004) Design Science in Information Systems Research. *MIS Quarterly* 28(1):75–105.

Hooker JN (2004) Is Design Theory Possible? *Journal of Information Technology Theory and Application* 6(2):73–83.

Huang X, Shan S, Khan YA, Salem S, Mohamed A, Attia EA (2022) Risk assessment of ICU patients through deep learning technique: A big data approach. *J Glob Health* 12:04044.

Hyland SL, Faltys M, Hüser M, Lyu X, Gumbsch T, Esteban C, Bock C, et al. (2020) Early prediction of circulatory failure in the intensive care unit using machine learning. *Nat Med* 26(3):364–373.

Iivari J (2007) A Paradigmatic Analysis of Information Systems As a Design Science. *Scandinavian Journal of Information Systems* 19(2):39–64.

Jain H, Padmanabhan B, Pavlou PA, Raghu TS (2021) Editorial for the Special Section on Humans, Algorithms, and Augmented Intelligence: The Future of Work, Organizations, and Society. *Information Systems Research* 32(3):675–687.

Jia Y, Bailey J, Ramamohanarao K, Leckie C, Houle ME (2019) Improving the Quality of Explanations with Local Embedding Perturbations. *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. (ACM, Anchorage AK USA), 875–884.

Johns E, Aodha OM, Brostow GJ (2015) Becoming the expert - interactive multi-class machine teaching. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (IEEE, Boston, MA, USA), 2616–2624.

Jung J, Bapna R, Ramaprasad J, Umyarov A (2019) Love Unshackled: Identifying the Effect of Mobile App Adoption in Online Dating. *MISQ* 43(1):47–72.

Jung J, Lim H, Lee D, Kim C (2022) The Secret to Finding a Match: A Field Experiment on Choice Capacity Design in an Online Dating Platform. *Information Systems Research* 33(4):1248–1263.

Kanis JA, Oden A, Johansson H, Borgström F, Ström O, McCloskey E (2009) FRAX® and its applications to clinical practice. *Bone* 44(5):734–743.

Kim Y, Han BG, the KoGES group (2017) Cohort Profile: The Korean Genome and Epidemiology Study (KoGES) Consortium. *International Journal of Epidemiology* 46(2):e20–e20.

Kim YS, Krzywicki A, Wobcke W, Mahidadia A, Compton P, Cai X, Bain M (2012) Hybrid Techniques to Address Cold Start Problems for People to People Recommendation in Social Networks. Anthony P, Ishizuka M, Lukose D, eds. *PRICAI 2012: Trends in*

Artificial Intelligence. Lecture Notes in Computer Science. (Springer Berlin Heidelberg, Berlin, Heidelberg), 206–217.

Kipf TN, Welling M (2017) Semi-Supervised Classification with Graph Convolutional Networks. *International Conference on Learning Representations 2017*.

Kiyotaki N, Wright R (1993) A Search-Theoretic Approach to Monetary Economics. *The American Economic Review* 83(1):63.

Kleinberg J, Mullainathan S, Raghavan M (2017) Inherent Trade-Offs in the Fair Determination of Risk Scores. *8th Innovations in Theoretical Computer Science Conference. Schloss Dagstuhl–LZI GmbH*. (Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik GmbH, Wadern/Saarbruecken, Germany, Wadern, Germany).

Kleinerman A, Rosenfeld A, Ricci F, Kraus S (2018) Optimally balancing receiver and recommended users' importance in reciprocal recommender systems. *Proceedings of the 12th ACM Conference on Recommender Systems*. (ACM, Vancouver British Columbia Canada), 131–139.

Koren Y, Bell R, Volinsky C (2009) Matrix Factorization Techniques for Recommender Systems. *Computer* 42(8):42–49.

Koster J, Leckie G (2014) Food sharing networks in lowland Nicaragua: An application of the social relations model to count data. *Social Networks* 38:100–110.

Kutty S, Chen L, Nayak R (2012) A people-to-people recommendation system using tensor space models. *Proceedings of the 27th Annual ACM Symposium on Applied Computing - SAC '12*. (ACM Press, Trento, Italy), 187.

Kwan VSY, Gosling SD, John OP (2008) Anthropomorphism as a special case of social perception: A cross-species social relations model analysis of humans and dogs. *Social Cognition* 26(2):129–142.

Laney D (2001) 3D data management: Controlling data volume, velocity and variety. *META group research note* 6(70).

Laugel T, Lesot MJ, Marsala C, Renard X, Detyniecki M, Laugel T, Lesot MJ, Marsala C, Renard X, Detyniecki M (2018) Comparison-Based Inverse Classification for Interpretability in Machine Learning. Bouchon-Meunier B, Yager RR, eds. *17th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU 2018)*. (Springer International Publishing, Cham), 100–111.

Laugel T, Renard X, Lesot MJ, Marsala C, Detyniecki M (2018) Defining Locality for Surrogates in Post-hoc Interpretablity. *Proceedings of the 2018 ICML Workshop on Human Interpretability in Machine Learning (WHI 2018)*. (Stockholm, Sweden).

Lee D, Gopal A, Park SH (2020) Different but Equal? A Field Experiment on the Impact of Recommendation Systems on Mobile and Personal Computer Channels in Retail. *Information Systems Research* 31(3):892–912.

Lee D, Hosanagar K (2019) How Do Recommender Systems Affect Sales Diversity? A Cross-Category Investigation via Randomized Field Experiment. *Information Systems Research* 30(1):239–259.

Leisman DE (2018) Rare Events in the ICU: An Emerging Challenge in Classification and Prediction. *Critical Care Medicine* 46(3):418–424.

Lewiecki EM, Ortendahl JD, Vanderpuye-Orgle J, Grauer A, Arellano J, Lemay J, Harmon AL, Broder MS, Singer AJ (2019) Healthcare Policy Changes in Osteoporosis Can Improve Outcomes and Reduce Costs in the United States. *JBMR Plus* 3(9):e10192.

Li RZ, Urbano J, Hanjalic A (2021) Leave No User Behind: Towards Improving the Utility of Recommender Systems for Non-mainstream Users. *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*. (ACM, Virtual Event Israel), 103–111.

Li Z, Fang X, Bai X, Sheng ORL (2017) Utility-Based Link Recommendation for Online Social Networks. *Management Science* 63(6):1938–1952.

Liao W, Bazarova NN, Yuan YC (2018) Unpacking Medium Effects on Social Psychological Processes in Computer-mediated Communication Using the Social Relations Model. *Journal of Computer-Mediated Communication* 23(2):90–106.

Lin YK, Chen H, Brown RA, Li SH, Yang HJ (2017) Healthcare Predictive Analytics for Risk Profiling in Chronic Care: A Bayesian Multitask Learning Approach. *MIS Quarterly* 41(2):473–495.

Liu LT, Dean S, Rolf E, Simchowitz M, Hardt M (2018) Delayed Impact of Fair Machine Learning. *Proceedings of the 35th International Conference on Machine Learning*. (PMLR), 3150–3158.

Lundberg SM, Lee SI (2017) A Unified Approach to Interpreting Model Predictions. *Advances in Neural Information Processing Systems 30*. (Curran Associates, Inc., Long Beach, US), 4765–4774.

Luo L, Liu K, Peng D, Ying Y, Zhang X (2020) A motif-based graph neural network to reciprocal recommendation for online dating. *International conference on neural information processing*.

Ma J, Zhao Z, Yi X, Chen J, Hong L, Chi EH (2018) Modeling Task Relationships in Multi-task Learning with Multi-gate Mixture-of-Experts. *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. (ACM, London United Kingdom), 1930–1939.

Manchikanti L, Falco FJE, Hirsch JA (2013) Ready or not! Here comes ICD-10. *J NeuroIntervent Surg* 5(1):86–91.

March ST, Smith GF (1995) Design and natural science research on information technology. *Decision Support Systems* 15(4):251–266.

Martens D, Provost F, Clark J, Fortuny EJ de (2016) Mining Massive Fine-Grained Behavior Data to Improve Predictive Analytics. *MIS Quarterly* 40(4):869–888.

McCradden MD, Joshi S, Mazwi M, Anderson JA (2020) Ethical limitations of algorithmic fairness solutions in health care machine learning. *The Lancet Digital Health* 2(5):e221–e223.

Mehrabi N, Morstatter F, Saxena N, Lerman K, Galstyan A (2021) A Survey on Bias and Fairness in Machine Learning. *ACM Comput. Surv.* 54(6):1–35.

Mikolov T, Sutskever I, Chen K, Corrado GS, Dean J (2013) Distributed Representations of Words and Phrases and their Compositionality. *Advances in Neural Information Processing Systems 26*. 3111–3119.

Mitchell TM (1980) *The Need for Biases in Learning Generalizations* (Laboratory for Computer Science Research, Rutgers Univ.).

Mortensen D, Pissarides CA (1994) Job Creation and Job Destruction in the Theory of Unemployment. *Review of Economic Studies* 61:397–415.

Nemati S, Holder A, Razmi F, Stanley MD, Clifford GD, Buchman TG (2018) An Interpretable Machine Learning Model for Accurate Prediction of Sepsis in the ICU: *Critical Care Medicine* 46(4):547–553.

Neve J, Palomares I (2019) Latent factor models and aggregation operators for collaborative filtering in reciprocal recommender systems. *Proceedings of the 13th ACM Conference on Recommender Systems*. (ACM, Copenhagen Denmark), 219–227.

Numata Y, Schulzer M, Wal RVD, Globerman J, Semeniuk P, Balka E, FitzGerald JM (2006) Nurse staffing levels and hospital mortality in critical care settings: literature review and meta-analysis. *Journal of Advanced Nursing* 55(4):435–448.

Nunamaker JF, Chen M, Purdin TDM (1990) Systems development in information systems research. *Journal of management information systems* 7(3):89–106.

Padmanabhan B, Fang X, Sahoo N, Burton-Jones A (2022) Machine Learning in Information Systems Research. *MIS Quarterly* 46(1).

Palomares I, Porcel C, Pizzato L, Guy I, Herrera-Viedma E (2021) Reciprocal Recommender Systems: Analysis of state-of-art literature, challenges and opportunities towards social recommendation. *Information Fusion* 69:103–127.

Paterek A (2007) Improving regularized singular value decomposition for collaborative filtering. *KDD Cup 2007*. 4.

Perozzi B, Al-Rfou R, Skiena S (2014) DeepWalk: Online Learning of Social Representations. *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '14*:701–710.

Petrongolo B, Pissarides CA (2001) Looking into the Black Box: A Survey of the Matching Function. *Journal of Economic Literature* 39(2):390–431.

Pizzato L, Rej T, Chung T, Koprinska I, Kay J (2010a) RECON: a reciprocal recommender for online dating. *Proceedings of the fourth ACM conference on Recommender systems*. 207–214.

Pizzato L, Rej T, Chung T, Koprinska I, Kay J (2010b) RECON: a reciprocal recommender for online dating. *Proceedings of the fourth ACM conference on Recommender systems*. (ACM, Barcelona Spain), 207–214.

Pollard TJ, Johnson AEW, Raffa JD, Celi LA, Mark RG, Badawi O (2018) The eICU Collaborative Research Database, a freely available multi-center database for critical care research. *Sci Data* 5(1):180178.

Powers DMW (1998) Applications and explanations of Zipf's law. *Proceedings of the Joint Conferences on New Methods in Language Processing and Computational Natural Language Learning - NeMLaP3/CoNLL '98*. (Association for Computational Linguistics, Sydney, Australia), 151.

Powers DMW (2011) Evaluation: From Precision, Recall, and F-measure to ROC, Informedness, Markedness & Correlation. *Journal of Machine Learning Technologies* 2(1):37–63.

Poyiadzi R, Renard X, Laugel T, Santos-Rodriguez R, Detyniecki M (2021) Understanding surrogate explanations: the interplay between complexity, fidelity and coverage. *arXiv:2107.04309 [cs]*.

Rai A (2016) Editor's comments: Synergies between big data and theory. *MIS Quarterly* 40(2):3–9.

Rai A (2017) Editor's Comments: Diversity of Design Science Research. *MIS Quarterly* 41(1).

Rai A (2020) Explainable AI: from black box to glass box. *J. of the Acad. Mark. Sci.* 48(1):137–141.

Reginster JY, Burlet N (2006) Osteoporosis: A still increasing prevalence. *Bone* 38(2):4–9.

Ribeiro MT, Singh S, Guestrin C (2016) "Why Should I Trust You?": Explaining the Predictions of Any Classifier. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. (ACM, San Francisco, California, USA).

Rindflesch TC, Kilicoglu H, Fiszman M, Rosemblat G, Shin D (2011) Semantic MEDLINE: An advanced information management application for biomedicine. *ISU* 31(1–2):15–21.

Roulin N, Levashina J (2016) Impression management and social media profiles. *Social Media in Employee Selection and Recruitment Theory, Practice, and Current Challenges*. 223–248.

Rowan KM, Kerr JH, Major E, McPherson K, Short A, Vessey MP (1993) Intensive Care Society's APACHE II study in Britain and Ireland--II: Outcome comparisons of intensive care units after adjustment for case mix by the American APACHE II method. *BMJ* 307(6910):977–981.

Safaei N, Safaei B, Seyedekrami S, Talafidaryani M, Masoud A, Wang S, Li Q, Moqri M (2022) E-CatBoost: An efficient machine learning framework for predicting ICU mortality using the eICU Collaborative Research Database Chen RJ, ed. *PLoS ONE* 17(5):e0262895.

Saito T, Rehmsmeier M (2015) The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets Brock G, ed. *PLoS ONE* 10(3):e0118432.

Salluh JIF, Soares M (2014) ICU severity of illness scores: APACHE, SAPS and MPM. *Current Opinion in Critical Care* 20(5):557–565.

Sanchez-Pinto LN, Luo Y, Churpek MM (2018) Big Data and Data Science in Critical Care. *Chest* 154(5):1239–1248.

Sheetrit E, Nissim N, Klimov D, Shahar Y (2019) Temporal Probabilistic Profiles for Sepsis Prediction in the ICU. *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. (ACM, Anchorage AK USA), 2961–2969.

Shin D, He S, Lee GM, Whinston AB, Cetintas S, Lee KC (2020) Enhancing Social Media Analysis with Visual Data Analytics: A Deep Learning Approach. *MISQ* 44(4):1459–1492.

Shmueli G, Koppius OR (2011) Predictive Analytics in Information Systems Research. *MIS Quarterly* 35(3):553–572.

Simon H (1996) *The sciences of the artificial* (MIT press).

Snoek J, Larochelle H, Adams RP (2012) Practical Bayesian Optimization of Machine Learning Algorithms. *Advances in Neural Information Processing Systems (NIPS)*.

Son J, Flatley Brennan P, Zhou S (2020) A Data Analytics Framework for Smart Asthma Management Based on Remote Health Information Systems with Bluetooth-Enabled Personal Inhalers. *MISQ* 44(1):285–303.

Srinivasan K, Currim F, Ram S (2018) Predicting High-Cost Patients at Point of Admission Using Network Science. *IEEE Journal of Biomedical and Health Informatics* 22(6):1970–1977.

Su C, Tong J, Zhu Y, Cui P, Wang F (2020) Network embedding in biomedical data science. *Briefings in Bioinformatics* 21(1):182–197.

Sun Y, Han J, Yan X, Yu PS, Wu T (2011) PathSim: meta path-based top-K similarity search in heterogeneous information networks. *Proc. VLDB Endow.* 4(11):992–1003.

Sunar N, Swaminathan JM (2022) Socially relevant and inclusive operations management. *Production and Operations Management* 31(12):4379–4392.

Thorsen-Meyer HC, Nielsen AB, Nielsen AP, Kaas-Hansen BS, Toft P, Schierbeck J, Strøm T, et al. (2020) Dynamic and explainable machine learning prediction of mortality in patients in the intensive care unit: a retrospective study of high-frequency data in electronic patient records. *The Lancet Digital Health* 2(4):e179–e191.

Tong ST, Corriero EF, Wibowo KA, Makki TW, Slatcher RB (2020) Self-presentation and impressions of personality through text-based online dating profiles: A lens model analysis. *New Media & Society* 22(5):875–895.

Unal AU, Kostek O, Takir M, Caklili O, Uzunlulu M, Oguz A (2015) Prognosis of patients in a medical intensive care unit. *North Clin Istanb* 2(3):189–195.

Veličković P, Cucurull G, Casanova A, Romero A, Liò P, Bengio Y (2018) Graph Attention Networks. *International Conference on Learning Representations 2018*.

Wallace DJ, Angus DC, Barnato AE, Kramer AA, Kahn JM (2012) Nighttime Intensivist Staffing and Mortality among Critically Ill Patients. *N Engl J Med* 366(22):2093–2101.

Wang W, Chen Junyang, Wang J, Chen Junxin, Liu J, Gong Z (2020) Trust-Enhanced Collaborative Filtering for Personalized Point of Interests Recommendation. *IEEE Transactions on Industrial Informatics* 16(9):6124–6132.

Wang X, He X, Wang M, Feng F, Chua TS (2019) Neural Graph Collaborative Filtering. *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*:165–174.

Wieting J, Bansal M, Gimpel K, Livescu K (2016) Towards Universal Paraphrastic Sentence Embeddings. *Proceedings of the 4th International Conference on Learning Representations*. (San Juan, Puerto Rico).

Wollschlager CM, Conrad AR (1988) Common complications in critically ill patients. *Disease-a-month* 34(5):225–293.

Xia P, Liu B, Sun Y, Chen C (2015) Reciprocal recommendation system for online dating. *2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*. 234–241.

Xue GR, Lin C, Yang Q, Xi W, Zeng HJ, Yu Y, Chen Z (2005) Scalable collaborative filtering using cluster-based smoothing. *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval - SIGIR '05*. (ACM Press, Salvador, Brazil), 114.

Yang Y, Morillo IG, Hospedales TM (2018) Deep Neural Decision Trees. *Proceedings of the 2018 ICML Workshop on Human Interpretability in Machine Learning (WHI 2018)*.

Yi HC, You ZH, Huang DS, Kwoh CK (2022) Graph representation learning in bioinformatics: trends, methods and applications. *Briefings in Bioinformatics* 23(1):1–16.

Ying R, He R, Chen K, Eksombatchai P, Hamilton WL, Leskovec J (2018) Graph Convolutional Neural Networks for Web-Scale Recommender Systems. *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*:974–983.

Young MP, Gooder VJ, McBride K, James B, Fisher ES (2003) Inpatient transfers to the intensive care unit: Delays are associated with increased mortality and morbidity. *J Gen Intern Med* 18(2):77–83.

Zhang M, Chen Y (2020) Inductive Matrix Completion Based on Graph Neural Networks. *International Conference on Learning Representations 2020*.

Zhang W, Ram S (2020) A Comprehensive Analysis of Triggers and Risk Factors for Asthma Based on Machine Learning and Large Heterogeneous Data Sources. *MISQ* 44(1):305–349.

Zhu Z, He Y, Zhao X, Zhang Y, Wang J, Caverlee J (2021) Popularity-Opportunity Bias in Collaborative Filtering. *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*. (ACM, Virtual Event Israel), 85–93.

Zimmerman JE, Kramer AA, Knaus WA (2013) Changes in hospital mortality for United States intensive care unit admissions from 1988 to 2012. *Crit Care* 17(2):R81.

Zimmerman JE, Kramer AA, McNair DS, Malila FM (2006) Acute Physiology and Chronic Health Evaluation (APACHE) IV: Hospital mortality assessment for today's critically ill patients*: *Critical Care Medicine* 34(5):1297–1310.