Design and Evaluation of Factorization-Based Algorithms for Recommendation Systems

Yali Du

Faculty of Engineering and Information Technology

University of Technology Sydney

This thesis is submitted for the degree of

Doctor of Philosophy

July 2019

I would like to dedicate this thesis to my loving parents

for letting me pursue my dream

for so long

so far away from home

Certificate of Original Authorship

I, Yali Du declare that this thesis, is submitted in fulfilment of the requirements for the award of Doctor of Philosophy, in the Faculty of Engineering and Information Technology at the University of Technology Sydney. This thesis is wholly my own work unless otherwise reference or acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis. This document has not been submitted for qualifications at any other academic institution. This research is supported by the Australian Government Research Training Program.

Acknowledgements

First and foremost, I am greatly thankful for my adviser, Dacheng Tao, for his consistent support and guidance during my PhD, and for providing me invaluable advice on many topics. His high standard in research helps me to inspect and improve my work all the time. This thesis would not have been completed without his support. I am also thankful for my undergraduate advisers Shenggui Zhang and Yufeng Nie for encouraging me to pursue a career in research.

I would like to extend my gratitude to Dr. Yong Luo for introducing me to the research career in artificial intelligence, Dr. Jinfeng Yi, for introducing me to adversarial machine learning research. For the work in this thesis, I enjoyed working with Chang Xu, Meng Fang, Jinfeng Yi, Jun Cheng, Rao Kotagiri. I appreciate their brilliant work and timely support. Especially, I would like to thank Dr. Chang Xu for his consistent efforts and fruitful discussions for improving this thesis.

I am grateful to have the opportunity of working in a group with many intelligent minds. I benefit a lot from discussing and working with them. They are Maoying Qiao, Liu Liu, Xiyu Yu, Huan Fu, Chaoyue Wang, Zhongwen Xu, Tongliang Liu, Mingming Gong, Baosheng Yu, Shaoli Huang, Zhe Chen, Zhibin Hong, Jiang Bian, Jiankang Deng, Jiayan Qiu, Jue Wang, Changxing Ding, Ruxin Wang; and faculty members, Professor Ivor Tsang, Professor Yi Yang, Dr. Liang Zheng, Dr. Lin Chen.

I am also grateful for all the other friends who made this journey unforgettable: Xun Yang, Chen Gong, Hongshu Chen, Hua Zuo, Yuangang Pan, Jianfeng Dong, Erkun Yang, Shan You, Yuxuan Du, Sujuan Hou, Yiliao Song, Xiaoqing Yin, Liping Xie, Long Lan, Xianye Ben, Wankou Yang, Shigang Liu, Qiang Li, Tao Lei. I am also thankful for my

housing roommates Tim Harper, Faris, Firas, Yuta, Ann, Vishal, Annia, Eric Wang, Cassie Maye, Pingo, David, Sahil who distracted me from my work and made my housing life happier than I can imagine. I would like to especially thank Donna Xu, Qian Zhang, Xinyuan Chen, and Yan Yan, who have given me support during both cheery and stressful times.

Finally, this thesis is dedicated to my family, for everything else, and Moe, for always being there.

Abstract

Recommendation systems (RecSys) are valuable for both industry and customers in many fields, including e-commerce and social media. Despite the great demand for such effective systems, many challenges still exist. A major obstruction is the sparsity and poor quality of data that hinder the learning of a satisfactory RecSys. Another obstacle lies in the open nature of RecSys: this poses a threat to their safety in applications.

In this thesis, we work towards meeting these two challenges. To improve RecSys performance, we study how to exploit information from user reviews, constraints on user behaviors and user/items demographic features. Three approaches are proposed: 1) we develop a privileged matrix factorization model that exploits reviews for the learning of both user/item factors; 2) we build a collaborative allocation model that investigates the geometric constraint on the user-preference matrix; 3) given that the features might be noisy in reality, we propose an approach to identifying noisy information and selecting useful side features.

Driven by concern for the security of RecSys, our first consideration is to develop an evaluation method for testing the robustness of target models before proposing an approach to improve their resistance to malicious attacks. The target model is evaluated by measuring the minimal number of features required to mis-predict a user's preference. To enhance the robustness of target models, we inject noise in the training phase to enforce resistance to perturbations. Target models are further guided by standard networks through the distillation of generalized knowledge to avoid performance degeneration. This way, the target model becomes more resistant to adversarial perturbations while still achieving similar performances to standard models.

We conclude the thesis by outlining main contributions and indicating primary results.

Table of contents

List of figures xv

List of tables xvii

1 Introduction 1

1.1 Designing RecSys with Features 2

1.1.1 Learning representations from reviews 2

1.1.2 Exploiting rating allocation for users 3

1.1.3 Learning from noisy side information 3

1.2 Evaluating and Improving the Robustness of RecSys 4

1.2.1 Evaluating the robustness of RecSys 4

1.2.2 Improving the robustness of RecSys 5

1.3 Summary 5

1.4 Publications 6

2 Learning Representations from Reviews 7

2.1 Introduction 7

2.2 Related Work 10

2.3 Privileged Matrix Completion 11

2.3.1 Problem statement 11

2.3.2 Privileged matrix factorization 11

2.4 Optimization 15

2.5 Experiments 17

2.5.1 Experimental settings 18
2.5.2 Baseline methods 18
2.5.3 Evaluation results 19
2.5.4 Parameter analysis 21
2.6 Conclusions 22

3 Exploiting Rating Allocation for Users 23
3.1 Introduction 23
3.2 Related Work 25
3.3 Problem Formulation 26
3.4 Distance Metric on Histogram Data 29
3.4.1 Pullback metric from Sphere 29
3.4.2 Euclidean maps of histogram data 30
3.5 Optimization 34
3.5.1 Euclidean gradients with metric on Simplex 35
3.5.2 Euclidean gradients with Aitchison mappings 36
3.5.3 Conjugate gradients with the Armijo line search method 38
3.5.4 Updating **X** on Simplex 41
3.6 Experiments 42
3.6.1 Image data 44
3.6.2 Recommendation data 45
3.6.3 Comparisons of different metrics 48
3.7 Conclusions 51

4 Learning from Noisy Side Information 53
4.1 Introduction 54
4.2 Related Work 56
4.3 Problem Formulation 57
4.4 Optimization 59
4.5 Recovery Analysis 63

4.5.1 Sampling complexity for exact recovery . . . 63
4.5.2 Sampling complexity for ε-recovery . . . 66
4.6 Experiments . . . 68
4.6.1 Baseline methods . . . 68
4.6.2 Evaluation on real data . . . 69
4.7 Proof Details . . . 72
4.7.1 Proof of Theorem 1 . . . 72
4.7.2 Proof of Theorem 2 . . . 77
4.8 Conclusions . . . 81

5 Evaluating the Robustness of RecSys 83
5.1 Introduction . . . 83
5.2 Related Work . . . 85
5.3 Proposed Framework . . . 86
5.3.1 Preliminaries . . . 86
5.3.2 The proposed threat model . . . 87
5.3.3 Random attack as a baseline . . . 91
5.4 Experiments . . . 92
5.4.1 Experimental settings . . . 92
5.4.2 Overall evaluation over different target systems . . . 95
5.4.3 Analysis of vulnerable features . . . 96
5.4.4 Attack under different number of permissible variables . . . 96
5.4.5 Attacks under different latent factors . . . 97
5.5 Conclusions and Discussions . . . 98

6 Enhancing the Robustness of RecSys Under Malicious Attacks 99
6.1 Introduction . . . 100
6.2 Related Work . . . 102
6.3 Preliminaries . . . 103
6.3.1 Collaborative filtering . . . 104

6.3.2 Adversarial attacks 106
6.4 Stage-wise Hints Training for Robust Neural Collaborative Filtering 107
6.4.1 Knowledge transfer from a teacher 107
6.4.2 Stage-wise hints training of the student model 109
6.4.3 Randomness by injecting noises 110
6.4.4 Relation to other works 111
6.5 Analysis of Stage-Wise Hints Training Strategy 111
6.5.1 Robustness of neural collaborative filtering system 112
6.5.2 Impact on model sensitivity to inputs 114
6.5.3 Impact on model generalizability 115
6.6 Experiments 115
6.6.1 Dataset information 116
6.6.2 Overview of the experimental setup 117
6.6.3 Influence on model's generalizability 119
6.6.4 Impact on adversarial perturbations 121
6.6.5 Model's robustness under different noise levels 122
6.7 Conclusions 122

7 Conclusions **123**

References **127**

List of figures

2.1 Box and whisker plot of different random split of data. Center line represents median. Box extents show first quarter and third quarter. Whisker extents illustrate maximum and minimum values 20

2.2 Mean square error of PriMF changing with α and β on different datasets . . 21

3.1 An illustration of the distance on the simplex. 26

3.2 Classification results vary with sampling fraction on MIT Scene and UIUC Scene. 44

3.3 MovieLens 1M: NMSE and NMAE changing with latent dimensions between 10 and 1000 under weak and strong generalization. 49

3.4 MovieLens 1M: NMSE and NMAE as objective decreasing with iterations under weak and strong generalization. r refers to latent factors of 10, 50, 100, 200, 300, 400, 500. 50

3.5 Classification results vary with sampling fraction on MIT Scene and UIUC Scene. 51

4.1 Recovered W and S under different sampling ratio p of NCI Challenge dataset 69

4.2 Recovered W and S under different sampling ratio p of MovieLens 100K dataset 71

5.1 Overview of a deep learning-based recommendation system based on Frappe dataset and its potential vulnerability to unnoticeable changes. The listed features including user ID, item ID are encoded into vectors of a fixed length. By alternating a feature from "Monday" to "Tuesday", the model's prediction changes dramatically. 88

5.2 Bin frequencies for context variables under attack-1 and attack-2. In a), we plot the bin frequencies of the perturbed variables in successful attacks in attack-1. In b), each valid attack may contain one or two perturbed variables. We group all the perturbed variables and plot the bin frequencies. 95

5.3 Success rate of *Random Attack* and our threat model on Frappe dataset with different numbers of features being perturbed. The latent factors are set to be 64 for two target models. 96

5.4 Evaluation of the robustness of NFM and Wide&Deep by our proposed threat model under different latent factors. 97

6.1 Overview of a neural collaborative filtering architecture. It encodes user and item features, then uses multi-layer perceptrons to predict ratings. 105

6.2 The graph illustration of student training procedure: (i) Step 1 is the hints training process which aligns the output between intermediate layers of teachers and students; θ_r denotes the regression parameters that align the output dimension of teacher and students; (ii) Step 2 is the knowledge distillation training process with noise layers added before MLP layers. . . 108

6.3 The stage-wise hints training framework. Steps 1 to m are the hints training routine that is illustrated in Figure 6.2 but need to be repeated m times for m different hints modules from the teacher. 109

6.4 The graph illustration of vulnerabilities of the collaborative filtering system 112

6.5 Comparisons of FNCF with FNCF-Single, FNCF-Multi and FNCF-Distill at temperature $T = 10$ and noise level $\sigma = 0.05$. 119

6.6 Attack success rate on FNCF, FNCF-Single, FNCF-Multi and FNCF-Distill at temperature $T = 10$ and noise level $\sigma = 0.05$. 120

List of tables

2.1 Dataset Information 18
2.2 Mean square error comparisons with baseline methods on different datasets. Values in brackets indicate standard deviation error. 20

3.1 Dataset information 43
3.2 Comparisons of different collaborative filtering methods in terms of RMSE ($\times 10^{-3}$) and NMAE ($\times 10^{-3}$) 43
3.3 Dataset Information 47
3.4 MovieLens 100K: Comparisons with baselines in terms of NMSE and NMAE 47
3.5 MovieLens 1M: Comparisons with baselines in terms of NMSE and NMAE 47
3.6 EachMovie: Comparisons with baselines in terms of NMSE and NMAE . . 48
3.7 Comparisons of different Aitchison embeddings for histogram data in terms of RMSE ($\times 10^{-3}$) and NMAE ($\times 10^{-3}$) 49
3.8 Comparisons of classification accuracy under different Aitchison embeddings with exact and relaxed recovery constraints 51

4.1 Comparisons of relative MSE of different methods under different splits of NCI Challenge dataset 70
4.2 Comparisons of relative MSE of different methods under different splits of the MovieLens 100K dataset 70

5.1 Dataset information 92

5.2 RMSE of the evaluated models under different latent factors on Frappe and MovieLens on test set. 93
5.3 Overall evaluation of base models on Frappe and Movielens. The number of permissible features are two for Frappe and one for Movielens. “#param” denotes the number of trainable parameters and “M” represents “one million". “Rand” represents *Random Attack*. 95
5.4 Comparison between *Random attack* and our attack on MovieLens dataset while attack two base models. The latent factors are set to be 64 for both models. 96

6.1 Overview of architectures of the teacher and student model. The second last dense layer determines the latent factor for the neural collaborative filtering model which is 64 and 32 for teacher and student model in this table. 116
6.2 Overview of hyperparameters for training the model. For two stage hints training of student model, we train the model by 10 epochs in each stage and 10 epocs in final knowledge distillation training. 117
6.3 Different models’ performance under different noise levels at temperature $T = 10$. 120
6.4 Model’s robustness against adversarial perturbations under different noise levels at temperature $T = 10$. 121