

UC Santa Cruz

UC Santa Cruz Electronic Theses and Dissertations

Title

Economics in E-Commerce Recommendation: Producer, Consumer and Platform

Permalink

<https://escholarship.org/uc/item/9vq8h87h>

Author

Zhao, Qi

Publication Date

2016

Copyright Information

This work is made available under the terms of a Creative Commons Attribution License, available at <https://creativecommons.org/licenses/by/4.0/>

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA
SANTA CRUZ

**ECONOMICS IN E-COMMERCE RECOMMENDATION:
PRODUCER, CONSUMER AND PLATFORM**

A dissertation submitted in partial satisfaction of the
requirements for the degree of

DOCTOR OF PHILOSOPHY

in

TECHNOLOGY AND INFORMATION MANAGEMENT

by

Qi Zhao

September 2016

The Dissertation of Qi Zhao
is approved:

Professor Yi Zhang, Chair

Professor Daniel Friedman

Professor David Helmbold

Dean Tyrus Miller
Vice Provost and Dean of Graduate Studies

Copyright © by

Qi Zhao

2016

Table of Contents

List of Figures	vi
List of Tables	viii
Abstract	x
Dedication	xii
Acknowledgments	xiii
1 Introduction	1
1.1 What is special about product recommendation?	1
1.2 Economics in E-commerce recommendation: producer, consumer and platform	5
1.3 The goal of the dissertation and overview of the solutions	9
1.4 Contributions of the dissertation	10
1.5 Outline	13
2 Literature review	15
2.1 Recommender systems	15
2.1.1 Collaborative filtering (CF)	16
2.1.2 Content based filtering	19
2.1.3 Hybrid recommendation algorithms	20
2.2 Evaluation metrics	21
2.2.1 Setup	21
2.2.2 Root mean squared error (RMSE)	22
2.2.3 Ranking based metrics	22
2.3 Economics Basics	23
2.3.1 Utility	23
2.3.2 Utility function	24
2.3.3 Cardinal utility	25
2.3.4 Utility function	26

2.3.5	Surplus	29
3	For producer: recommendation by personalized promotion	32
3.1	Methods	34
3.1.1	Bidding & Lottery Procedures	34
3.1.2	Personalized Promotion	37
3.2	Experimental procedures	38
3.2.1	Data collection	38
3.2.2	Subjects' training and selection	41
3.2.3	Data collection	41
3.2.4	Evaluation Metrics	42
3.3	Experimental Results	42
3.3.1	Further Analysis	46
3.4	Caveats	47
3.5	Related Work	48
3.6	Summary	51
3.7	Acknowledgment	52
4	For consumer: Multi-product utility maximization based recommendation	54
4.1	Basic Components	56
4.1.1	Indifference Curves	56
4.2	Multiple Product Utility Maximization Framework	58
4.2.1	Modeling Marginal Rate of Substitution	58
4.2.2	Multi-product Utility Modeling	61
4.2.3	CF-based Reparameterization	62
4.2.4	Discrete Choice Modeling	62
4.2.5	Budget Constraint	64
4.2.6	Model Parameter Learning	64
4.3	Multi-product Recommendation	65
4.4	Experiment	67
4.4.1	Dataset Description	67
4.4.2	Evaluation Metric	70
4.4.3	Experimental Results	70
4.4.4	Further Analysis: Empirical Study of Economic Intuition	72
4.5	Related Work	73
4.6	Summary and Future Work	76
4.7	Acknowledgment	78
5	For platform: Total Surplus Maximization based recommendation	79
5.1	The Framework for OSA	84
5.1.1	Problem Formalization	85
5.1.2	Personalized Utility	86

5.1.3	Total Surplus Maximization	88
5.2	Model Specification	89
5.2.1	E-commerce	89
5.2.2	Online Peer-to-Peer Lending	93
5.2.3	Online Freelancing Platforms	95
5.2.4	Remarks	98
5.3	Results	100
5.3.1	E-commerce Dataset Description	100
5.3.2	Parameter Selection and Estimation	101
5.3.3	Purchase Prediction and Recommendation	104
5.3.4	Evaluation on Total Surplus	107
5.3.5	P2P Lending Networks	109
5.3.6	Online Freelancing	112
5.4	Related Work	115
5.5	Discussion	117
5.5.1	Temporal Factor	117
5.5.2	Opportunity Cost	120
5.6	Summary	122
5.7	Acknowledgment	124
6	Conclusion and Future Work	125
6.1	Contribution	125
6.1.1	General contributions	126
6.1.2	Specific contributions	128
6.2	Limitations and Future Work	130
6.2.1	On WTP elicitation	130
6.2.2	On multi-product utility modeling	132
6.2.3	On personalized utility modeling and total surplus maximization	134
	Bibliography	135

List of Figures

1.1	Typical architecture of a product recommender system. The recommender systems take feedback from the end users, run the recommendation algorithm and update the representation of the users. The system then recommends a list of items that might interest the user based on current user's representation.	3
2.1	Illustrative indifference curves for common pairs, perfect substitutional pairs, and perfect complementary pairs. The utilities of the three illustrative curves satisfy $I_1 < I_2 < I_3$	26
2.2	An intuitive explanation of surplus derived from marginal utility and marginal cost.	30
3.1	Overview of the experiment flow. The purpose of step 1 is to collect information about current user. The information will be used for later modeling.	39
4.1	Illustrative indifference curves for common pairs, perfect substitutional pairs, and perfect complementary pairs. The utilities of the three illustrative curves satisfy $I_1 < I_2 < I_3$	56
4.2	Scattered relations of the product popularity v.s. the average Elasticity of Substitution (ES) of the corresponding product as well as the ranking of ES values.	74
5.1	Illustration of total surplus for free market where the price is the equilibrium price. The total surplus is the sum of the consumer surplus and producer surplus.	80
5.2	Comparison of the recommendation performance for CF and TSM [?] . The y-axis is the conversion rate, and the x-axis is the length N of each recommendation list.	105
5.3	Comparison on Total Surplus (TS) for CF and TSM [?] . Note that TS@N means the total social surplus of the top N recommendations when they are all accepted by the corresponding consumer.	108

5.4 P2P lending problem considering temporal factor. In the graph, t_1 and t_2 denote two different time points. At each time point, a set of active loan requests and their potential lenders are identified. The relationship between loans and lenders can be described by a bipartite graph in which the nodes are loans and lenders and an edge is placed between a loan and lender if the lender is likely to invest the loan. 118

List of Tables

2.1	Different ways to assign utilities. Each column corresponds to a utility assignment and all assignments describe the bundle preference relationship: $A \succ B \succ C$	24
2.2	Commonly used cardinal utility function	28
3.1	The features used to predict WTP modeling, and their weight coefficients learned by Equation 3.7. Please note that the <i>brand variety</i> feature and <i>switch brand</i> feature might interact, as consumers who are loyal with one brand are less likely to switch to another brand at the bidding stage, and vice versa. In this paper, the small number of data samples limits us to linear regression models. In the future, it would be interesting to explore other models which can better handle feature interactions, e.g. decision trees.	43
3.2	Performance of baseline algorithms and our linear regression algorithm in 10-fold cross validation setting. The baseline algorithms are flat discount of original sale price at different rates. Profit is evaluated w.r.t different production cost assumptions. The production costs are expressed as percentage of original sale price. For each production cost assumption, the best result of the baseline algorithms is in bold.	44
4.1	Choices of $h(q_k, q_j)$ and its respective utility function. $z(\cdot)$ denotes any monotonic function. Please refer to text for more details.	59
4.3	Basic statistics of the two datasets	67
4.2	Evaluation results for Top- K recommendation performance on Precision, Recall, and F_1 -measure. The baseline algorithms are <i>Bayesian Personalized Ranking</i> (BPR) and <i>Matrix Factorization</i> based <i>Collaborative Filtering</i> (CF) as shown in Equation 2.6. Please refer to Section 4.4.3 for more information about BRP.	68
4.4	Number of model parameters for each algorithm.	70
5.1	Specifications for three exemplary applications of the Total Surplus Maximization framework.	85

5.2	Statistics of the Shop.com dataset.	101
5.3	Evaluation on Conversion Rate (CR@N) and Total Surplus (TS@N) for Top-N recommendation, where TSM^* stands for our TSM approach with regularization coefficient $\eta = *$ in Eq.(5.10).	102
5.4	Summary of parameters. The number of latent factors K and the CF regularization coefficient λ are identified by cross validation, and are fixed throughout the reported results; η varies so as to examine its influence; c_j is the marginal cost of product j	104
5.5	Statistics of the selected Prosper dataset, where ‘rate’ represents the interest rate of a loan.	110
5.6	Results on total surplus with and without our Total Surplus Maximization (TSM) framework.	110
5.7	Some key statistics of the ZBJ dataset.	112
5.8	Conversion rate on job recommendation.	113
5.9	Total surplus of online freelancing job allocations under typical choices of latent factor K	114

Abstract

Economics in E-commerce Recommendation: Producer, Consumer and Platform

by

Qi Zhao

Product recommender systems have become increasingly important, as consumers are exposed to massive amounts of product information on the Internet and product recommender systems can suggest products that interest consumers. A lot of work has been done over the past few decades to develop recommendation algorithms. Most existing algorithms reduce the recommendation problem to rating prediction and measure the recommendation quality by Root Mean Squared Error (RMSE). The existing algorithms have proven successful in a wide range of applications including movie , news and music recommendations.

However, when it comes to product recommendation, there are specific aspects the recommendation algorithm needs to consider. First, price is important and the recommendation algorithm needs to consider consumer's *Willingness-to-Pay* (WTP); second, it is important to consider inter-product relationships; third, it is necessary to take account of the benefits to both consumer and producer. Without such consideration, recommendation algorithms might perform poorly. For example, the recommendations are too expensive for the consumer to buy, or the recommendations do not complement the products the consumer already has, or the system prices the products in favor of the consumers with little consideration of the benefit to producers. Unfortunately, these

three aspects are not well addressed in existing recommendation algorithms and pose challenges to product recommendation.

In this dissertation, we propose to address these aspects by leveraging well-established economic principles. In particular, we adopt *producer surplus*, *consumer surplus* and *total surplus* to represent the benefits of producer, consumer and the platform, respectively. For producers, we propose to elicit consumer WTP in the E-commerce setting and perform personalized promotion based on the estimated WTP. The goal of personalized promotion is to maximize producer's profit. For consumers, we propose to recommend by multi-product utility maximization. Our proposed method can automatically learn the relationship from real-world transaction data. For the platform, we propose and implement a *Total Surplus Maximization* (TSM) based recommendation framework, in which the benefits of both producer and consumer are considered. The TSM framework can conveniently specialize into several typical applications, including E-commerce, P2P lending and the online freelancer market. The proposed methods have been evaluated in real-world datasets and the experimental results demonstrate that the proposed methods are advantageous to existing algorithms.

To my parents & Yuting

Acknowledgments

Firstly, I would like to express my sincere gratitude to my advisor Professor Yi Zhang for the continuous support of my Ph.D study and related research, for her patience, motivation, and immense knowledge. Her guidance helped me in all the time of research and writing of this thesis.

Besides my advisor, I would like to thank the rest of my thesis committee: Professor Daniel Friedman and Professor David Helmbold for their insightful comments and encouragement, but also for the challenging questions which motivates me to improve the dissertation to the next level.

I am grateful for my parents for their unconditional support in all my pursuits. They are the unsung hero. Without their support, I would not have been where I am now.

Last but not least, I would like to thank Yuting, without whom this dissertation would not have been possible. It is her encouragement that aspires me to overcome all the barriers along the way. Every time I found myself in difficulty, she talked to me patiently, helped me to make the right decision and encouraged me to achieve the goal. I believe I am one of the luckiest men in this world because of her. Words cannot describe my gratitude to her. Thank you!

Qi Zhao

Aug. 16th, 2016

Chapter 1

Introduction

1.1 What is special about product recommendation?

Recommender systems have achieved success in a range of applications and become increasingly popular over the past few decades. For example, Netflix's recommendation engine is reported to drive about 75% of the website's traffic. Besides movies, recommender systems are also widely adopted in news (Google News), music (Pandora Inc.) and E-commerce shopping (Amazon Inc.).

Regardless of the specific application, the primary objective of recommender systems is to suggest items to users in a proactive manner. Recommendation algorithms achieve this objective by learning a user's preferences based on the user's feedback, and predicting on unseen items. Figure 1.1 illustrates a typical architecture of product recommender systems. A typical form of prediction is rating prediction, where recommendations are simply the highest rated items. Rating prediction-based recommendation

has been quite successful in applications like movie recommendation, but it might not be a fit for product recommendation. Consider the following cases:

Case 1: *A customer purchases a camera on an E-commerce website and the system recommends the customer a lens that matches the camera. The customer likes the lens but hesitates to buy it because the price is too high.*

Case 2: *Again, a customer buys a camera, but this time the E-commerce website recommends another camera instead of a lens. Though the recommendation might look interesting, the customer is not likely to buy another camera.*

Case 3: *The situation is the same as Case 1, except that the E-commerce platform offers a huge promotion in order to incentivize the customer to buy the product. Such strategy favors the customer at the possible cost of sacrificing the producer's benefit.*

In Case 1, the recommendation sounds reasonable, as the recommended item is highly relevant and the customer is likely to be interested in the product. Further-more, if the customer is asked to rate the recommendation, the rating is likely to be high, so it counts as a good recommendation for the rating prediction-based recommendation algorithm. However, for E-commerce product recommendation, price is important, and could dominate the customer's decision on whether or not to purchase the recommended product. The situation is very different from other applications where the price matters little. Take Netflix movie recommendations for example: all movies cost the same, so the quality of recommendation is more about how well the recommendation is aligned with the customer's preference. In order to convince the customer to purchase the recommendation on the E-commerce platform, the price needs to be properly adjusted.

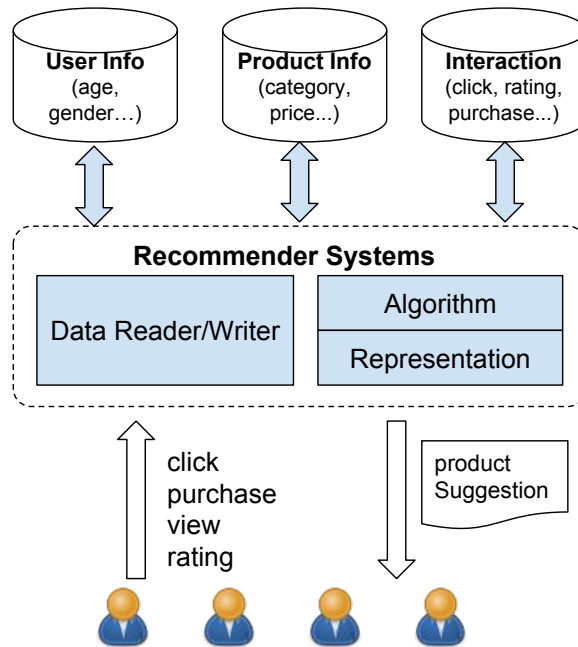


Figure 1.1: Typical architecture of a product recommender system. The recommender systems take feedback from the end users, run the recommendation algorithm and update the representation of the users. The system then recommends a list of items that might interest the user based on current user’s representation.

A natural question would be: “what should the price be and how should it be tailored to individuals?”. This is one of the main research questions this dissertation attempts to answer.

In Case 2, we see that the recommendation algorithm tends to recommend items similar to what the customer purchased in the past. Recommending similar items is the nature of the recommendation algorithm, and is desirable for certain applications such as movie recommendation. However, the same behavior could become undesirable

for product recommendation, specifically, for durable products like cameras. Recommending a matching lens for a camera, as described in Case 1, is much more reasonable. Hence, knowing how two products are related, e.g. whether one can replace or complement the other, is necessary for production recommendation. To go one step further, recommendation might be greatly benefited by considering the inherent connection among multiple products. The question then arises: “is there a principled approach to measure the recommendation quality for multiple products?”. We are strongly motivated to answer this question in this dissertation.

Situations described in Case 3 are fairly common in existing recommender systems. Existing systems usually focus on the benefits of one side; the benefits of the other side are ignored or even sacrificed. For example, movie recommendation only considers the benefits of the consumers, though it seems fair for this specific application. In a typical E-commerce environment, consumers and producers are both major players. For consumers, it is in their best interest to pay as little as possible, while producers are profit maximization driven and tend to price the product as high as they can. As producers and consumers have conflicting interests, it is a challenge for the system to balance their interests and in many cases the system simply favors one side. However, studies in economics suggest that it is better to consider the benefits of all parties in the system as whole; otherwise, the neglected side will be less motivated and even compelled to quit the system [2]. In order for the platform to sustain, it is necessary to consider the benefits of both the consumers and producers. In this dissertation, we aim to propose a new metric to assess the performance of the recommender systems. With the new

metric, the benefits of all stake-holders will be accounted for.

1.2 Economics in E-commerce recommendation: producer, consumer and platform

This dissertation proposes to answer the questions raised in the previous section by combining principles developed in the economics field and recommendation techniques developed in the computer science field. Economists have studied the behavior of producers and consumers for decades and have established solid principles and theories to explain the rationale of consumer and producer behavior. One of the important questions economists care about is: “what are the objectives for consumers and producers and how are they achieved?”. To answer this question, economists introduced the notion of *surplus* as a measurement of the benefit of a stake-holder in the market[164]. The surplus for consumers and producers are called *consumer surplus* and *producer surplus*, respectively. The sum of consumer surplus and producer surplus is called *total surplus*. In this dissertation, we adopt each of the surplus notions as the optimization objective and develop the product recommendation framework on top of it. This dissertation contains three major components:

1. **Producer centric recommendation:** This component studies the recommendation framework when the producer’s benefit is the primary concern. As described in Case 1, a recommendation may not be accepted by the consumer if the price is not right. In the market, producers are profit driven and spare no effort in making

the transaction take place. The key is to know the consumer's *willingness-to-pay* (WTP). We proposed to estimate WTP at the individual level and applied the estimated WTP for personalized promotion.

2. Consumer centric recommendation: This component focuses on a recommendation framework that maximizes the consumer's benefit. In economics, consumers are assumed to be rational and utility maximizers, which means consumers spend their money to gain the most utility from products[110]. As seen in Case 2, products are related in certain ways and consideration of such relationships is needed for multi-product utility calculation. This dissertation proposes to discover and represent the inter-product relationship. Further-more, it provides a recommendation framework based on multi-product utility maximization.

3. Platform centric recommendation: In contrast to the above components which only concern the benefit of one side, this component concerns the benefits of both the consumers and producers. The objective of the recommender system is making the platform sustainable. This is achieved by maximizing the total surplus resulting from matching producers to consumers. This dissertation proposes a total surplus maximization (TSM) based framework. The framework can be easily specified for several typical applications. Such applications include the E-commerce setting, P2P lending and the online freelancer marketing.

Compared to other related methods developed in recommender systems and economics, the methods presented in this dissertation have the following advantages:

First, we adopted economics-based metrics such as surplus and utility for model training and prediction. Compared to other commonly used metrics, our economics-based metrics can reveal the economic significance of the recommendation. For example, as we shall see in Chapter 3, recommendation by personalized promotion allows us to know the producer’s profit gained from the recommendation. Compared to metrics such as precision or conversion rate, profit is much more informative and intuitive for the marketing practitioner. Similarly, when consumer’s benefit is concerned in Chapter 4, the quality of a recommendation could be measured as utility or surplus in monetary units.

Second, our WTP elicitation mechanism is applicable to the E-commerce setting. WTP elicitation is an important task in economics and marketing research, and a number of approaches have been developed. However, it is quite a challenge to apply all existing methods in the E-commerce setting, as the existing methods were developed in settings largely different from E-commerce application. Our WTP elicitation method is both scalable and has high cost efficiency. It can complement existing methods and might also provide useful insights in building personalized promotion-based marketing tools.

Third, we are aware that some of the existing recommendation algorithms consider the relationship between the recommended items. For example, some algorithms are designed to diversify the recommendation result[6, 179, 189]. As existing methods tend to be generic, they might not perform the best when it comes to product recommendation where items are related in specific manners. In contrast, as our multi-

product relationship is founded on well established economic principles, it can better characterize and reflect the economic rationale behind consumers' behavior.

Fourth, compared to utility models developed in economics, our proposed multi-product utility model is less restricted. For example, the Constant Elasticity of Substitution (CES) utility model requires products sharing the same elasticity of substitution (ES). Conversely, our utility model is more flexible by allowing ES on a product pair basis. Besides, our utility model could further transform into a cardinal utility function. This could be useful for personalized promotion of multiple products as a bundle.

Fifth, we propose to use *total surplus* (TS) as the evaluation metric for recommender systems. Our proposed *total surplus maximization* (TSM) based recommendation framework emphasizes the well-being of the platform. The framework concerns both consumers and producers, as opposed to existing methods which usually optimize the benefit of one side. Our TSM based framework demonstrates its advantage on several typical applications.

With the above advantages, we hypothesize that the economic principle-based methods proposed in this dissertation are valuable for product recommendation.

1.3 The goal of the dissertation and overview of the solutions

The economic principle based approaches proposed in this dissertation are not intended to extend or replace existing recommendation algorithms. Instead, they aspire to provide a different perspective of the product recommendation problem by melding well established economic theory in consumer behavior and state-of-the-art recommendation techniques developed in the computer science field. Understanding the economic rationale behind consumer choice and the generic recommendation techniques are both essential to build a better product recommender system.

The goal of this dissertation is to treat the E-commerce product recommendation problem from the stance of different stakeholders, namely, consumer, producer and platform. For each treatment, we explore how to define the learning objective in terms of economics based metrics, formalize the problem by combining economic theory and personalization techniques and evaluate the approach under both conventional evaluation metrics and economic based metrics.

The approaches proposed in this dissertation are motivated by the following economic principles:

First, consumers are heterogeneous and could have different willingness-to-pay (WTP) for the same product. As consumers are deemed to be rational, they only purchase when their WTP is above the price. Motivated by this, we propose to learn and predict individual leveled WTP in the E-commerce setting. Knowing consumer

WTP enables personalized pricing (or promotion) so that the producer can achieve maximized profit.

Second, consumers are utility-maximization driven subject to their budget constraint - in other words, at any moment, they seek to purchase the set of products that give the most desirability given their limitation on their expenditure. Motivated by this, we propose to model multi-product utility by considering inter-product relationships.

Third, in economics, search and matching theory studies the formation of mutually beneficial relationships over time [122, 44, 46]. Matching theory has been applied to a number of problems including the formation of jobs – unemployed workers to job vacancies, marriage – unmatched people to other unmatched people and sales – buyers to sellers [79]. Motivated by this, we treat the recommendation problem as matching producers to consumers and propose aTSM-based framework for learning and evaluation.

1.4 Contributions of the dissertation

The main contributions of the dissertation include:

- This dissertation proposes to add personalized promotion to E-commerce recommender systems. Price is an important attribute for product recommendation, as opposed to other recommendation applications where price matters little. In economics, purchase happens only when the price is no greater than consumer’s willingness-to-pay (WTP). As WTP is individual specific, it appeals to the prin-

ciplined method that predicts WTP for each individual consumer for any given product. The task of obtaining individual leveled WTP is often referred to as *WTP elicitation* in economics literature. WTP elicitation is considered essential for dynamic pricing (or first degree price discrimination) in economics and a number of methods have been developed for this task. Though existing methods have been shown to be quite successful in some application scenarios, they face challenges when applied to the E-commerce setting. Aspiring to tackle these challenges, this dissertation presents an incentive-compatible mechanism that elicits consumer's WTP in the E-commerce setting. Based on the collected WTP, we are able to build machine learning models that predict WTP on unseen products. In economics, the main purpose of dynamic pricing is to increase profit for the producers. We propose to use profit as an evaluation metric in addition to RMSE as the traditional evaluation metric. The empirical results on the data collected from Amazon Mechanical Turk platform demonstrate the advantage of our personalized pricing compared to the baseline algorithms. In sum, the contribution of our personalized pricing work is twofold: first, we introduce a novel WTP elicitation procedure that is adapted to E-commerce systems; second, up to the writing of this dissertation, our work is the first attempt to add personalized promotion to E-commerce recommender systems, and we measure recommendation performance with metrics of direct interest to industry.

- This dissertation explores how to recommend by maximizing multi-product utility

for the consumer. Multi-product utility representation is regarded as important by economists, as it is an essential ingredient for the development of consumer choice theory. In order to represent multi-product utility, it is instructive to study the utility of two products. The challenge is to find a function that can flexibly express product relationships ranging from *perfect substitutes* to *perfect complements*. In the face of this challenge, we propose a general method to derive the utility function for any given *marginal rate of substitution* (MRS). Our method is mathematically sound and generic. More importantly, it provides important guidance for us to find the desired utility function. For example, the well known *constant elasticity of substitution* (CES) utility function is a natural result of our method when the exponential function is chosen for MRS. It is possible for us to obtain other interesting utility function alternatives by examining different choices of MRS function. Besides the in-depth study of the utility of two products, we also propose to represent multi-product utility in terms of pairwise product utilities. Compared to the multi-product utility function developed in economics, our representation is more flexible. We adopt multi-product utility for both model learning and evaluation. The empirical results on two real-world E-commerce datasets show that our approach outperforms baseline recommendation algorithms under both conventional evaluation metrics and utility.

- This dissertation proposes a Total Surplus Maximization (TSM) framework to integrate both consumer benefit and producer benefit into recommendation sys-

tems. With TSM, the system creates a bigger pie (total surplus) for consumers and producers to divide. There is a large gap between the traditional application of the economists' insight (a competitive market for a uniform commodity, with lots of small producers and consumers) and online allocation of very personalized and heterogeneous services. To fill the gap, we develop surplus-oriented metrics for personalized recommendations for heterogeneous products, and illustrate their use in several online markets. We will offer evidence that TSM can improve performance to the benefit of both sides. Indeed, our analysis and results for three real-world datasets (E-commerce, P2P loan, online freelancing) conclude that TSM-based recommendation performs better than standard recommendation techniques.

1.5 Outline

The dissertation is structured as follows. Chapter 2 is a literature review which includes recommendation methods developed in the computer science field, recommendation performance evaluation metrics and necessary economics background knowledge. Knowing the related work and the necessary background knowledge helps the reader to better understand the rest of the dissertation. Chapter 3 describes how we add personalized promotion to product recommendation. Chapter 4 describes how we recommend by learning and representing multi-product utility. Chapter 5 presents our total surplus maximization recommendation framework and its usage in several key application sce-

narios. Chapter 6 notes some conclusive remarks, discusses the limitations of the work and possible directions of future work.

Each chapter is relatively self-contained. Readers familiar with recommender systems can skip Section 2.1. Readers familiar with economics can skip Section 2.3. Readers interested in the general idea can read Chapter 6 only. Readers interested in the details of the proposed methods can read Chapter 3, Chapter 4 and Chapter 5 respectively.

Parts of the thesis have been published in conferences or in submission. Chapter 3 is based on a paper published in the International ACM Recommender System Conference in 2015[187], Chapter 4 is based on a paper submitted to the ACM SIGIR conference in 2016 and Chapter 5 is based on a paper published in the International WWW Conference in 2016[186].

Chapter 2

Literature review

For ease of reading, this chapter provides necessary background knowledge. Section 2.1 reviews existing recommendation algorithms and commonly used evaluation metrics. Section 2.3 briefly covers several key economic notions which will be used in our proposed methods.

2.1 Recommender systems

The earliest work related to recommender systems could be the research on document recommendation in the information retrieval community in the 1970s, and user modeling in cognitive science [133]. Recommender systems started to flourish in the mid 1990s and have also received attention from researchers in marketing [42], forecasting theory [10], and management science [118].

In early days, the Grundy system built stereotypes for recommendation [133]. In [64], the authors described the Tapestry system which asks users to identify similar

users manually, and recommends based on the similar users. Tapestry system first introduces the concept of collaborative filtering which later became one of the most influential recommendation algorithms.

The fast growth of the internet leads to a much larger volume of information about users, items and their interactions. Such change poses scale-up challenges to manual construction based algorithms (e.g. the Tapestry system) and appeals to the development of automatic recommendation algorithms. Over the past decades, a significant number of recommendation algorithms have been proposed. Existing algorithms can be roughly categorized into *collaborative filtering*, *content based filtering*, and *hybrid* algorithms, each of which will be described below.

2.1.1 Collaborative filtering (CF)

CF is motivated by the idea that people get the best recommendations from someone with similar tastes to themselves. A typical workflow of a CF system takes the following steps,

1. A user provides ratings on items (e.g. movies or E-commerce products) of the system. The ratings are usually considered to reflect the user's interest on the items.
2. The system finds neighboring users similar to current user.
3. The system then recommends current user items which have been highly rated by the neighboring users but not yet rated by current user.

A key problem of the above procedure is how to measure user similarity and combine the results from neighboring users. This problem inspires *memory-based* methods which can be implemented as user-based [131, 80, 69] or item based [146, 97, 92]. User based and item based methods have similar mechanisms, but item-based methods are more used as they perform better at scale and lower rating density. For example, in a web E-commerce setting where the number of items is much larger than the number of users and each user purchases several items, user-based methods are slow and make poor predictions. Below are the key ingredients of item-based methods. Let i, j denote two product indices and \vec{i}, \vec{j} denote their respective user ratings. The similarity of i, j can be calculated as,

$$sim(i, j) = \begin{cases} \frac{\vec{i} \cdot \vec{j}}{\|\vec{i}\|^2 \|\vec{j}\|^2} & \text{Cosine similarity} \\ \frac{\sum_{u \in U} (R_{ui} - \bar{R}_i)(R_{uj} - \bar{R}_j)}{\sqrt{(R_{ui} - \bar{R}_i)^2} \sqrt{(R_{uj} - \bar{R}_j)^2}} & \text{Pearson correlation} \end{cases} \quad (2.1)$$

In the recommendation stage, the system predicts the ratings of unseen items by using the above item-item similarity score. *Weighted sum* is a common method to make predictions. As its name implies, the weighted sum method computes the prediction on an item i by the weighted sum of the ratings given by the user on the items similar to i . Let Φ_i denote the set of items most similar to unseen item i , user's rating of i can be predicted as,

$$r'_{ui} = \frac{\sum_{j \in \Phi_i} sim(i, j) R_{uj}}{\sum_{j \in \Phi_i} sim(i, j)} \quad (2.2)$$

For computational efficiency, it is favorable to use just a few of the most similar items,

namely, $k = |\Phi|, k \ll N$ where N is the total number of items in the system. As k is an adjustable model parameter, item-based CF algorithm can be viewed as model-based method.

As we mentioned earlier, data scale and sparsity poses challenges to memory-based CF methods, and model-based methods have been proposed to address such challenges. As its name implies, model based methods introduce model parameters and learn such parameters by fitting them to the observed ratings. In [70], the authors propose an aspect model and two-sided clustering model. The aspect model expresses a user's preference as a convex combination of preference factors. Two-sided models simultaneously partition the user and the item into clusters such that each user or item is assigned to the respective cluster. In [152], the author propose to improve over [70] by allowing the user or item to be assigned to more than one cluster. Badrul et al. in [145] propose to represent user and item by latent vectors and solved the latent vectors by doing SVD on the rating matrix. However, as the rating matrix is always sparse, this creates a difficult non-convex problem, so a naive solution is not going to work[156]. Salakhutdinov et al. in [142] propose to use a *restricted Boltzmann machine* (RBM) - a two-layer undirected graphical model for user's ratings of movies. Compared to SVD, RBM models enjoy better learning and inference efficiency and perform as well as carefully tuned SVD models. Matrix factorization (MF) based methods [84, 87, 86, 126], like SVD, represent users and items by latent vectors, but they only consider the observed ratings, as opposed to SVD where missing entries are populated as zero values. Being advantageous in computational efficiency and recommendation performance, MF

is widely used and also adopted in this dissertation. It is necessary to briefly describe its mechanism. Let R_{ui} denote the rating on item i by user u , R_{ui} is modeled as,

$$R_{ui} \sim \mathcal{N}(\mathbf{x}_u^T \mathbf{y}_i + \alpha_u + \beta_i + \gamma, \sigma^2) \quad (2.3)$$

$$\mathbf{x}_u, \mathbf{y}_i \in \mathbb{R}^D, \alpha_u, \beta_i, \sigma, \gamma \in \mathbb{R} \quad (2.4)$$

where σ^2 is the variance of the Gaussian distribution and is usually given. The model parameters is,

$$\Phi = \{\mathbf{x}_u, \mathbf{y}_j, \alpha_u, \beta_i, \gamma, \sigma\} \quad (2.5)$$

The unknown model parameters can be obtained by solving the following optimization problem,

$$\min_{\Phi} \sum_{u=1}^M \sum_{i=1}^N I_{ui} (R_{ui} - (\mathbf{x}_u^T \mathbf{y}_i + \alpha_u + \beta_i + \gamma))^2 + \lambda \|\Phi\|^2 \quad (2.6)$$

where I_{ui} indicates whether user u has rated item i and λ is a regularization coefficient.

The above objective function can be efficiently optimized by *stochastic gradient descent* (SGD) or Alternating Least Squares (ALS) algorithms [83].

2.1.2 Content based filtering

Content based filtering methods are based on the assumption that the features (meta data, words in description, price, tags, etc.) used to describe the items that a user likes or dislikes tell much about the user's preferences. It usually recommends new items similar to previous items the user liked. The underlying research focuses on estimating a user's profile from explicit feedback on whether she liked previous items. Two approaches prevail. The first adaptively updates a user-specific retrieval model based

on algorithms originally designed for the retrieval task. Some examples are Rocchio, language models, Okapi, and pseudo-relevance feedback. The filtering system uses a retrieval algorithm to score each incoming document and delivers the document to the user if and only if the score is higher than a threshold. The second approach is to represent a user profile as a classifier and to deliver a document to the user if the classifier deems it relevant or if the probability of relevance is high. State of the art text classification algorithms, such as Support Vector Machine D-3 (SVM), K nearest neighbors (K-NN), neural networks, logistic regression, and Winnow, have been used to solve this binary classification task [94, 147, 13, 136, 177, 181]. Both of the above approaches focus on identifying relevant items, which are often represented in an attribute space indexed by text features such as keywords.

2.1.3 Hybrid recommendation algorithms

Hybrid recommendation algorithms combine collaborative filtering with content based filtering, and usually perform better than either filtering method alone [18]. Social network based recommendation algorithms explore a user's social networks to make recommendations[76, 63, 82, 65, 161, 167, 103]. This research topic has drawn much attention recently, largely due to the popularity of web2.0. There are two major threads of research that connect social networks with recommendations. The first approach looks at how a user's social network can help predict the user's preferences [103, 38]. The second thread traces information propagation in social networks and its influence on user behavior [138, 78, 154, 153, 89, 50] For example, [76] suggests combin-

ing social networks and collaborative filtering techniques for recommendation. Massa & Bhattacharjee showed the potential power of trust networks, and argue for a way of propagating trust over all users [112]. Konstas, Stathopoulos & Jose adopt the generic framework of Random Walk on social networks and find it further improves recommendation performance [82]. In general, researchers have shown that trust networks are very useful for recommendation [63, 82, 65, 161, 167, 103]. As predicted by the theory and research, the effectiveness of social networks for recommendation could be dramatic in the real world. For example, within the first 48 hours after launch, Apple’s network for music reached more than 1 million users, largely through the social music discovery’ service that initially included a Facebook Connect option for finding friends.

2.2 Evaluation metrics

Evaluation is needed for every algorithm and different evaluation metrics might be used, depending on the application or the requirement. In the following, we first describe the experimental setup in a general sense and then describe a few most used evaluation metrics.

2.2.1 Setup

In a typical evaluation setting, the dataset is randomly split into training and testing subsets; that is, for each user in the dataset, part of the user’s ratings are used for model training and the rest for testing. In the training stage, cross-validation is used to select the best model. The model obtained in training stage then makes predictions

on the testing dataset and the prediction performance is evaluated according to the following metrics.

2.2.2 Root mean squared error (RMSE)

RMSE is a popular evaluation metric for rating based recommendation, e.g. movie recommendation. Let R_{ui} be the ratings of the testing dataset and R'_{ui} be their respective predicted ratings, RMSE is defined as,

$$RMSE = \sqrt{\frac{\sum_u \sum_i I_{ui} (R_{ui} - R'_{ui})^2}{\sum_u \sum_i I_{ui}}} \quad (2.7)$$

where $I_{ui} = 1$ if the rating R_{ui} exists in the testing dataset, 0 otherwise. Another metric related to RMSE is *mean absolute error* (MAE) which defines the error of a single prediction as the absolute value of the difference between the true rating value and its predicted value,

$$MAE = \frac{\sum_u \sum_i I_{ui} |R_{ui} - R'_{ui}|}{\sum_u \sum_i I_{ui}} \quad (2.8)$$

2.2.3 Ranking based metrics

RMSE is suited for rating prediction based tasks and does not apply to implicit feedback situations where users do not express feedback explicitly. Implicit feedback is commonly encountered in applications, e.g. user clicks a web link, user stays through or skips a song, or in E-commerce, user purchases the product. It is common practice to adopt ranking related metrics for implicit feedback situation. One of the most commonly used metrics is *precision at top-K*. Let T_u be the set of items for user u in the testing

set and T'_u be the recommendation list of length K . The precision and recall for top K recommendations are,

$$precision_u@K = \frac{|T_u \cap T'_u|}{K} \quad (2.9)$$

$$recall_u@K = \frac{|T_u \cap T'_u|}{|T_u|} \quad (2.10)$$

$$FScore_u@K = \frac{2precision@K \times recall@K}{precision@K + recall@K} \quad (2.11)$$

Averaging over all testing users gives the averaged precision/recall statistics, that is,

$$precision@K = \frac{\sum_{u \in U} \frac{|T_u \cap T'_u|}{K}}{|U|} \quad (2.12)$$

$$recall@K = \frac{\sum_{u \in U} \frac{|T_u \cap T'_u|}{|T_u|}}{|U|} \quad (2.13)$$

$$FScore@K = \frac{\sum_{u \in U} \frac{2precision@K \times recall@K}{precision@K + recall@K}}{|U|} \quad (2.14)$$

2.3 Economics Basics

2.3.1 Utility

In Victorian days, *utility* was viewed as a numeric measure of a person's happiness. Based on this idea, consumers were thought to make choices to maximize their utility, that is, to make themselves as happy as possible[165]. This classical view of utility has some conceptual problems, as it did not measure utility at a quantitative level. For example, it was not able to tell the exact difference between the utilities associated with different goods; also, it couldn't compare one person's utility to another. Due to such conceptual problems, economists have abandoned the classical view of utility. Instead, they have reformulated the consumer behavior in terms of *consumer preferences*,

Table 2.1: Different ways to assign utilities. Each column corresponds to a utility assignment and all assignments describe the bundle preference relationship: $A \succ B \succ C$

Bundle	U_1	U_2	U_3
A	3	5	1
B	2	1	0
C	1	0	-1

and utility is seen as a way to describe consumer preferences over some set of goods or services. Let's say that there are two product bundles of two products and one bundle (x_1, x_2) is preferred to the other (y_1, y_2) . All we can say is x-bundle has higher utility than y-bundle, but how much higher doesn't matter that much. The *preferences* of consumers set the foundation for consumer choice theory and utility is a simple way of describing preferences.

2.3.2 Utility function

A **utility function** is a way of assigning numerical values to all consumption bundles such that the more preferred bundles get larger numbers than less preferred bundles. Mathematically, that is to say,

$$(x_1, x_2) \succ (y_1, y_2) \text{ if only if } U(x_1, x_2) > U(y_1, y_2) \quad (2.15)$$

It can be seen that there is more than one solution for $U(x_1, x_2)$ and $U(y_1, y_2)$ as long as the above preference order is preserved. To further illustrate this, consider three bundles

with the preference order: $A \succ B \succ C$. Table 2.1 shows three ways of assigning the bundle utilities. Utility functions like Equation 2.15 that emphasizes ordering of product bundles is referred to as *ordinal utility*. There is no unique solution for ordinal utility function, as *a monotonic transformation of a utility function is still a utility function that represents the same preferences as the original function*. In this dissertation, we represent monotonic transformation by a function $z(u)$ that transforms input utility u to utility $z(u)$. Examples of monotonic transformation include scaling by a positive number (e.g. $z(u) = 5u$), offsetting by any number (e.g. $z(u) = u - 3$), raising to an odd power (e.g. $z(u) = u^3$), and so on.

2.3.3 Cardinal utility

As opposed to ordinal utility for which the magnitude of utility does not have significance, *cardinal utility* theories attach significance to magnitude of utility. Ordinal utility is intended to represent a given person preferring one bundle of goods to another, but it does not tell how much more the person likes one bundle than the other. With cardinal utility, the difference between two bundles of goods is supposed to have significance. For example, suppose an apple has utility of 10 and an orange has utility of 5; with cardinal utility, we can say that the apple is twice as good as the orange. A basic question for cardinal utility would be, “how do we know one likes one bundle twice as much as another?”. There might be various ways to construct such utility assignments. For example, a user likes one bundle twice as much as another if she is willing to pay twice as much for it, or is willing to wait twice as long to get it. Though it’s possible

to obtain cardinal utilities in a compelling way, economists do not regard such utilities as bringing extra benefit for describing consumer choice. The reason is consumers make choices based on preferences over the consumption bundles and ordinal utilities are sufficient to describe the preferences. Nevertheless, we argue that cardinal utilities would be useful when it comes to evaluation of surplus, as we shall see in Chapter 3 and Chapter 5.

2.3.4 Utility function

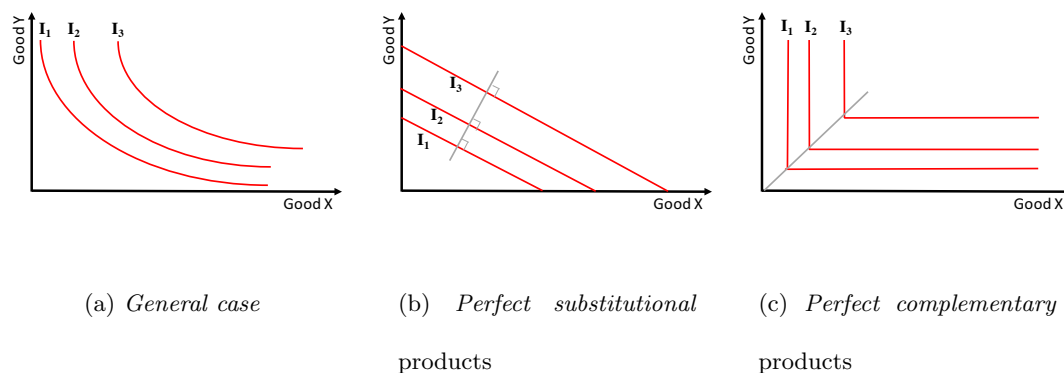


Figure 2.1: Illustrative indifference curves for common pairs, perfect substitutional pairs, and perfect complementary pairs. The utilities of the three illustrative curves satisfy $I_1 < I_2 < I_3$.

Utility functions can be constructed based on the *indifference curve* as shown in Figure 2.1. Indifference curves are a way to represent preferences over consumption bundles. As we shall see in Section 4.2.1, it is possible to derive the utility function based on the property of the given indifference curves. Without loss of generality, we consider the utility of two products $U(x_1, x_2)$.

Perfect substitutes Consider the utility of smaller and larger bottles of water. Let's say the larger bottle is twice the size of the smaller one. The utility would be, $U(x_1, x_2) = x_1 + 2x_2$. The utility function is linear with x_1 and x_2 . Given a fixed utility, one product can be substituted by the other product at a constant rate. As illustrated in Figure 2.1(b), the indifference curves are straight lines. The generic form of perfect substitutes utility is $U(x_1, x_2) = ax_1 + bx_2$ where $a > 0$ specifies the substitution rate. As we have seen earlier, any monotonic transformation of $ax_1 + bx_2$ e.g. $U(x_1, x_2) = a^2x_1^2 + 2abx_1x_2 + b^2x_2^2$, is also a utility function.

Perfect complements

Consider the utility of left shoes and right shoes. As the utility is determined by the number of pairs, extra left shoes or right shoes do not add utility. In cases like this, where two products complement each other, the utility can be represented as $U(x_1, x_2) = \min\{ax_1, bx_2\}$. An example indifference curve for perfect complements is shown in Figure 2.1(c). Please note that this function is not differentiable and is hard to work with directly. As we shall see in Chapter 4, we use a differential function to approximate perfect complements.

Cobb-Douglas utility

Cobb-Douglas utility is one of the most used utility functions. Its function form is $U(x_1, x_2) = x_1^a x_2^{1-a}$ where $0 < a < 1$. Compared to perfect substitutes and perfect complements which represent two extreme cases, Cobb-Douglas represents general cases as illustrated in Figure 2.1(a).

CES utility

Table 2.2: Commonly used cardinal utility function

Name	Function	Parameters
Exponential utility	$U(q) = \frac{1 - \exp(-aq)}{a}$	$a > 0$
KPR ¹ utility	$U(q) = a \ln(1 + q)$	$a > 0$
Cobb-Douglas utility	$U(q) = aq^\alpha$	$a > 0, 0 < \alpha < 1$

CES, or Constant Elasticity of Substitution, takes the function form as $(ax_1^r + (1 - a)x_2^r)^{\frac{1}{r}}$ $0 < a < 1$. CES is rather generic. The above utility function can be viewed as special cases of CES. If $r = 1$, we get linear utility or perfect substitutes. if r approaches 0, we get Cobb-Douglas utility; if r approaches negative infinity, we get Leontief or perfect complements. Due to its flexibility, we adopted CES for multi-product utility modeling in Chapter 4.

So far we have been mainly concerned with ordinal utility functions. It is also important to consider cardinal utility functions when dealing with surplus. Among the many functional forms, three of them will be particularly useful for us. They are listed in Table 2.2. For a single item, utility $U(q)$ is a function of the consumed quantity q . If that item is indeed good, then more is better and the marginal utility is positive, i.e., $U'(q) > 0$. The *Law Of Diminishing Marginal Utility* [134] states that there is a decline in the marginal utility a person derives from consuming each additional unit of the product, i.e., $U''(q) < 0$. For example, a hungry person may obtain a huge amount of

satisfaction when consuming the first slice of bread, but the increase in satisfaction per slice declines as she consumes more and more slices. The parameter $a > 0$ in Exponential Utility can be interpreted as the person's absolute risk aversion, but here it will just represent how fast marginal utility decreases as consumption increases. The parameter $a > 0$ in KPR utility has a similar interpretation. Please note that KPR has the same curve shape for all products; in other words, the marginal utility for all products diminishes at the same rate. This is a rather restrictive assumption as products do vary in their diminishing rates of utility. In contrast, Cobb-Douglas controls the function shape through exponent α . Thus, we can model α as function depending on product. All these utility functions have zero utility for zero consumption, i.e. $U(0) = 0$. We will regard all these functions as money metrics in the sense of Varian [164].

With those normalizations and interpretations, utility can be understood in monetary terms. $U(q)$ is the dollar value to the consumer of being able to consume q units of the good in question, and $U'(q)$ is her Willingness To Pay (WTP) [56, 187], i.e., the maximum amount of money she would pay to acquire another unit of good. This enables us to align the utility in the same scale as price so as to calculate the surplus of consumers and producers [54].

2.3.5 Surplus

Consumer Surplus (CS) is the amount of utility that consumers experience beyond the amount that they pay (i.e., the price per unit times the number of units). Similarly, Producer Surplus (PS) is the amount of revenue that the producer earns

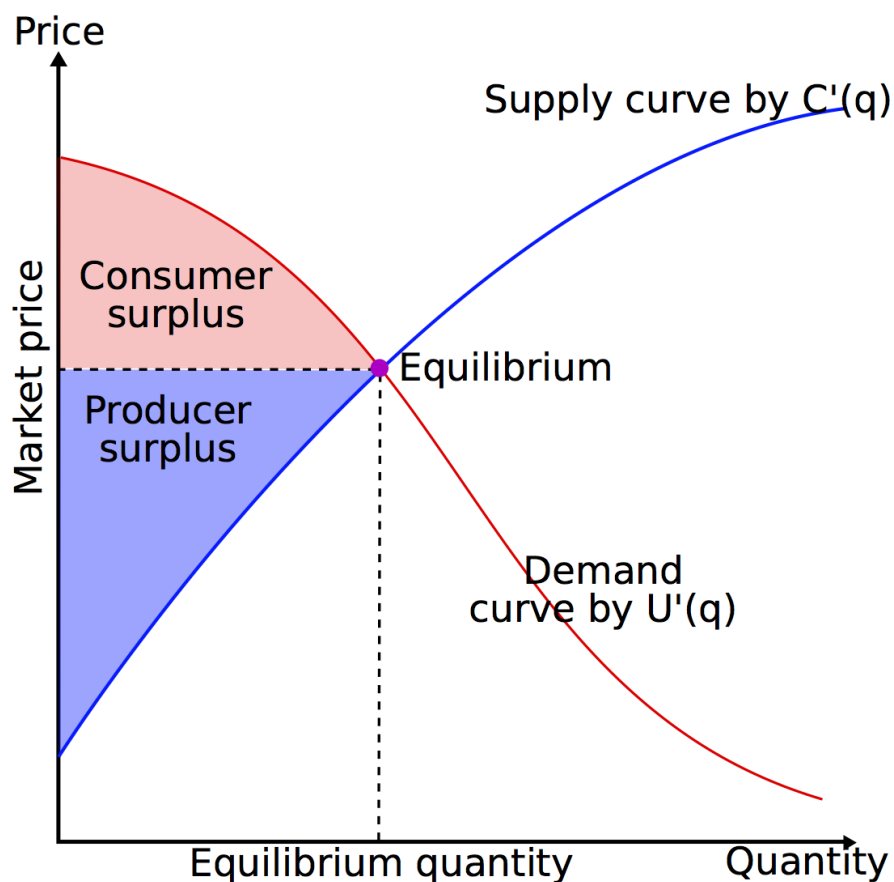


Figure 2.2: An intuitive explanation of surplus derived from marginal utility and marginal cost.

beyond the (variable) cost of producing those units.

The consumer and producer surpluses are indicated in Figure 2.2, where the demand curve is the marginal utility function $U'(q)$, which decreases according to the exposition in the previous section. The supply curve is the marginal cost $C'(q)$, which increases according to the *Law of Diminishing Marginal Returns* [134].

In a competitive market, the price P is determined by the equilibrium (intersection) of the two curves. Typically the preferences of many small buyers and the costs of many small sellers lie behind the demand and supply curves of the whole market,

but for present purposes it is convenient to think of a single representative buyer and a single representative seller. Given the quantity of consumption $q = q_c$, Consumer Surplus CS is obtained by integrating the marginal utility that exceeds the price at each unit of consumption until q_c :

$$CS = \int_0^{q_c} (U'(q) - P) dq = U(q_c) - Pq_c \quad (2.16)$$

and Producer Surplus PS is similarly determined by:

$$PS = \int_0^{q_c} (P - C'(q)) dq = Pq_c - C(q_c). \quad (2.17)$$

Total Surplus TS is defined as the sum of the surplus gained by the consumer and producer, which is:

$$TS = CS + PS = U(q_c) - C(q_c) \quad (2.18)$$

Moreover, the total surplus of an economic system is the sum of the surpluses for all parties involved in all the transactions of the system.

We see in Eq.(2.18) that the price component offsets and does not affect the total surplus in a single transaction. Of course, the same is true for any set of transactions, even if they occur at different prices. This is the source of an important insight in economics: the total surplus in a given set of transactions may be split more or less advantageously for buyer or for seller, but the only way to increase the total (potentially making both sides better off) is to change the set of transactions. Society is best served when social surplus is maximized, because this provides the largest possible “pie” to split among all participants.

Chapter 3

For producer: recommendation by personalized promotion

As we have seen earlier, price is a key factor in determining a consumer's purchase decision and producer surplus. An important limitation of existing studies is that they assume that the properties of items (i.e. products) are static. However, an E-commerce company could tailor some properties of a product for a particular customer, and that could dramatically improve the effectiveness of a recommendation. This can be better understood by referring to the surplus generated by the market activities, as illustrated in Figure 2.2. As the market price of a product is the equilibrium price where the demand curve and supply curve intersects, there is consumer surplus for consumers whose WTP is above the market price. The consumer surplus can be captured as producer surplus if the price is set to the individual's WTP. Similarly, for consumers whose WTP is below the market price, their consumer surplus can also be captured as

producer surplus if the price is set to their WTP. In this chapter, we argue that price is a controllable property that the recommender system should incorporate. Outside of the literature on recommender systems, the crucial role of pricing is widely recognized. Researchers in marketing, for example, have shown the importance of personalized promotion for increasing sales volume [66].

In this chapter, we introduce personalized promotion into E-commerce recommender systems. The objective is to improve the effectiveness of product recommendations through customizing product price on an individual basis. To achieve this goal, we create a novel auction/lottery mechanism to elicit consumer willingness to pay (WTP) for relevant products in an E-commerce setting. Using an online shopping website with products from Amazon, we recruit experimental subjects via Mechanical Turk and use responses elicited for some chosen products to predict an individual's WTP on other products. The predicted WTP and the given production cost enable us to find profitable personalized prices. The data indicates that the new approach achieves nearly a 200% improvement in gross profit when compared with Amazon's default pricing strategy.

Our contribution is threefold. First, to the best of our knowledge, our work is the first attempt to add personalized promotion to E-commerce recommender systems, and we measure recommendation performance with metrics of direct interest to industry. Second, we introduce a novel WTP elicitation procedure that is adapted to E-commerce systems. Most previous WTP elicitation studies are conducted in settings that are very different from an E-commerce website — typically only a handful of products in a narrow category are pre-selected by the experimenters — and so the applicability to

E-commerce is unclear. Third, we have several methodological observations that could be very useful for further research on personalized pricing in E-commerce.

3.1 Methods

We envision a recommender system that not only predicts whether a consumer likes a product, but also predicts a consumer’s *willingness-to-pay* (WTP) for it. WTP is the highest price the consumer is willing to pay for a product [164]. If the WTP is lower than the default product price, the system might give a personalized promotion to the consumer to increase the possibility of accepting the recommendation.

We propose that the E-commerce system run a lottery (related to the BDM¹ mechanism just described in Section 3.5) for customers to collect WTP for a small number of recommended products, and then use machine learning on each customer’s lottery data to build a WTP prediction model. The WTP predictions then allow the recommender system to set personalized promotion prices automatically, potentially enhancing customer satisfaction as well as seller profit.

The rest of this section details this approach.

3.1.1 Bidding & Lottery Procedures

Our procedure elicits WTP for N products. To describe it clearly, we begin with the case of a single product ($N = 1$). The participant’s true WTP, denoted y , is unknown to us. We endow her with a large amount of cash C that likely exceeds y ,

¹BDM for Becker-DeGroot-Marschak.

and tell her that the mechanism will draw a random price r uniformly from the range $[R_L, R_U]$, which is chosen to include all plausible values of her WTP. She is asked to state her actual WTP by entering a bid $y' \in [R_L, R_U]$, and told that the mechanism will sell her the product at price r only if that price is less than her bid.

Using the symbol P to denote probability, her payoff is

$$\mathcal{M} = \begin{cases} C & \text{if } r > y', P = \frac{R_U - y'}{R_U - R_L}, \\ C + y - r & \text{if } r \leq y', P = \frac{y' - R_L}{R_U - R_L}. \end{cases} \quad (3.1)$$

That is, she keeps the cash if the random price r exceeds her stated WTP y' , and otherwise she purchases (and gains true benefit y) at the random price, which is lower (and hence a better deal) than her her stated WTP y' .

To see that this BDM procedure is incentive compatible, first note that the expected payoff is:

$$\begin{aligned} E[\mathcal{M}|y', y] &= P(r > y')C + P(r \leq y')E[C + y - r|r \leq y'] \\ &= C + P(r \leq y')(y - E[r|r \leq y']). \end{aligned} \quad (3.2)$$

Use the convenient property of the uniform distribution that the last conditional expectation is $E[r|r \leq y'] = \frac{R_L + y'}{2}$ to simplify the equation to:

$$E[\mathcal{M}|y', y] = C + \frac{2yy' - y'^2 - 2R_Ly + R_L^2}{2(R_U - R_L)} \quad (3.3)$$

Equation 3.3 shows that the participant's expected payoff is a quadratic function of her decision variable y' whose unique global optimum is reached when $y' = y$. In other words, it is in the subject's best interest to state her WTP truthfully.

Note for later reference that when the participant indeed sets $y' = y$, her expected payoff can be written as

$$E[\mathcal{M}|y, y] = C + \frac{(y - R_L)^2}{2(R_U - R_L)} > C \quad (3.4)$$

In our experiment we set $R_L = 0$ and $R_U = p$, where p is the known outside (undiscounted) market price for the product. In this case,

$$E[\mathcal{M}|y, y] = C + \frac{1}{2}k^2p \quad (3.5)$$

where $y = k * p$, and we refer to k as the normalized bidding price.

What if the subject wants to bid above R_U or below R_L ? In principle, there is no problem; in the first case she is sure to obtain the object and reveals that her true WTP exceeds R_U , and in the second case she is sure to keep the cash and reveals that her true WTP is below R_L .

A potential problem does arise when the number of products $N > 1$. Running a separate BDM elicitation for several products that are substitutes for each other will motivate participants to underbid, since the probability of winning at least one of several bids is higher than winning a single bid. We eliminate this problem by having each participant enter a bid for each product, but only drawing a random bid for one of the products chosen randomly. That is, we conduct a lottery over which of the products will actually be available to the participant. The expected payoff (or $1/N$ of the expected payoff) is still maximized by truthfully reporting WTP for each product.

An extension of this lottery approach enables us to economize payments in the experiment. We can also conduct a lottery over which participants actually get the

cash C and perhaps one of the products. Again, that attenuates the expected payoff but doesn't change where it is maximized.

3.1.2 Personalized Promotion

Our lottery/auction procedure gives us training data for predicting a consumer's WTP on a range of products that might be recommended. Given the values of y elicited from consumer u for product i , we run the regression

$$y_{ui} = b_0 + b_u + f(\mathbf{x}_{ui}, \mathbf{w}), \quad (3.6)$$

where b_0 is the global bias and b_u is the user bias, \mathbf{x}_{ui} is a feature vector representing information about consumer u , product i and their relationships. The functional form of the regression function f depends on the machine learning algorithm used, such as linear regression or gradient boosted trees. We used *linear regression* (LR) in our experiments. Thus our model parameters are:

$$(b_0, b_u, \mathbf{w}^*) = \underset{b_0, b_u, \mathbf{w}}{\operatorname{argmin}} \sum_{u,i} L(y_{ui} - b_u - b_0 - f(\mathbf{x}_{ui}, \mathbf{w})) \quad (3.7)$$

where in our implementation L is a quadratic loss function.

Given the model parameters learned from the training dataset, we use Equation 3.6 to predict WTP for any user u and item i pair. To capture the uncertainty in our estimation, we assume that the true WTP value follows a Gaussian distribution centered on the predicted WTP:

$$Y_{ui} \sim \mathcal{N}(\mu_{ui}, \sigma^2) \quad (3.8)$$

where $\mu_{ui} = b_u + b_0 + f(\mathbf{x}_{ui}, \mathbf{w})$. The parameter σ^2 could be estimated, but for simplicity we assume that is the same for all user item pairs and approximate it using the variance of y on the training dataset in our experiments.

Based on the distribution of Y_{ui} , we can find the optimal price t^* that maximizes the expected seller profit as follows:

$$t^* = \operatorname{argmax}_t (t - c)P(Y_{ui} \geq t) \quad (3.9)$$

where c is the production cost of product i and

$$\begin{aligned} P(Y_{ui} \geq t) &= \int_t^\infty \frac{1}{\sigma\sqrt{2\pi}} \exp^{-\frac{(x-\mu_{ui})^2}{2\sigma^2}} dx \\ &= \frac{1}{2} - \frac{1}{2} \operatorname{erf}\left(\frac{t - \mu_{ui}}{\sqrt{2}\sigma}\right) \end{aligned} \quad (3.10)$$

where $\operatorname{erf}(x)$ is Gauss error function encountered in integrating the normal distribution. Equation 3.9 is solved numerically using standard optimization techniques.

3.2 Experimental procedures

Can the proposed lottery-auction mechanism and the prediction model actually work with actual E-commerce customers? Are those methods of practical value to sellers? To begin to answer these questions we ran an experiment as follows.

3.2.1 Data collection

We first developed a website that links to Amazon.com, a leading online shopping website. Our website implements a full-fledged product search engine and displays

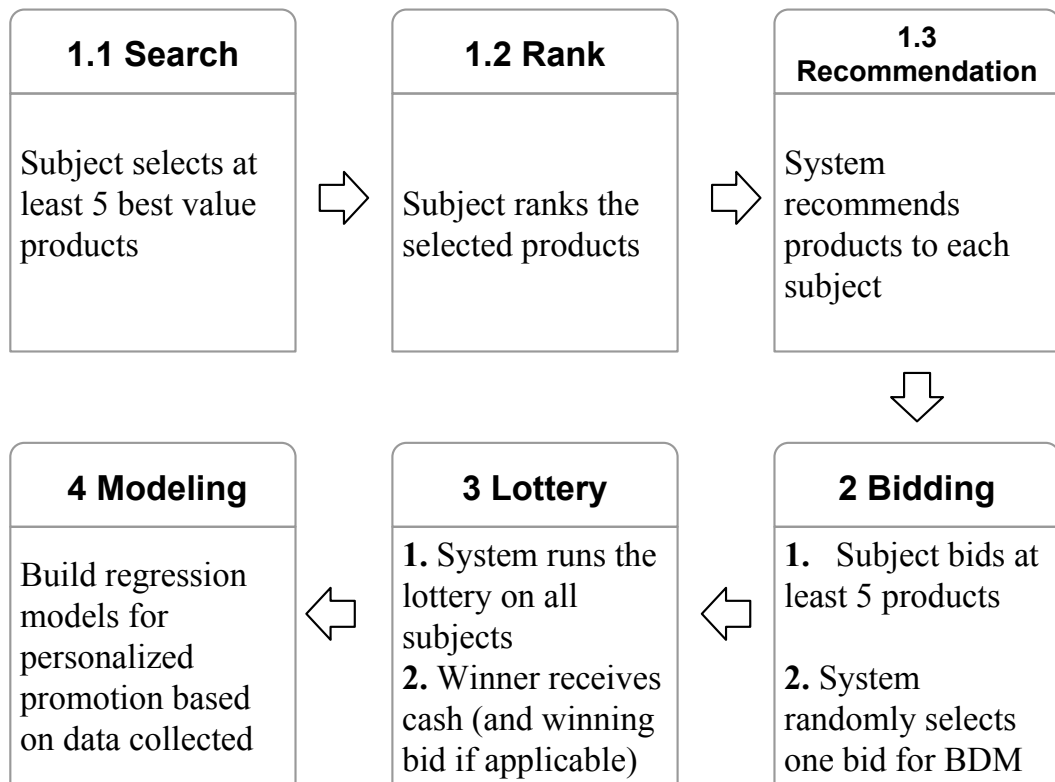


Figure 3.1: Overview of the experiment flow. The purpose of step 1 is to collect information about current user. The information will be used for later modeling.

the product information in a similar way to Amazon. The website hosts about 120,000 skin care products. We choose skin care products as they have short repurchase cycle, reducing problems that arise from unknown current holdings of the participants. The products are from Amazon.com and the product information is synchronized with Amazon.com at real time. For each product, Amazon provided both its list price and sales price.

Subjects recruited from Amazon Mechanical Turk were paid a participation fee of \$0.50 for around 10 minutes to complete the experiment. Figure 3.1 is an overview of the procedure. First, the subject sees a set of recommended products produced by a standard recommendation algorithm. This step would be straightforward for a typical E-commerce system, which can find relevant products based on the subject's past purchasing or click data. Since our custom website does not have the recruited subject's purchasing history, we let each subject search and identify at least 5 products which he or she thinks are worth buying; these are referred to as *best value* products. The subject proceeds to next rank their best value products in decreasing order of interest. Next, the system recommends a list of products using Amazon's "consumers who bought this also bought these" recommendations.

Having completed the steps labelled 1.1-1.3 in Figure 3.1, the recruited subject then participates in N BDM lotteries. That is, she enters a bid on each of $N \geq 5$ products that she chooses from those recommended in the previous step, knowing that at most one of the products will be randomly selected to go through the BDM procedure. Finally, one of the subjects recruited that day is randomly selected as the lottery winner,

and one of her chosen N products is randomly selected. If her bid on that product exceeds the random buyout price r , then she receives the product and \$100 minus r . Otherwise she gets \$100 cash.

3.2.2 Subjects' training and selection

We showed earlier that truth telling (bidding one's actual WTP) is the unique optimal strategy in BDM. However, earlier empirical research has shown that people may have misconceptions about the game and may not use the optimal bidding strategy [30]. Guided by the earlier research, we provided detailed explanations about the game rule and encouraged our subjects to practice with a bidding simulator to see how the outcome changes with different bids. Then each subject was required to take a quiz to check whether the participant understands the optimal bidding strategy. Only the 79 subjects who passed the quiz (out of 130 who responded initially) were allowed to participate in the game.

3.2.3 Data collection

To clean the data, records with normalized bidding prices either below 10% or more than two times higher than the Amazon sale price are filtered out, as are the bids of subjects who took less than 5 minutes to complete the experiment. After cleaning, we ended up with 339 product bids from 54 subjects. The bid price is normalized using the product's Amazon sale price, as k in Equation 3.5.

To construct \mathbf{x}_{ui} , we used the 9 features described in Table 3.1. The features

are based on information collected from each experiment step.

3.2.4 Evaluation Metrics

RMSE (Root Mean Square Error) is the most commonly used metric and is naturally adopted in our experiments. However, to understand the commercial value of personalized promotion based on the estimation of consumer’s WTP, we go beyond RMSE and introduce *seller profit* to measure the effectiveness of WTP prediction. Seller profit of a single product purchase is simply the purchase price minus the cost of producing the product. For the proposed WTP prediction method in Section 3.1.2, the profit is expected profit given pricing via Equation 3.9. Hence

$$\text{Profit}_{ui} = \begin{cases} t_{ui}^* - c_i & \text{if } y_{ui} \geq t_{ui}^* \\ 0 & \text{otherwise} \end{cases} \quad (3.11)$$

where t_{ui}^* is the optimal personalized promotion price from Equation 3.9, y_{ui} is the consumer’s true WTP and c_i is the unit production cost of product i .

3.3 Experimental Results

We compared the proposed approach with the Amazon sale price with 0, 10% and 20% discounts and ZeroR. The Amazon sale price methods serve as baselines. ZeroR algorithm is a simple learning algorithm which only relies on the target variable and ignores the predict variables for prediction. For classification problem, ZeroR uses the most frequent class label of the training samples as prediction for testing samples; for

Name	Description	Weight
list price	Amazon list price of the product	-0.0013
number of reviews	number of reviews on Amazon	-0.001
brand variety	the variety of the brands of consumer's best value products, measured by entropy. The larger value of this feature, the more variety.	0.018
discount	Amazon promotional discount	0.399
Amazon recommendation rank	rank in Amazon product recommendation list	0.02
average rating	Amazon average rating	0.044
user rank	rank in consumer's best value products	-0.0017
switch brand	whether product(s) of the same brand as the bidding product have been identified by consumer as best value products	-0.059
brand popularity	the number of times getting bid	0.054
<i>intercept</i>	<i>intercept term of the LR model</i>	-0.058

Table 3.1: The features used to predict WTP modeling, and their weight coefficients learned by Equation 3.7. Please note that the *brand variety* feature and *switch brand* feature might interact, as consumers who are loyal with one brand are less likely to switch to another brand at the bidding stage, and vice versa. In this paper, the small number of data samples limits us to linear regression models. In the future, it would be interesting to explore other models which can better handle feature interactions, e.g. decision trees.

Algorithm	RMSE	Profit (\$) w.r.t Production Cost in Terms of Percentage of Original Sale Price									
		None	10%	20%	30%	40%	50%	60%	70%	80%	90%
Original Sale Price	0.4549	\$1,089.69	\$980.72	\$871.75	\$762.78	\$653.81	\$544.84	\$435.88	\$326.91	\$217.94	\$108.97
90% Original Sale Price	0.4012	\$1,583.74	\$1,407.77	\$1,231.80	\$1,055.83	\$879.86	\$703.89	\$527.91	\$351.94	\$175.97	\$0.00
80% Original Sale Price	0.3674	\$2,093.15	\$1,831.50	\$1,569.86	\$1,308.22	\$1,046.57	\$784.93	\$523.29	\$261.64	\$0.00	-\$261.64
70% Original Sale Price	0.3591	\$2,382.65	\$2,042.28	\$1,701.90	\$1,361.52	\$1,021.14	\$680.76	\$340.38	\$0.00	-\$340.38	-\$680.76
60% Original Sale Price	0.3782	\$2,387.02	\$1,989.19	\$1,591.35	\$1,193.51	\$795.67	\$397.84	\$0.00	-\$397.84	-\$795.67	-\$1,193.51
50% Original Sale Price	0.4207	\$2,311.58	\$1,849.27	\$1,386.95	\$924.63	\$462.32	\$0.00	-\$462.32	-\$924.63	-\$1,386.95	-\$1,849.27
40% Original Sale Price	0.4807	\$2,059.32	\$1,544.49	\$1,029.66	\$514.83	\$0.00	-\$514.83	-\$1,029.66	-\$1,544.49	-\$2,059.32	-\$2,574.15
30% Original Sale Price	0.5523	\$1,727.06	\$1,151.38	\$575.69	\$0.00	-\$575.69	-\$1,151.38	-\$1,727.06	-\$2,302.75	-\$2,878.44	-\$3,454.13
20% Original Sale Price	0.6317	\$1,230.25	\$615.13	\$0.00	-\$615.13	-\$1,230.25	-\$1,845.38	-\$2,460.51	-\$3,075.63	-\$3,690.76	-\$4,305.88
10% Original Sale Price	0.7163	\$675.54	\$0.00	-\$675.54	-\$1,351.08	-\$2,026.61	-\$2,702.15	-\$3,377.69	-\$4,053.23	-\$4,728.77	-\$5,404.31
ZeroR	0.3586	\$2,350.58	\$2,023.66	\$1,696.74	\$1,369.82	\$1,042.90	\$715.98	\$389.05	\$62.13	-\$264.79	-\$591.71
Linear Regression	0.2200	\$2,734.14	\$2,260.44	\$1,831.22	\$1,442.21	\$1,104.83	\$818.60	\$609.63	\$439.67	\$297.73	\$176.45

Table 3.2: Performance of baseline algorithms and our linear regression algorithm in 10-fold cross validation setting. The baseline algorithms are flat discount of original sale price at different rates. Profit is evaluated w.r.t different production cost assumptions. The production costs are expressed as percentage of original sale price. For each production cost assumption, the best result of the baseline algorithms is in bold.

regression problem such as our WTP prediction, ZeroR uses the mean value of the target variables of the training samples as prediction for testing samples. For example, if there are three training samples and their respective target variables (normalized bidding price) are 0.7,0.5,0.6, ZeroR predicts the normalized bidding price of the testing samples as $\frac{0.7+0.5+0.6}{3} = 0.6$. In our experiment, the normalized bidding price for product i by ZeroR is 0.72 (i.e. the learned discount rate is 0.28). Our proposed algorithm LR sets the price at the point where the expected profit is maximized (Equation 3.9). All algorithms are evaluated by 10 fold cross validation.

In our experiments, we do not know the production cost of each product. Because retail businesses commonly set prices using a fixed markup on cost, we assume that production cost is a certain percentage of the original sale price. We varied the percentage from 0 to 90% with an incremental step of 10%. Table 3.2 reports the corresponding profit for each algorithm at different production cost levels.

Table 3.2 shows that our proposed method (i.e. LR) outperforms Amazon’s pricing strategy and ZeroR. One-sided t-testing result on the profits by the baseline algorithm and linear regression algorithm shows that the improvements for all product cost levels are statistically significant. The LR algorithm maximizes expected profit as in Equation 3.9, using $\sigma = 0.22$ based on the training dataset. It is important to model the *uncertainty* of WTP and set the price by maximizing the expected profit. We compared the proposed method with simply setting the price at the most likely WTP (i.e. $b_0 + b_i + f(x_{ui}, w)$), and we found the simple method to be statistically significantly worse.

3.3.1 Further Analysis

The feature weights of normalized bidding price k in the LR model may be of interest. Table 3.1 indicates that:

List price: normalized WTP decreases slightly as product's list price increases.

Number of reviews: normalized WTP decreases as the number of reviews increases. This is counter intuitive, as consumers presumably would tend to perceive goods with more reviews as more valuable. Of course, the expression is reduced form and for normalized (not direct) WTP, so future work may be able to find an explanation.

Discount: Products with larger Amazon discounts have higher normalized WTP. Like the list price impact, this effect probably also works mainly through the normalization (again recall Equation 3.5) but it is much stronger.

Average rating: Consumers are willing to pay more for products rated higher.

User rank: Not surprisingly, a consumer is willing to pay more for products she ranks more highly.

Switch brand: A consumer switching to a new brand tends to have lower WTP. This seems closely related to the *brand variety* effect, as variety seekers are more likely to switch brand than loyal customers.

Brand popularity: Not surprisingly, the more popular the brand, the more consumer is willing to pay.

3.4 Caveats

The analysis above noted many technical simplifications (e.g., normally distributed estimation errors for normalized WTP with constant variance), most of which can be relaxed in future work. This may be a good juncture to mention more general complications that we have put to the side.

- Most products have substitutes (and complements); we mostly sidestepped the resulting complications. As this line of research matures it may be useful to consider bundles of goods and sets of alternatives explicitly.
- Personalized discounts, if not properly framed, can provoke customer backlash. Large scale application of our suggestions must tread carefully to avoid causing resentment.
- Our profit metric only took into account unit production costs. These could easily include shipping and handling, and researchers with access to cost data should have no problem extending our approach. However, a different treatment would be required to account for the costs of running a recommender system, or the costs of eliciting WTP. Presumably these are an order of magnitude smaller than production costs but they still could be important.
- Our Mechanical Turk subjects were convenient and turned out to be quite useful. There is no reason to believe that naturally occurring E-commerce customers will behave in qualitatively different ways, but that remains an empirical question to

be tested by researchers with access to large numbers of such customers. True A/B testing on that subject pool would be the gold standard.

3.5 Related Work

Most existing research in recommender systems overlooks a factor that is extremely important in a consumer’s decision making: the product price. One notable exception is [34], which shows that price is a factor that is transferrable across categories to help recommendation system predictions. Another exception, [168], explores price’s marginal net utility role in E-commerce recommendation. However, these works still assume price is given and focus on boosting recommendation prediction performance with price as additional information; they do not treat prices as controllable factors to influence user decisions.

Two recent pilot studies examine pricing strategies in recommendation systems. Backhaus et al. [14] ask consumers questions over the phone, cluster consumers into different groups, used conjoint analysis to estimate customers’ WTP, and then use the estimates to price a small set of intangible value-added services on B2B market when making recommendations. Massoud and Abo-Rizka[113] propose a conceptual model of personalized pricing recommender systems. However, no experiments have been done to implement or study the proposed conceptual model. Kamishma and Akaho [73] go a little further and proposed a method for adding price personalization to standard recommendation algorithms based on customer preference data and purchasing history.

However, only artificially generated data (based on the MovieLens dataset) are used. It is not clear how the approaches proposed by these pilot studies can be applied in a real E-commerce system with a large and varied set of users and many products of different types, brands and price ranges.

There is an established literature in economics and marketing showing that personalized promotion is an important marketing tactic for increasing sales volume [66]. Roughly speaking, there are four empirical approaches: estimating WTP from transactions data, direct surveys, indirect surveys or laboratory auctions. Transaction data is incentive compatible in that it represents actual purchase decisions. However, the transaction price is only the lower bound on WTP, and equally relevant non-purchase decisions are missing from transactions data. Direct surveys (see, e.g., [157, 55]) estimate a consumer's WTP by directly asking indicate their maximum acceptable price for a given product [1]. However, as pointed out by Breidert et al. [22], direct surveys are not incentive compatible — the respondent is not motivated to reveal his or her true WTP, and often is motivated to understate it substantially. Indirect surveys offer respondents a list of alternative products (often hypothetical products with their properties varied independently) and ask them to either to rank them according to personal preference [106] as in conjoint analysis, or simply to choose the most favored alternative as in choice-based conjoint analysis (CBC)[99, 8, 9, 11, 12, 47, 51]. Similar to direct surveys, the conjoint analysis and CBC are not incentive compatible. To overcome this problem, Ding et al. in [47] propose an ICBC - incentive aligned choice-based conjoint method, which applies the BDM mechanism to the WTP inferred from conjoint analysis. Dong

et al. in [51] propose ranking revealed products based on the WTP inferred from conjoint analysis data. The authors argue their proposed approach is incentive-aligned as subjects will receive top ranked products.

The remaining empirical approach is WTP elicitation via laboratory auction. The famous Vickrey auction [166] requires each bidder to submit a sealed bid (not seen by other bidders); the bidder with the highest bid wins the auction and pays a price equal to the second highest bid. This procedure is incentive compatible. The intuition is that a person's own bids only determine whether or not she wins the auction but never affects the price that she pays, and therefore she has no incentive to understate (or overstate) her WTP. Closely related to the Vickrey auction is the Becker-DeGroot-Marschak (BDM) mechanism to elicit WTP [19]. Under BDM, each bidder submits a bid to purchase a product. A sales price is randomly drawn from an interval which covers all plausible bids. If that sales price is lower than a participant's bid, then she receives the product and pays the sales price. The BDM is theoretically incentive compatible for the same reason as the Vickrey auction; indeed (although it was invented independently) it is equivalent to a Vickrey auction with two bidders, one human and the other an automaton who bids randomly. Our own approach exploits the fact that BDM is easier to operate in an E-commerce setting since, unlike the Vickrey auction, it does not require gathering all participants' bids in order to determine the winner.

Despite considerable research on WTP elicitation [117] and existing pilot research on pricing strategies when recommending, we still face several challenges when adapting existing approaches to real world E-commerce settings. Existing approaches

often require an artificial laboratory setting with only a handful of products preselected by researchers, and/or with product data generated artificially. In E-commerce systems, however, consumers are exposed to a much larger number of products of different types and price ranges, and the system needs to estimate WTP for products that might be of interest to an individual user. The gaps between the settings make it unclear whether the conclusions derived from laboratory experiments generalize to E-commerce settings.

3.6 Summary

This chapter proposes to include personalized promotion in E-commerce recommender systems. We develop a lottery-auction mechanism to elicit consumers' willingness to pay on a small subset of products, and a machine learning model to predict each consumer's WTP on a wide range of other products. We demonstrate the feasibility of the proposed approach in an experiment with real world products from Amazon.com and subjects recruited from Amazon Mechanical Turk. The results suggest that personalized promotion leads to significantly higher profits for sellers compared to the baseline pricing.

Our work also shows the viability of doing E-commerce experiments via crowdsourcing; our Mechanical Turk subjects turn out to be quite useful. Besides recommender systems, the proposed approach also has practical implication for managerial marketing.

Our work is only first step; much remains to be done. First and foremost, per-

sonalized promotion and recommendation should be considered jointly within a unified framework, and not remain as separate problems. Second, we focus on seller's profit as the evaluation metric. However, we believe it is crucial to include consumer surplus into the objective function and we see no conceptual obstacles in doing so. Finally, as alluded to in the caveats, it will be crucial to study the longer term effects on consumer satisfaction with personalized promotion. We hope that researchers in industry, with better access to normal E-commerce consumers, will be inspired to do so.

3.7 Acknowledgment

This chapter is based on the ACM RecSys2015 conference paper [187]. The paper is the result of a joint effort of Qi Zhao, Yi Zhang, Daniel Friedman and Fangfang Tan. The specific contribution of each co-author is as follows: Qi Zhao came up with the initial experimental design and improved it iteratively with the help from the rest of the authors. Qi Zhao implemented the proposed idea by developing an E-commerce websites, hiring experimental subjects from Amazon Mechanical Turk and building and testing the machine learning models. He is also a major contributor of the writing. Daniel Friedman helped evolve the design by offering high leveled economic insights, mainly including incentive compatibility and cost efficiency. Daniel also contributed significantly to polishing the final stage writing. Yi made several key contributions to the WTP mechanism design and experimental evaluation. She is also a major contributor to the final stage writing. Fangfang took an active involvement of the experimental design

in the early stage. She, in a joint effort from Daniel, has provided useful suggestions regarding revealing subjects' preference in an incentive compatible manner.

Chapter 4

For consumer: Multi-product utility maximization based recommendation

It has been well recognized that products are related. Two products could be *substitutes* – buy A instead of B or *complements* – buy A together with B. Identifying and making use of such relationships is useful for recommendation systems. For example, knowing a consumer's recent purchase of a digital camera, the recommender system should avoid recommending more digital cameras and instead recommend matching lenses or photography books. Another example is to recommend a shower faucet and a matching valve at the same time.

In economics, utility is used to measure the value of a product perceived by the consumer, and it is fundamental for describing and predicting consumer choices. With a utility metric, a good recommendation should be product(s) with the biggest utility for a given consumer. The existence of inter-product relationships makes modeling

product utility a non-trivial task. For example, how much utility will an additional camera provide versus a complementary lens? The task becomes even harder when the relationship is less obvious – e.g. what is the utility of the camera if the consumer already owns an iPhone 6S with a built in camera? To answer these questions, a principled approach is needed to quantitatively measure the total utility of two or more products.

In this chapter, we propose to measure the total utilities for multiple products and recommend by *Multi-Product Utility Maximization* (MPUM). We first propose to derive two utility functions given their *Marginal Rate of Substitution* (MRS) specification. Our derivation approach can naturally lead to the well-known *Constant Elasticity of Substitution* (CES) utility function; then we extend the pairwise utility function to more than two products. We assume users make choices to maximize multi-product utility and use a multinomial consumer choice model for that; then the multi-product utility model can be learned to maximize the likelihood of observed user data.

The rest of the chapter is organized as follows: we introduce some basic definitions and concepts that form the basis of this work in Section 4.1; we then propose our MPUM framework as well as the personalized transaction-based recommendation strategy in Section 4.2 and Section 4.3, respectively. In Section 4.4, we present extensive experimental results based on two different real-world datasets. We review the related work in Section 4.5 and conclude this work with some future research directions in Section 4.6.

4.1 Basic Components

In this section, we design some of the key components for our model, and these components will be further integrated into our Multi-Product Utility Maximization (MPUM) framework later.

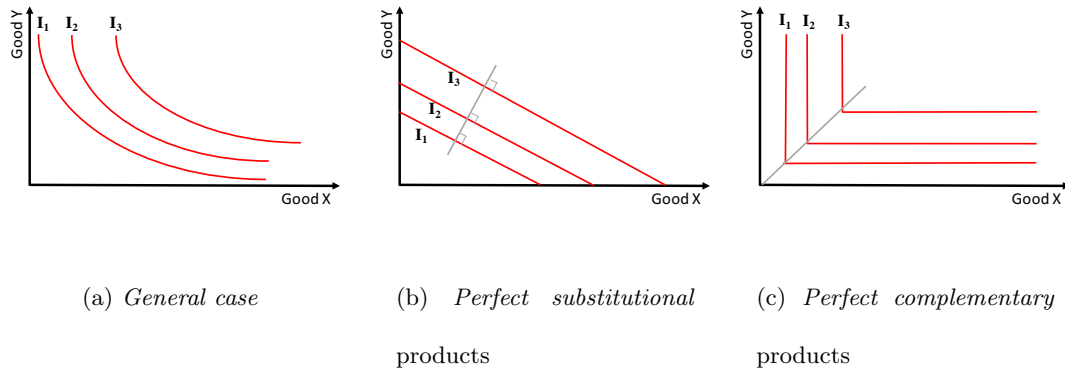


Figure 4.1: Illustrative indifference curves for common pairs, perfect substitutional pairs, and perfect complementary pairs. The utilities of the three illustrative curves satisfy $I_1 < I_2 < I_3$.

4.1.1 Indifference Curves

In economics, *indifference curves* are used to describe the relationship between two arbitrary products. More specifically, economists are interested in knowing how one product can be substituted by the other product. As illustrated in Figure 4.1, each indifference curve represents the compositions of two given products that give the same utility. Indifference curves should possess the following properties:

- Any two points of the same curve give the same utility.
- Curves do not intersect.

- The tangent of any point on the curve is negative, meaning the increment of the consumed quantity of one product requires the decrease of the other product so that the utility remains unchanged.

Let q_j, q_k denote the consumed quantity of product j and product k respectively. Since points of the same curve have the same utility, the total derivative at any point should be 0, we have:

$$\begin{aligned} dU(q_j, q_k) &= \frac{\partial U}{\partial q_j} dq_j + \frac{\partial U}{\partial q_k} dq_k \\ &= U'_{q_j} dq_j + U'_{q_k} dq_k = 0 \end{aligned} \quad (4.1)$$

Let $h(q_j, q_k) = \frac{dq_j}{dq_k}$ denote the Marginal Rate of Substitution (MRS) at point (q_j, q_k) , we have,

$$h(q_j, q_k) = \frac{dq_j}{dq_k} = -\frac{U'_{q_k}}{U'_{q_j}} \quad (4.2)$$

Intuitively, the larger $|h(q_j, q_k)|$, the more consumption of product j is needed to compensate the decrease of the consumption of product k . As a matter of fact, MRS can fully capture the relationship between two products. To understand this, it might be helpful to look at the three indifference curve patterns as illustrated in Figure 4.1. Each pattern represents a typical product relationship. Figure 4.1(a) corresponds to the generic case where MRS transits smoothly; Figure 4.1(b) corresponds to *perfect substitutes* where the MRS is a constant. In other words, two products can be exchanged at a fixed rate at any time. This can happen when two products are interchangeable, e.g. pens that differ only in color and consumer is indifferent about color. Figure 4.1(c) corresponds to *perfect complements* where the utility is determined by the minimum of

the two product quantity. One might understand this by thinking about the utility of left and right shoes – given a certain quantity of left shoes, the utility will not change by having more right shoes than left shoes and vice versa. Compared to Figure 4.1(a) and 4.1(b), the MRS of perfect complements is more tricky: it changes from infinity to zero at a certain point. So far, it can be seen that MRS can indicate whether two products are substitutes or complements.

4.2 Multiple Product Utility Maximization Framework

In this section, we provide detailed formal treatment of the whole framework by putting the aforementioned essentials together.

4.2.1 Modeling Marginal Rate of Substitution

First, our goal is to find a proper utility functional form for $U(q_j, q_k)$ so that it can capture all possible product relationships as shown in Figure 4.1. However, the right functional form for the utility function is not obvious, and it is not practical for us to try all possible alternatives of $U(q_j, q_k)$ by testing them against the cases in Figure 4.1 to see how well the utility function can model substitutional and complementary products. Since product substitute and complementary relationships are better illustrated by MRS, we propose to find a proper functional form for MRS, based on which we recover the utility function by solving partial differential equations.

As MRS can be derived from the indifference curve (i.e. $U(q_j, q_k) = const.$), we can alternatively express q_j as a function of q_k , i.e., $q_j = f(q_k)$. The MRS defined

Table 4.1: Choices of $h(q_k, q_j)$ and its respective utility function. $z(\cdot)$ denotes any monotonic function. Please refer to text for more details.

	Polynomial	Exponential
$h(q_j, q_k)$	$-\frac{a}{1-a} \left(\frac{q_j}{q_k}\right)^b$ $0 < a < 1, b > 0$	$-\frac{a}{1-a} e^{b(q_j - q_k)}$ $0 < a < 1, b > 0$
$U(q_j, q_k)$	$z(aq_j^{1-b} + (1-a)q_k^{1-b})$	$z(ae^{-bq_k} + (1-a)e^{-bq_j})$

in Eq. (4.2) becomes,

$$\frac{dq_j}{dq_k} = f'(q_k) = h(q_j, q_k) \quad (4.3)$$

where h is the MRS function that we need to decide.

When choosing h , we are mainly concerned about two aspects: mathematical convenience and flexibility. Thus we propose to consider two choices listed in Tab 4.1: polynomial functional form and exponential functional form.

Regardless of the specific form of h , the problem of recovering $U(q_j, q_k)$, or $f(q_k)$, boils down to solving the differential function as Eq. 4.3 for f . We can see the results for each alternative of h in Tab. 4.1.

4.2.1.1 Polynomial Function

We first take a brief look at h to see whether it is expressive enough, i.e., whether it can describe the three cases shown in Figure 4.1. The answer is positive. When $b = 1$, the resulted MRS is constant $\frac{a}{1-a}$, which is for the case of perfect substi-

tutes; when b goes to very large ($+\infty$), h is large when $\frac{q_j}{q_k} > 1$ and immediately drops to near 0 when $\frac{q_j}{q_k} < 1$, corresponding to the perfect complements case; when $0 < b < 1$, the resulted MRS is for the general case shown in Figure 4.1(a).

After applying some differential equation tricks to Eq. (4.3), we reach the following equation,

$$\left(aq_j^{1-b} + (1-a)q_k^{1-b} \right) = \text{const.} \quad (4.4)$$

Remember that MRS is defined when utility is set to an unknown constant. The above equation suggests that the utility function might be some monotonic function of the left side of the above equation, namely,

$$U(q_j, q_k) = z \left(aq_j^{1-b} + (1-a)q_k^{1-b} \right) \quad (4.5)$$

where $z(\cdot)$ is any monotonic function such as log and power. In particular, when $z(x) = x^{\frac{1}{1-b}}$, it results in the well known *Constant Elasticity Substitution* (CES) utility function in Economics, and $s = \frac{1}{b}$ is called the *Elasticity of Substitution*, which denotes the degree of substitutability/complementarity between a pair of products. Specifically, the utility function models (perfect) substitutional product pairs when s is sufficiently large (towards $+\infty$ in extreme cases), and (perfect) complementary pairs when s is sufficiently small (towards 0 in extreme cases).

4.2.1.2 Exponential Function

Similarly, we examine the exponential function form (Tab. 4.1) for different values of b . When $b = 0$, the resulted MRS is constant $\frac{a}{1-a}$; when b goes to ∞ , the

resulted MRS goes to infinity when $q_j > q_k$, and drops to zero when $q_j < q_k$. These suggest that the exponential functional form can capture complements and substitutes.

Solving the differential equation Eq. 4.3 yields:

$$ae^{-bq_k} + (1 - a)e^{-bq_j} = \text{const.} \quad (4.6)$$

The corresponding utility function is,

$$U(q_j, q_k) = z \left(ae^{-bq_k} + (1 - a)e^{-bq_j} \right) \quad (4.7)$$

4.2.2 Multi-product Utility Modeling

In practice, it is very common that there are more than two products in a single transaction/order and it is desirable for us to represent the utility of an arbitrary number of products. Let Ω_{it} be the set of products purchased by user i at time t . We consider the utility of Ω_{it} as the sum of the utility of all product pairs within Ω_{it} , namely,

$$U(\Omega_{it}) = \frac{1}{|\Omega_{it}|-1} \sum_{j,k \in \Omega_{it}, j \neq k} U(q_j, q_k) \\ = \begin{cases} \frac{1}{|\Omega_{it}|-1} \sum_{j,k \in \Omega_{it}, j \neq k} \left(a_{ij}q_j^{1-b_{jk}} + (1 - a_{jk})q_k^{1-b_{jk}} \right)^{\frac{1}{1-b_{jk}}} & \text{for Eq.(4.5)} \\ \frac{1}{|\Omega_{it}|-1} \sum_{j,k \in \Omega_{it}, j \neq k} \left(a_{ij}e^{-b_{jk}q_j} + (1 - a_{jk})e^{-b_{jk}q_k} \right) & \text{for Eq.(4.7)} \end{cases} \quad (4.8)$$

where a_{ij} and b_{ij} are product pair specific parameters, $|\Omega_{it}|$ is the number of products in set Ω_{it} , and $U(q_j, q_k)$ is the utility of two products described in the previous subsection.

4.2.3 CF-based Reparameterization

As seen in Eq. (4.8), there are two unknown parameters a_{jk}, b_{jk} for product j and k . Inspired by *Matrix Factorization* (MF) based *Collaborative Filtering* (CF) as shown in Equation 2.6, we propose to model the parameters as below,

$$a_{jk} = \sigma(\alpha + \beta_j + \beta_k + \vec{x}_j^T \vec{x}_k) \quad (4.9)$$

$$b_{jk} = \exp(\mu + \gamma_j + \gamma_k + \vec{p}_j^T \vec{p}_k) \quad (4.10)$$

$$\vec{x}_j, \vec{p}_j \in \mathbb{R}^d, \beta_j, \gamma_j, \alpha, \mu \in \mathbb{R}$$

where $\sigma(\cdot)$ is a Sigmoid function that ensures $0 < a_{jk} < 1$ and an exponential function that ensures $b_{ij} > 0$. The parameters now are,

$$\Theta = \{\vec{x}_j, \vec{p}_j, \beta_j, \gamma_j, \alpha, \mu\} \quad (4.11)$$

4.2.4 Discrete Choice Modeling

In economics, *discrete choice models* characterize and predict consumer's choices between two or more alternatives, such as buying Coke or Pepsi, or choosing between different hotels for traveling. In practice, for a given consumer, we only observe the products that the consumer purchases at a moment t . These products are represented by a set Ω_{it} where i is consumer index and t is time. Please note that each element of Ω_{it} is a product with quantity information attached. When a consumer is deciding which products to purchase, it is typical that the consumers have multiple purchase alternatives under consideration. In theory, any subset of the universe of the products could be a purchasing alternative for Ω_{it} . However, for computational efficiency, it is

feasible to consider a small number of purchase alternatives. This requires a sampling procedure that generates the alternatives for Ω_{it} . We denote such generation procedure as a function $g(\Omega_{it})$, that is, for any purchased products Ω_{it} , the function will generate a certain number of purchase alternatives and each alternative is a set of products. As each element of $g(\Omega_{it})$ is an alternative that is considered but not purchased by the consumer, $g(\Omega_{it})$ can be understood as a set of negative samples and Ω_{it} is the positive sample. Let $\Pi_{it} = \Omega_{it} \cup g(\Omega_{it})$ to represent all samples for user i at time t . The k -th sample of Π_{it} is denoted as Π_{it}^k . For convenience, we assume Π_{it}^1 is the positive sample Ω_{it} , and Π_{it}^1 is chosen over the negative samples $\Pi_{it}^k, k > 1$ by the consumer. Each element of Π_{it} represents a choice for the consumer and economists have developed *random utility models* (RUMs) for the discrete choice problem[116], that is, models for characterizing Π_{it}^1 is chosen over $\Pi_{it}^k, k > 1$. RUMs attach a random utility to each choice,

$$U_i(\Pi_{it}^k)' = U_i(\Pi_{it}^k) + \epsilon_k \quad (4.12)$$

where ϵ_k is a random variable that follows a certain probability distribution. The probability that a consumer chooses Π_{it}^1 (i.e. Ω_{it}) over other alternatives is:

$$P\left(U_i(\Pi_{it}^1)' > U_i(\Pi_{it}^k)'\right) = P\left(\epsilon_k - \epsilon_1 < U_i(\Pi_{it}^1) - U_i(\Pi_{it}^k)\right) \quad (4.13)$$

where $k = 2, \dots, |\Pi_{it}|$. If ϵ_1 and ϵ_k follow *iid* Gumbel distribution, it can be shown that the probability of choosing Π_{it}^1 is the following multinomial logistic model (MNL):

$$P(y_{it} = 1) = \frac{\exp(U_i(\Pi_{it}^1))}{\sum_{k=1}^{|\Pi_{it}|} \exp(U_i(\Pi_{it}^k))} \quad (4.14)$$

Alternatively, if ϵ_k follows a Gaussian distribution, $P(y_{it})$ becomes a Probit models. In the rest of this chapter, we will use multinomial logistic regression.

4.2.5 Budget Constraint

The theory of consumer choice in microeconomics[56] is concerned with how consumers maximize the utility of their consumption subject to their budget constraint. The utility of consumption is determined by consumers' preferences and their corresponding utility mode as explained in Section 4.2.2. In economics, the budget is usually included as constraint for consumer choice optimization. Here, for simplicity, we include budget in an implicit manner. Remember we introduced the purchase alternative generation function $g(\Omega_{it})$ in previous section, but have not mentioned how it is specified. In fact, function g is implemented through the budget constraint. More specifically, for any Ω_{it} , we first derive the consumer's budget as the sum of the expense for each product in Ω_{it} . With the same products as in Ω_{it} and the same budget, we then generate different quantity assignments for the products. Each quantity assignment is a purchase alternative for Ω_{it} . So each element of $g(\Omega_{it})$ differs from Ω_{it} only by the product quantities.

4.2.6 Model Parameter Learning

Given the observed transactions/orders and the consumer discrete choice modeling framework, the model parameters Θ can be optimized by minimizing the *Negative*

Log-Likelihood (NLL) of the training data:

$$\begin{aligned} & \underset{\Theta}{\operatorname{argmin}} \operatorname{NLL}(D; \Theta) \\ & = - \sum_{i,t:I_{i,t}=1} \log(P(y_{it} = 1)) + \eta \|\Theta\|^2 \end{aligned} \tag{4.15}$$

where D is the training dataset. $I_{it} = 1$ if user i places an order at time t . $P(y_{it})$ is the multinomial logistic regression model described in Section 4.14. $\eta \in \mathbb{R}^+$ is regularization coefficient which is determined using cross validation. Θ is the model parameters as defined in Equation 4.11.

The optimal model parameters can be found using gradient based methods such as stochastic gradient descent.

4.3 Multi-product Recommendation

The objective of our recommendation algorithm is to recommend a set of products that give the maximum utility. It is very challenging to recommend a set of items to the consumer without knowing what products the consumer already has. To avoid such situation, we assume that the consumer already has a few products in the shopping list and the recommender systems need to recommend more products for the same list. Let $\Gamma_{it} \in \Omega_{it}$ be the items in the shopping list. The goal of recommendation is to identify a set of items that brings the most utility gain if the items are added to the existing product list. Let Γ'_{it} be the recommendations and the size of Γ'_{it} is K . The

recommendation task can be formally expressed as,

$$\operatorname{argmax}_{\Gamma'_{it}} U(\Gamma_{it} \cup \Gamma'_{it}) \quad (4.16)$$

where $|\Gamma'_{it}| = K$ and $\Gamma'_{it} \cap \Gamma_{it} = \emptyset$. Due to the large number of products in the system, it would be computationally expensive to evaluating all Γ'_{it} in an exhaustive manner, especially when K is large. For computational efficiency, we construct Γ'_{it} in a greedy manner. The procedure is described in Algorithm 1,

Algorithm 1: Multiple-product Recommendation by Greedy Matching

Data: Initial product list Γ_{it}

Result: Recommendations: Γ'_{it}

```

1  $\Gamma'_{it} = \emptyset$ ;
2 for  $step \leftarrow 1$  to  $K$  do
3     // Find a product that gives the most utility gain
4      $\operatorname{argmax}_{j \notin \Gamma_{it}} U(\Gamma_{it} \cup \{j\})$ ;
5     //  $j$  is the solution of the above objective function
6      $\Gamma'_{it} = \Gamma'_{it} \cup \{j\}$ ;
7      $\Gamma_{it} = \Gamma_{it} \cup \{j\}$ ;
8 end

```


4.4 Experiment

We studied the proposed framework based on two real world E-commerce datasets. The experimental design and results are reported in this section.

4.4.1 Dataset Description

The following two real-world datasets are used in our experiments:

Table 4.3: Basic statistics of the two datasets

Dataset	#Transactions	#Products	Average Size	Train/Test
shop.com	86k	370k	~ 8	80%/20%
Amazon	7.8k	18k	~ 12	

*shop.com Data*¹ : This is a dataset from shop.com, which is an E-commerce website. This dataset is proprietary and is not publicly available. Each record in the dataset is a purchasing transaction with consumer ID, product price, product quantity and purchasing time. We treat each transaction as a positive training data point for Equation 4.14. The key data statistics are summarized in Table 4.3. As we are focusing on multiple products, we processed the dataset by removing transactions with less than two products.

Amazon Baby Registry Data Amazon’s Baby Registry² allows consumer to add and

¹<http://shop.com>

²<https://www.amazon.com/babyregistry>

Table 4.2: Evaluation results for Top- K recommendation performance on Precision, Recall, and F_1 -measure. The baseline algorithms are *Bayesian Personalized Ranking* (BPR) and *Matrix Factorization* based *Collaborative Filtering* (CF) as shown in Equation 2.6. Please refer to Section 4.4.3 for more information about BRP.

Dataset	Amazon Baby Registry Transactions								
@ K	1			5			10		
Method	CF	BPR	MPUM	CF	BPR	MPUM	CF	BPR	MPUM
Precision (%)	0.092	0.117	0.275	0.437	0.473	0.609	0.262	0.513	0.669
Recall (%)	0.074	0.112	0.103	0.761	0.866	1.279	1.178	1.150	2.844
F_1 -measure	0.082	0.114	0.150	0.555	0.612	0.825	0.429	0.710	1.083

Dataset	shop.com Transactions								
@ K	1			5			10		
Method	CF	BPR	MPUM	CF	BPR	MPUM	CF	BPR	MPUM
Precision (%)	0.022	0.076	0.470	0.012	0.038	0.286	0.012	0.026	0.160
Recall (%)	0.017	0.003	0.465	0.035	0.013	1.390	0.073	0.185	1.531
F_1 -measure	0.019	0.006	0.467	0.018	0.019	0.474	0.021	0.046	0.290

manage products for babies. Each registry is like a wishlist which contains a list of products the list owner wants to purchase or receive. As the lists are publicly available,

we crawled the lists and their products to generate this data set. Each product comes with title, price, brand, category information. Some of the key statistics of the dataset are summarized in Table 4.3. We treat each wish list as a positive training point for Equation 4.14.

Each dataset involved can be viewed as a collection of transactions. Each transaction holds a set of products a consumer purchased or wanted at a certain time. The transactions are randomly split into two subsets - 80% of them are used for model training and the other 20% for testing. For each testing transaction, a small portion (20%) of the products are randomly masked and predicted by the recommendation algorithm based on other observed products in the same transaction.

For each training transaction, negative samples (purchase alternatives not chosen by the user) are generated, as they are required in Equation 4.14. For computational efficiency, we only generate negative product sets closer to the target positive chosen set. Given a chosen product set (an order or a wishlist), we assume the budget is the total cost of the products in the set. We keep the products unchanged and enumerate all quantity combinations that are subject to the same budget) constraint. Each quantity combination acts as a purchasing alternative. For computational efficiency, we further limit the size of Π_{it} to be 10 by randomly sampling from $g(\Omega_{it})$.

Table 4.4: Number of model parameters for each algorithm.

. As our method represents item-item relationship, the resulted parameters is much fewer than other baseline algorithms.

	MPUM	BPR	CF
Number of parameters	$N * D$	$(M + N) * D$	$(M + N) * D$
M : Number of users, N : Number of items, D : Latent vector dimension			

4.4.2 Evaluation Metric

Precision and recall at top-K are used for evaluation, as they are the most widely used ranking evaluation metrics in existing literature. Let Γ_i be the masked items in the i -th testing transaction and Γ'_i is a list of recommended items by the recommendation algorithm under consideration. The metrics are defined as follows:

$$\begin{aligned}
 \text{Precision@K} &= \frac{1}{N} \sum_{i=1}^T \frac{|\Gamma'_i \cap \Gamma_i|}{K} \\
 \text{Recall@K} &= \frac{1}{N} \sum_{i=1}^T \frac{|\Gamma'_i \cap \Gamma_i|}{|\Gamma_i|} \\
 F_1\text{-measure@K} &= \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}
 \end{aligned} \tag{4.17}$$

where K is the length of the recommendation list.

4.4.3 Experimental Results

We investigated the performance of our MPUM framework for the task of product recommendation for a transaction. For performance comparison, we considered the

Matrix Factorization based *Collaborative Filtering* (CF) algorithm described in Equation 2.6 and *Bayesian Personalized Ranking* (BPR) [127] as the baseline algorithms. BPR optimizes item ranking at individual level. Like MF based CF algorithm, BPR also represents user and item by latent vectors. It works by sampling non-purchased items and constructing preference pairs by pairing purchased items and non-purchased items. For example, if a user purchases product A and the algorithm samples product B which is not purchased by the user, a preference pair can be formed as $A \succ B$. BPR then represents each preference pair by a sigmoid function. The learning objective of BPR is to fitting the ranking pairs generated from the training dataset. Please refer to the paper for more details[127]. Both CF and BPR recommend by predicting the purchasing quantities directly. $|\Pi_{it}|$ in Eq. (4.14) is set to 10 and SGD learning rate is set to 0.01. For fair comparison, shared parameters of different models are set to be the same: the latent factor size is set to 10 and the regularization coefficient η is set to 0.01.

The evaluation results on Amazon and shop.com datasets are reported in Table 4.2, and the largest value on each dataset and for each evaluation measure is significant at the 0.01 level.

It can be seen from the results that our proposed MPUM algorithm outperforms the baseline algorithms in nearly all the cases, and in particular, the performance advantage is more pronounced on the shop.com dataset. A possible reason is that the shop.com dataset has much lower density (0.00205%) than the Amazon dataset (0.0655%). Compared to baseline algorithms, our method is less sensitive to low den-

sity. This is because the CF and BPR approaches introduce latent vectors for users (i.e., transactions in our problem) and products, and then learn the vectors through user-product interaction pairs, while our MPUM algorithm only concerns product-product relationships and models the transactions indirectly through its products without the need to consider the vastly sparse user-product pairs. As a result, our MPUM requires much less model parameters than the baseline algorithms. Please refer to Table 4.4 for the number of parameters for each algorithm.

4.4.4 Further Analysis: Empirical Study of Economic Intuition

We conducted some further analysis to investigate the economic intuition of our approach in terms of learned utility functions. In our analysis, we focus on the CES utility function in Eq.(4.5), because by examining the Elasticity of Substitution (ES) for real-world products learned by our model, we hope to find intuitive explanations for our principled economic-driven approach in practical applications.

We also compute the average ES for each product in the Amazon Baby Registry dataset by averaging its estimated ES with all other products. We find that the popularity of a product in the dataset is highly negatively correlated with the corresponding ES. This means popular products have relatively smaller ES values, which suggests popular items tend to be more complementary with other products.

More specifically, Figure 4.2 shows the logarithm of popularity of a product (y axis) against the average ES of the product (x axis). The correlation between $\log(\text{popularity})$ and ES values is -0.916 for these products. Because we care more about

the product ranking lists for recommendation rather than the absolute ES values in practice, we further rank the products according to ES and investigate the relation between $\log(\text{popularity})$ and the rankings (Figure 4.2, right). The correlation is -0.931. Further analysis shows that the products with small average ES values in Figure 4.2 are mostly baby care necessities (e.g., pacifier, plug cover, and teether) that are complementary with many products, which makes them generally popular in most of the transactions.

These findings are encouraging and suggest that our proposed utility maximization approach conforms with human intuitions. It makes it possible to discover product substitutional/complementary relationships from real-world transaction data automatically, based on combining machine learning techniques with principled economic theories.

4.5 Related Work

Most existing recommendation methods predict an individual product score for each user and rank products accordingly, without considering relationships between the recommended items. One major problem is that the top ranked recommendations might be very similar or even duplicates, which usually is not desirable. To address this issue, researchers proposed to diversify the recommendation results[71, 124, 188, 98]. As the potential benefits of diversity to individual users and business are huge, the diversity problem has been heavily studied, mainly on other datasets such as news, movies

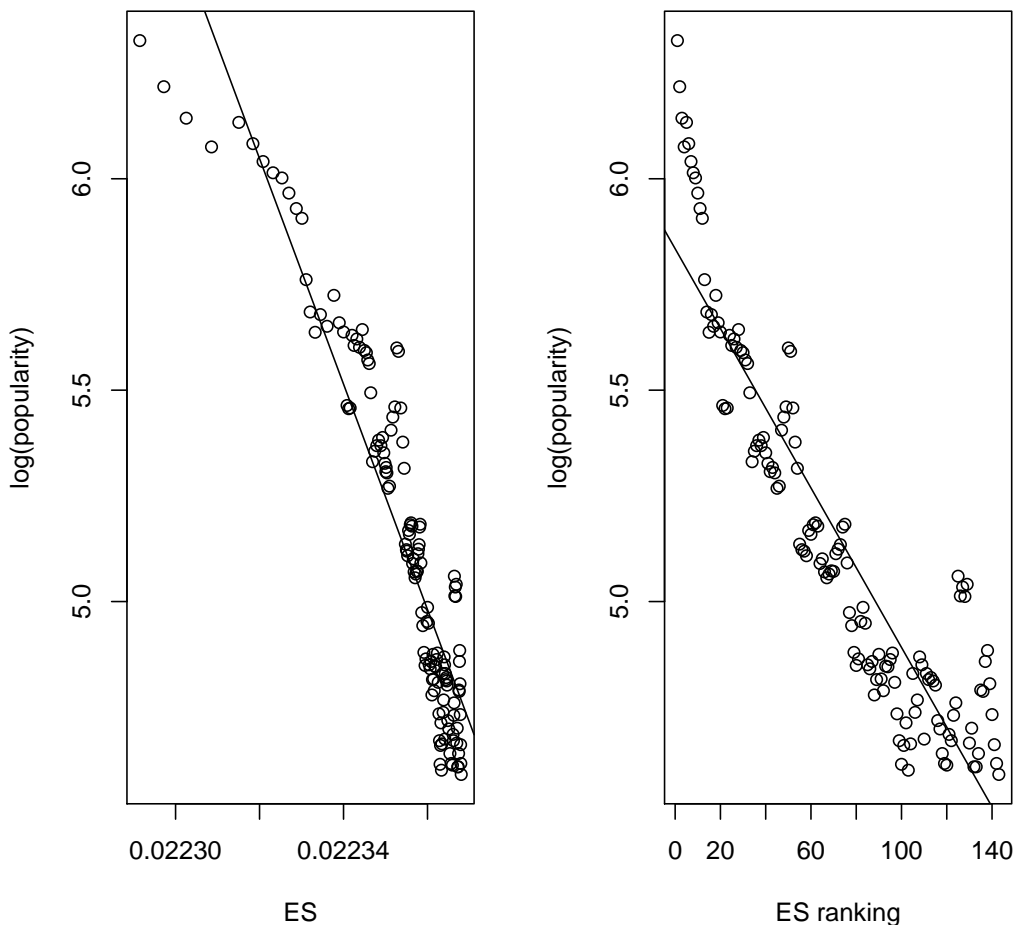


Figure 4.2: Scattered relations of the product popularity v.s. the average Elasticity of Substitution (ES) of the corresponding product as well as the the ranking of ES values.

and music [31][4]. Diversity is used either at recommendation candidates selection, at the item score prediction stage, or at the top-N product re-ranking/filtering stage after individual item scores have been predicted. A typical approach is to introduce certain diversity measures such as the number of categories/singers, relative share of recommendations above or below a certain popularity rank percentile [24], or measure over product graph [4]. Another approach is to use measures that will achieve diversity indirectly, such as the risk of a user portfolio of multiple products [149]. Although diversity

is not the focus of this chapter, the proposed method will lead to diversity naturally as the result of Diminishing Marginal Utility. How to trade off diversity and accuracy, and how to diversify differently for different product categories will be inferred from the utility model learned from consumer choice data.

The most related work to ours is [114]. This study focuses on the category of clothes and assumes products purchased together are complements and products viewed together are substitutes. They showed that the co-viewed and co-purchased product relationships could be discovered based on the visual appearance of the clothes. However, relationships exist for different reasons for different products, and the visual methods will not generalize to all other product categories. In contrast, our approach is very general and directly leads to many different product recommendation results.

Another line of research related to this chapter is work about the *next basket* recommendation problem, which models the sequential pattern of user purchases and recommends a set of items for the user's next visit based on previous purchases. A series of methods have been developed for next basket recommendation [129, 172, 59, 40, 173], among which the Hierarchical Representation Model (HRM) [173] represents state of the art. HRM combines general taste by conventional CF and information from previous transaction aggregated by a nonlinear function. Although this chapter is not focused for the next basket recommendation problem, the proposed multi-product utility model can be applied to solve this problem. Assuming products the user has purchased before are already in the set and fixed, the system just needs to find and recommend more products to optimize the total utility for the user.

In recent years, there have been some efforts in bringing economic principles into E-commerce recommendation systems. In [169], the authors propose to adopt the law of diminishing marginal utility at the individual product level so that perishable and durable products are treated differently. In [187], a mechanism is developed to estimate consumer's willingness-to-pay (WTP) in the E-commerce setting and the estimated WTP is used to price products at the individual level so that seller's profit is maximized. In [186], a total surplus based recommendation framework is proposed to match producer and consumer so that the total benefit is maximized. Our research falls into this category and tries to handle the multi product recommendation problem based on solid economics principles and practical recommendation techniques.

In particular, recognition of product substitutability and complementarity has been considered important for the study of how the demand of one product is affected by other products[25, 139, 140, 150]. Our proposed research is motivated by these existing economic studies.

4.6 Summary and Future Work

Utility is commonly used by economists to characterize consumer preference over alternatives and it serves as the cornerstone for consumer choice theory[56]. Motivated by existing research in economics, we introduced a general utility based framework for multiple products recommendation. Starting with Marginal Rate of Substitution defined over the products indifference curve, we derived several candidate utility functional

forms that can model both substitutes and complements. The model parameters are learned based on existing consumer data. Recommendations of multiple products are generated by maximizing the learned utility model. Experimental results on both Amazon and shop.com e-commerce data sets demonstrated the effectiveness of the proposed approach for recommendation. Further analysis also shows complements and substitutes found by the model look reasonable.

Modeling the relationships between products is a fundamental problem for various recommendation tasks, such as package recommendation, next basket recommendation and top K product recommendation. Although our experiments are about top K products recommendation, the proposed framework can be applied to other usage scenarios in the future. We expect the proposed framework to complement some very different existing methods that implicitly capture product relationships, such as list-wise CF or list-wise learning to rank, and it would be interesting to compare them in the future. This is a first toward multi-product utility modeling and there is much room to further improve the techniques. For example, the functional form of MRS could be adjusted to capture other product relationships besides complements and substitutes. We can also introduce product features and user features into this framework. In Section 4.3, we used a greedy method to generate a set of recommendations. We admit that the greedy approach is a simple treatment due to the computational efficiency consideration. In the future, it is worthwhile to look into other more principled approach.

4.7 Acknowledgment

This chapter is based on the paper submitted to ACM SIGIR 2016 conference. The paper is the result of a joint effort of Qi Zhao, Yongfeng Zhang³, Yi Zhang and Daniel Friedman. The specific contribution of each co-author is as follows: Qi Zhao initiated the idea, formulated the learning framework, implemented and evaluated the proposed method on real world datasets. Yongfeng proposed the idea of deriving two product utility from their MRS function. He also did the comparative study and wrote the corresponding sections. Yi contributed to the experimental design by proposing to use the Amazon Baby Registry dataset for evaluation. She is also a major contributor of paper writing. Daniel offered valuable insights into the recovery of two product utility function based on the choice of their MRS function. He also contributed part of the conference paper writing. Further-more, he pointed out the limitations of the CES utility function and proposed to use Translog utility function as a replacement of CES. His valuable feedback on the current work could serve as important basis for further improvement and extension in future work.

³the first two authors contribute equally

Chapter 5

For platform: Total Surplus

Maximization based recommendation

In Chapter 3, we propose to maximize producer profit, or producer surplus, through customizing promotion rate at individual level. In Chapter 4, we propose to recommend products to consumers by maximizing consumer utility which can potentially lead to consumer surplus if the consumer utility becomes cardinal utility. So far, we have studied how product recommender systems might be designed to optimize one side's benefit but we have not discussed how recommender systems could improve the benefit of the entire market. In welfare economics, total surplus, as defined by Equation 2.18, is used to indicate whether or not a market is efficient. If the total surplus resulted from all sales in the market is maximized, market efficiency is achieved. The opposite of market efficiency is market failure. In this chapter, we focus on designing product recommender systems that optimizes the well-being of the platform based on

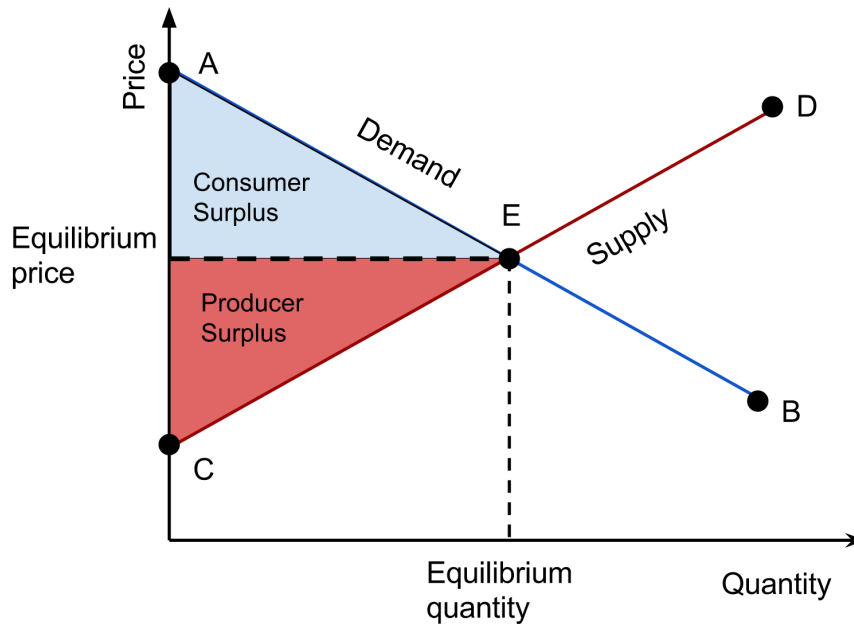


Figure 5.1: Illustration of total surplus for free market where the price is the equilibrium price. The total surplus is the sum of the consumer surplus and producer surplus.

total surplus maximization.

Why should we care about market efficiency? To answer this, let us consider the simplest market which comprises one product and three consumers. The quantity of the product is 2. The cost of the product is 10\$ and the consumer's WTP is 15\$, 20\$, 25\$. Each consumer only needs 1 product. There are 3 ways to allocate the product to the consumers and each allocation results different total surplus. Clearly, when the product is allocated to the two consumers of the highest WTP, it results the most total surplus – 25\$. Maximized total surplus is desirable, as it means the goods or services have been allocated to consumers who value them mostly. Efficient allocation is especially important when the resources are limited.

We have seen how seen the significance of total surplus in a hypothetical setting. As the objective of this Chapter is to propose a recommendation framework that

is based on optimizing total welfare of the market, it is necessary for us to know whether or not a free market is efficient. Consider the demand and supply of one product in the market. Figure 5.4 illustrates how demand and supply changes with the product price. The demand curve is downwards sloped which can be explained by *diminishing marginal utility* theory. For providers who produce the products, supply curve is used to describe the relationship between the quantity the providers want to produce and the price the product sells. The supply curve is upwards sloped which can be explained by the *diminishing marginal product* theory for production. Given the demand curve and supply curve, the product price has the tendency to move to the equilibrium point where demand and supply equal. As shown in Figure 5.4, the total surplus for market in equilibrium is the area between the demand curve and supply curve up to the equilibrium pricing point. In order to answer the question about whether the free market earlier, we need to study the total surplus for different prices, e.g. price below or above equilibrium price. It can be shown that the total surplus is maximized at the equilibrium price [56]. That being said, free market is efficient, or more precisely, Competitive Equilibrium is efficient absent externalities, in a complete information setting. Externalities refer to the impact one party have on others in ways that are not captured by the market price. For example, if production emits pollution, it potentially impacts many people and it imposes the society costs which are not taken into account by the producer until some other party imposes the costs on the producer. Our study of free market efficiency so far assumes the absence of externalities and complete information. Though both consumers and producers are self-interested and neither of them cares about whether the market

is efficient, surprisingly, they are together led by an invisible hand to an equilibrium where the market efficiency is achieved[2].

The above analysis seemingly suggests that since free market already maximizes total surplus, no extra effort is needed for matching products to consumers. However, this is not the case as the above analysis is based on very simplistic setting. For example, only one product is considered and the analysis studies how demand and supply is influenced by price, assuming other factors, e.g. consumer income, substitute products, are constant. The analysis of total surplus for real-world market, especially E-commerce, is much more complicated than the above one product market setting.

Online applications and services have grown tremendously in recent years. Consumers find products on E-commerce websites like Amazon or via social networks like Facebook, borrowers and lenders find each other via P2P lending services like Prosper, and freelancing websites like Amazon Mechanical Turk and Upwork match short term workers with employers. We call this matching between producers and consumers as *Online Service Allocation* (OSA). OSA seems destined to grow rapidly in the years ahead.

Because consumers typically have the right and the ability to choose freely among available online services, an enforced allocation is usually impractical. Service allocation therefore is typically performed online via search [23] and recommendation [132] systems. Search engines, such as Google or Amazon product search, leverage knowledge about consumers' intentions, while many recommender systems for products or social networks try to infer consumer needs without explicit user queries.

By its nature, service allocation is a two-sided matching activity, e.g., of consumers with producers. Economists since Adam Smith (1776) [2] have taken a balanced view of service allocation. The key insight is that maximizing total surplus – the sum of producers’ profit and consumers’ net benefit – is in the best interest of society, and potentially enables *both* sides to be better off than they would be if that maximum is not achieved.

Existing recommendation systems in online service matching platforms typically lack this balanced perspective. Most are designed with a focus on the benefits to only one side, while the benefits of the other side are ignored or even sacrificed, because there is always a potential conflict of interest between consumers and producers [56]. For example, the widely adopted Collaborative Filtering (CF) [132, 158] approach for recommendation is based on the preferences of consumers, and the benefits of producers play little role. Some online P2P lending systems focus on improving the revenue of lenders, while neglecting the surplus of borrowers. Such an imbalance is problematic because if one side does not gain much benefit, it may do better elsewhere and leave the platform.

The purpose of this chapter is to illustrate how to operationalize the economists’ insight in online service allocation into personalized recommendation systems to solve the problem for better social good of the online society. We propose a Total Surplus Maximization (TSM) framework to integrate both consumer surplus and producer surplus into recommendation systems. By TSM, the system creates a bigger pie (total surplus) for consumers and producers to divide. There is a large gap between the

traditional application of the economists' insight (a competitive market for a uniform commodity, with lots of small producers and consumers) and online allocation of very personalized and heterogeneous services. To fill the gap, we develop surplus-oriented metrics for personalized recommendations for heterogeneous products, and illustrate their use in several online markets.

We will offer evidence that the TSM framework can improve performance to the benefit of both sides. Indeed, our analysis and results for three real-world datasets (E-commerce, P2P lending, online freelancing) conclude that TSM-based recommendation performs better than standard recommendation techniques on traditional metrics. Furthermore, society as a whole is better off in terms of total surplus when better satisfying the needs of both the consumers and the producers.

The rest of this chapter is organized as follows. Section 5.1 presents the Total Surplus Maximization (TSM) framework for Online Service Allocation (OSA), and Section 5.2 tailors it for three typical applications. Results from fitting the models appear in Section 5.3. Section 5.4 notes connections to related work, and Section 5.6 offers concluding remarks.

5.1 The Framework for OSA

In this section, we propose our Total Surplus Maximization (TSM) framework for the problem of Online Service Allocation (OSA). For clarity in organization and easy understanding, we introduce the key components in a logical order, and then unify

Table 5.1: Specifications for three exemplary applications of the Total Surplus Maximization framework.

Application	E-commerce	P2P Lending	Freelancing
$CS_{ij}(Q_{ij})$	$\hat{a}_{ij} \ln(1 + Q_{ij}) - P_j Q_{ij}$	$(r_j - \hat{r})Q_{ij}$	$h(\hat{r}_{ij})s_j Q_{ij}$
$PS_{ij}(Q_{ij})$	$(P_j - c_j)Q_{ij}$	$(r_j^{max} - r_j)Q_{ij}$	$h(\hat{r}_{kj})s_j Q_{ij}$
\mathbb{S}	\mathbb{N}	\mathbb{R}_+	$\{0,1\}$
M	$M_j = \infty$	$0 < M_j < \infty$	$M_j = 1$
$p(Q_{ij})$	$p(Q_{ij} = q) = \lambda_{ij}^q e^{-\lambda_{ij}} / q!$	$\log(Q_{ij}) \sim \mathcal{N}(\mu_{ij}, \sigma_{ij})$	$p(Q_{ij} = 1) = \alpha_{ij}, P(Q_{ij} = 0) = 1 - \alpha_{ij}$
\bar{Q}_{ij}	λ_{ij}	μ_{ij}	$I_{\alpha_{ij}} = \max\{\alpha_{i'j}\}_{i'=1}^m$

these components to present the whole framework.

5.1.1 Problem Formalization

We consider the problem of providing online services, i.e., distributing goods among given users as an analogy, so that the total surplus is maximized during this process. The problem takes as given a finite set of consumers $i = 1, \dots, m$ with preferences not yet specified, and goods $j = 1, \dots, n$, each of which is produced by a single producer $k(j) \in \{1, \dots, r\}$.

Each good is available in limited quantity; $M_j \geq 0$ is the total amount of good j that its producer can supply to the system. For example, $M_j = 1$ in online freelancing networks because each job can only be provided once to only one freelancer; in P2P lending networks like Prosper, we have $0 < M_j < \infty$, which is the amount of money requested by each loan; for E-commerce websites like Amazon, however, we can treat

$M_j = \infty$ because for most normal goods, the producer can replenish the stock in case of an increasing market demand.

The Online Service Allocation (OSA) problem thus aims to find an Allocation Matrix $Q = [Q_{ij}]_{m \times n}$, where $Q_{ij} \geq 0$ is the quantity that consumer $i \in 1 \dots M$ is provided with good $j \in 1 \dots N$.

The capacity vector $M = [M_1, M_2, \dots, M_n]$ leads to the feasibility constraint $\sum_i Q_{ij} \leq M_j$ for each good j , i.e., $\mathbf{1}^T Q \leq M$, where $\mathbf{1}$ is a column vector of 1's. In different real-world application scenarios we may apply extra constraints on Q to meet specific task characteristics. For example, $Q_{ij} \in \mathbb{N}$ for E-commerce goods, or $Q_{ij} \in \{0, 1\}$ for online freelancing services.

The problem of OSA widely exists and finds its instantiation in a lot of on-line services or mobile applications wherever there is service consumption. Besides E-commerce, P2P lending, and freelancing services as we showed here, other applications include riding services such as Uber and Lyft, group purchase services such as Groupon, or even lodging services such as Airbnb.

5.1.2 Personalized Utility

Different consumers may experience different utility even from the same quantity of the same good. On E-commerce websites, for example, one consumer who owns an SLR camera may obtain a high consumer surplus when supplied with an SLR lens; however, the surplus may be extremely low when a lens is provided to someone without a camera. For P2P lending, similarly, the same amount of money could mean a huge

surplus to someone that has an urgent need (thus willing to accept a higher interest rate), while the surplus may be lower for those who are not that desperate for money (thus insists on lower interest rates).

This ‘personalized’ feature of utility makes up the inherent driving power for service allocation, which makes it reasonable for us to match the appropriate good with the appropriate consumer so as to maximize the potential total surplus in the whole system. This process can come in the form of personalized recommendation or intelligent marketing assistance to decision makers in practical applications.

In Section 2.3.1, we have already seen some of the commonly used utility functions. In the E-commerce application, we adopt the KPR¹ utility function for personalized utility $U_{ij}(q)$ on a consumer-to-good level, namely:

$$U_{ij}(q) = a_{ij} \ln(1 + q) \tag{5.1}$$

where $U_{ij}(q)$ indicates the utility when supplying a quantity q of good j to consumer i , which is parameterized by the personalized shape parameter a_{ij} .

The estimation procedure for a_{ij} varies with the availability of data and the applicable economic theory. For example, we adopt the *Law of Zero Surplus for the Last Unit* [54] for the inference of a_{ij} in E-commerce, while the property of percentage surplus is applied in freelancing. We will exposit in more details in the Model Specification section below.

¹King-Plosser-Rebelo

5.1.3 Total Surplus Maximization

Given perfect information on the personalized utilities $U_{ij}(q)$ and the cost function $C_j(q)$ for each good j , the TSM approach for online service allocation would seek an exact allocation matrix Q so as to maximize the total social surplus subject to the relevant constraints:

$$\begin{aligned} & \underset{Q}{\text{maximize}} \sum_i \sum_j (U_{ij}(Q_{ij}) - C_j(Q_{ij})) \\ & \text{s.t. } \mathbf{1}^T Q \leq M, Q_{ij} \in \mathbb{S} \end{aligned} \tag{5.2}$$

where \mathbb{S} is the set of feasible values for a specific application, e.g., $\mathbb{S} = \mathbb{N}$ for E-commerce and $\mathbb{S} = \{0, 1\}$ for online freelancing.

However, in practical applications we have only estimates of utilities and costs, and consumers may not always choose what appear to be the optimal quantities of goods. To account for measurement error (or decision error by consumers), we regard observed consumer choice as stochastic. That is, the elements Q_{ij} in allocation matrix Q are random variables with a probability distribution. The choice of probability distribution of Q , namely, $p(Q_{ij})$, depends on the application. Please refer to Table 5.1 for the distribution used for the applications investigated in this work. We pose the service allocation problem as maximizing *expected* total surplus:

$$\begin{aligned} & \underset{\Theta}{\text{maximize}} \sum_i \sum_j \int (U_{ij}(Q_{ij}) - C_j(Q_{ij})) p(Q_{ij}) dQ_{ij} \\ & \text{s.t. } \mathbf{1}^T \int Q p(Q) dQ \leq M, Q_{ij} \text{ is a random variable} \end{aligned} \tag{5.3}$$

where $p(Q_{ij})$ is the probability density function of each quantity Q_{ij} , $p(Q) = [p(Q_{ij})]_{m \times n}$, and the integral on Q is per element wise. Here Θ are the model parameters by which

the probability distribution $p(Q)$ can be determined. For example, in our E-commerce example, Θ are the user and item latent vectors for the *Collaborative Filtering* (CF) modeling on the arrival rate parameter λ_{ij} of the Poisson distribution defined in Equation 5.1. Please note that Equation 5.3 could reduce to optimizing w.r.t the mean value of the probability distribution of Q if Q can be separated out from $U_{ij}(Q_{ij})$. This is the case for the P2P lending example as shown in Equation 5.16. However, without loss of generality, we reserve this probability notation.

The model produces the optimal density functions $p(Q)$ as the final output, and we take the expectation $\bar{Q} = \int Qp(Q)dQ$ as the final allocation matrix to make system decisions. This probabilistic interpretation simplifies the computation in some applications, as explained in the next section.

5.2 Model Specification

We now discuss how to implement our TSM framework for online service allocation for three different online applications. Table 5.1 summarizes key specifications.

5.2.1 E-commerce

We first estimate personalized utility $U_{ij}(q)$ from consumer purchasing records. Although $U_{ij}(q)$ is not directly observed in the data, it is subject to the *Law of Zero Surplus for the Last Unit* [54]: as suggested in Figure 2.2, a rational consumer i will purchase quantity q_{ij} on good j because each unit up to that point brings additional satisfaction worth more than the price, but additional units beyond q_{ij} are not worth

the price. To spell it out, let $CS_{ij}(q_{ij}) = U_{ij}(q_{ij}) - P_j q_{ij}$ be the *Consumer Surplus* (CS) obtained from such a purchasing behavior, where P_j is the price of product j . Then the law of zero surplus gives us the following constraints:

$$\begin{aligned}\Delta CS_{ij}(q_{ij}) &= CS_{ij}(q_{ij}) - CS_{ij}(q_{ij} - 1) \geq 0 \\ \Delta CS_{ij}(q_{ij} + 1) &= CS_{ij}(q_{ij} + 1) - CS_{ij}(q_{ij}) < 0\end{aligned}\tag{5.4}$$

In the spirit of collaborative filtering as described in Section 2.1.1, we model the personalized parameter in Eq.(5.1) as $a_{ij} = \alpha + \beta_i + \gamma_j + \vec{x}_i^T \vec{y}_j$, where \vec{x}_i is the K -dimensional consumer latent factor of consumer i , and \vec{y}_j is similarly the latent factor corresponding to good j . Hence, the shape parameters a_{ij} become intermediate parameters that can be derived from the actual parameters $\Theta = \{\alpha, \beta_i, \gamma_j, \vec{x}_i, \vec{y}_j\}$ in model optimization. Based on this, we minimize the following *Negative Log-Likelihood* (NLL) of the observed purchasing records:

$$\begin{aligned}& \underset{\Theta}{\text{minimize}} \quad -\log p(D) + \text{Regularization on } \Theta \\ &= -\sum_{i=1}^m \sum_{j \in B_i} \log (Pr(\Delta CS_{ij}(q_{ij}) \geq 0) Pr(\Delta CS_{ij}(q_{ij} + 1) < 0)) \\ &+ \lambda (\alpha^2 + \sum_{i=1}^m \beta_i^2 + \sum_{j=1}^n \gamma_j^2 + \sum_{i=1}^m \|\vec{x}_i\|_2^2 + \sum_{j=1}^n \|\vec{y}_j\|_2^2) \\ & \text{s.t. } \vec{x}_i, \vec{y}_j \geq 0, \forall 1 \leq i \leq m, 1 \leq j \leq n\end{aligned}\tag{5.5}$$

where B_i represents the products purchased by user i . Please note that in our current implementation, as we only consider the purchased products, but it would be interesting to include some non-purchased products in the objective function. We leave this as future work. The regularizer with coefficient $\lambda > 0$ is used to prevent model overfitting. We apply the commonly used non-negative constraints [91, 90] on the latent

factors $\{\vec{x}_i\}_{i=1}^m$ and $\{\vec{y}_j\}_{j=1}^n$, and adopt the sigmoid function to model the probabilities, which are

$$Pr(\Delta CS_{ij}(q_{ij}) \geq 0) = \frac{1}{1 + \exp(-\Delta CS_{ij}(q_{ij}))} \quad (5.6)$$

and

$$Pr(\Delta CS_{ij}(q_{ij} + 1) < 0) = 1 - Pr(\Delta CS_{ij}(q_{ij} + 1) \geq 0). \quad (5.7)$$

An optimal solution of Eq.(5.5) can be obtained by gradient descent, which involves the computation of the gradients on the shape parameter a_{ij} . To simplify the computation and make it possible for model estimation, we adopt the KPR utility function $U_{ij}(q) = a_{ij} \ln(1 + q)$. After the above model fitting process, we have the parameter estimates to compute the shape parameters \hat{a}_{ij} . These in turn give us the personalized utility functions $U_{ij}(q)$

$$U_{ij}(q) = \hat{a}_{ij} \ln(1 + q) = (\alpha + \beta_i + \gamma_j + \vec{x}_i^T \vec{y}_j) \ln(1 + q). \quad (5.8)$$

To obtain producer surplus we assume constant marginal cost of selling E-commerce goods. The (variable) cost function then is $C(q) = c_j q$, where c_j is the cost of a unit service of good j . Because $Q_{ij} \in \mathbb{N}$, we assume that the elements Q_{ij} in the allocation matrix Q of Eq.(5.3) follow a Poisson distribution, i.e.,

$$p(Q_{ij} = q) = \lambda_{ij}^q e^{-\lambda_{ij}} / q! \quad (5.9)$$

where the λ_{ij} 's are the distribution parameters. Finally, the framework for OSA based on total surplus maximization in Eq.(5.3) can be specified to the following optimization

problem:

$$\begin{aligned} \text{minimize}_{\Theta'} - \sum_{i=1}^M \sum_{j \in B_i} \sum_{q=0}^{\infty} \frac{\lambda_{ij}^q e^{-\lambda_{ij}}}{q!} (\hat{a}_{ij} \ln(1+q) - c_j q) \\ + \eta \sum_{i=1}^M \sum_{j \in B_i} (\lambda_{ij} - q_{ij})^2 \end{aligned} \quad (5.10)$$

where $\Lambda = [\lambda_{ij}]_{m \times n}$ are the distribution parameters of the Poisson distribution for the quantity matrix Q , $\eta > 0$ is regularizer coefficient, B_i is the set of products purchased by user i , and q_{ij} is the actual purchasing quantity. The quantity constraints are left out because $M_j = \infty$. As $(\hat{a}_{ij} \ln(1+q) - c_j q)$ is nonlinear function w.r.t q , the summation over q in Equation 5.10 cannot be eliminated. Please note that Λ should be viewed as intermediate parameters, as in fact they are resulted by applying *Collaborative Filtering* (CF) approach defined in Equation 2.6 to user i and product j , that is,

$$\lambda_{ij} = \bar{x}_i'^T \bar{y}_j' + \alpha'_i + \beta'_j + \gamma' \quad (5.11)$$

where $\bar{x}_i', \bar{y}_j' \ j \in \mathbb{R}^D, \alpha'_i, \beta'_j, \gamma' \in \mathbb{R}$. So the model parameters $\Theta' = \{\bar{x}_i', \bar{y}_j', \alpha'_i, \beta'_j, \gamma'\}$. In practice, it is sufficient for us to sum only the first few terms for evaluating the expectation w.r.t Poisson distribution when λ_{ij} is small. For our E-commerce experiment, as the purchase quantities q_{ij} are small numbers, summing q from 0 to 10 gives us good approximation of the true value.

The minimization of both Eq.(5.5) and Eq.(5.10) can be conducted based on gradient descent. Once the distribution parameters Λ are obtained from Eq.(5.10), we have the expected allocation matrix \bar{Q} as:

$$\bar{Q}_{ij} = \sum_{q=0}^{\infty} q \cdot \frac{\lambda_{ij}^q e^{-\lambda_{ij}}}{q!} = \sum_{q=0}^{\infty} \frac{\lambda_{ij}^q e^{-\lambda_{ij}}}{(q-1)!} = \lambda_{ij} \quad (5.12)$$

which we take for service allocation and product recommendation. Note that in the regularizer of Eq.(5.10), λ_{ij} is actually the expectation of quantity Q_{ij} according to the nature of Poisson distribution (Eq.(5.12)). As a result, the regularization component applies a guidance to the learning process, so that the estimated allocation quantities for those observed transections in the training dataset would be close to their actual values.

5.2.2 Online Peer-to-Peer Lending

In P2P lending services like Prosper, the borrowers are loan request producers, since the loan requests can be viewed as financial products. The lenders are consumers of these financial products. Here the OSA problem is how the lenders (i.e., consumers) should distribute their assets among the loan requests (i.e., determining the allocation matrix Q), so that total surplus in the system is maximized.

In a standard online lending process, the borrower (request producer) k initiates a loan request j by specifying two features: the size of the loan M_j , and its maximal interest rate r_j^{max} that she is willing to offer. Once a request is generated, the lenders (request consumers) i bid the request by providing the amount of money they would like to lend and the interest rates they ask for, which should be lower than or equal to r_j^{max} . When the total amount of money in bid exceeds the request in a given time period, the loan request then makes a deal, and the top bidders (those with the lowest interest rates) whose money amounts to the request win the bid. The highest interest rate among the winners is set as the final interest rate r_j for the loan j .

The consumer surplus for the lenders is the interest they obtain from this loan $r_j Q_{ij}$, less the opportunity cost $\hat{r} Q_{ij}$ of investing the money in other ways. In practice, the opportunity cost should be individual specific and nontrivial to track. In this work, we simply treat the opportunity cost as some risk-free interest investment. We understand this assumption is rather restrictive and is hardly viable in practice. In Section 5.5, we will discuss more feasible alternative ways of opportunity cost modeling. Even still, please be aware that as P2P lending is essentially an investment problem from the lender's perspective, it requires a plethora of investment theory and statistical modeling. We are interested to delve deeper in this direction in the future. As we view the opportunity cost of all people as saving money in bank, the *Consumer Surplus* (CS) is,

$$CS_{ij}(Q_{ij}) = (r_j - \hat{r})Q_{ij} \quad (5.13)$$

where \hat{r} is the saving interest. Similarly, the *producer surplus* (PS) for the borrowers is the interest they would be willing to pay $r_j^{max} Q_{ij}$, less the actual interest they have to pay $r_j Q_{ij}$, namely,

$$PS_{ij}(Q_{ij}) = (r_j^{max} - r_j)Q_{ij} \quad (5.14)$$

Thus the total surplus is:

$$TS_{ij}(Q_{ij}) = CS_{ij}(Q_{ij}) + PS_{ij}(Q_{ij}) = (r_j^{max} - \hat{r})Q_{ij} \quad (5.15)$$

Because Q_{ij} represents a quantity of money that is a non-negative continuous variable, we can apply a probability distribution defined on non-negative random variable. For example, a log-normal distribution could be adopted for Q_{ij} , namely,

$\log(Q_{ij}) \sim \mathcal{N}(\mu_{ij}, \sigma_{ij})$. However, as Q_{ij} in Equation 5.15 is linear w.r.t $p(Q_{ij})$, regardless of whatever probability distribution used for Q_{ij} , the expected total surplus maximization in Equation 5.3 can be simplified as,

$$\begin{aligned} & \underset{U}{\text{maximize}} \sum_i \sum_j \mu_{ij} (r_j^{max} - \hat{r}) \\ & \text{s.t. } \mathbf{1}^T U \leq M, \mu_{ij} \in \mathbb{R}_+ \end{aligned} \tag{5.16}$$

where U are the model parameters, μ_{ij} is the mean of the probability distribution $p(Q_{ij})$ and M_j is the total money lender j holds. The optimal solution of Equation 5.16 can be found by linear programming. Finally, we take the expected quantity under Gaussian distribution as the allocation matrix, i.e.,

$$\bar{Q}_{ij} = \mu_{ij} \tag{5.17}$$

This result is interesting in that it allows us to allocate the investments in a greedy manner according to the per capita surplus $(r_j^{max} - \hat{r})$ of each loan request, which is an intuitional rule for investment in practice and easily applicable in real-world systems.

5.2.3 Online Freelancing Platforms

In online freelancing networks like Mturk and Upwork, the employer (job producer) k posts job j online, and the freelancers (job consumers) i apply for the jobs that they are willing to take. Because a job can only be assigned to a single freelancer and a freelancer can only decide to take a job or not, rather than take part of a job, the elements Q_{ij} in allocation matrix Q can only be binary values in $\{0, 1\}$.

The employer and freelancer negotiate to decide the salary s_j for job j . After the job is accomplished, they rate each other, indicating their satisfaction about the other side. We denote the rating given by freelancer i and employer k about the job j as r_{ij} and r_{kj} , respectively, which are integers in a specific rating scale.

To estimate the consumer and producer surplus experienced on a given job, we adopt the economic assumption that the percentage surplus against the price that the consumer pays or the producer obtains is proportional to the normalized ratings that they cast on each other [54, 187], i.e., a higher rating implies a higher percentage surplus.

To do so, we predict the freelancer-job ratings \hat{r}_{ij} and employer-job ratings \hat{r}_{kj} , respectively, based on the Collaborative Filtering (CF) approach of Eq.(2.4) introduced in section 2.1.1. By the sigmoid function $h(x) = \frac{2}{1+\exp(-x)} - 1$, we further model the percentage surplus for freelancers as:

$$\frac{U_{ij}(Q_{ij}) - s_j}{s_j} = h(\hat{r}_{ij})Q_{ij} = \left(\frac{2}{1 + e^{-\hat{r}_{ij}}} - 1 \right) Q_{ij} \quad (5.18)$$

and the percentage producer surplus as:

$$\frac{s_j - C_j(Q_{ij})}{s_j} = h(\hat{r}_{kj})Q_{ij} = \left(\frac{2}{1 + e^{-\hat{r}_{kj}}} - 1 \right) Q_{ij} \quad (5.19)$$

where $Q_{ij} \in \{0, 1\}$ can be viewed as a binary indicator that whether or not a job is assigned, so that a surplus can be obtained for consumers and producers in Eq.(5.18) and (5.19).

As a result, the consumer, producer, and total surpluses implied in a specific

job assignment i to j are:

$$\begin{aligned}
CS_{ij}(Q_{ij}) &= U_{ij}(Q_{ij}) - s_j = h(\hat{r}_{ij})s_jQ_{ij} \\
PS_{ij}(Q_{ij}) &= s_j - C_j(Q_{ij}) = h(\hat{r}_{kj})s_jQ_{ij} \\
TS_{ij}(Q_{ij}) &= (h(\hat{r}_{ij}) + h(\hat{r}_{kj}))s_jQ_{ij}
\end{aligned} \tag{5.20}$$

On considering that Q_{ij} is binary valued, we apply a Bernoulli distribution to model its probabilistic nature, i.e.:

$$p(Q_{ij} = 1) = \alpha_{ij}, P(Q_{ij} = 0) = 1 - \alpha_{ij} \tag{5.21}$$

where $0 \leq \alpha_{ij} \leq 1$. Let $A = [\alpha_{ij}]_{m \times n}$ be the parameter set, and let $M_j = \sum_i \alpha_{ij} = 1$ because each individual job is by nature provided only once. The OSA problem for online freelancing services is thus specified as:

$$\begin{aligned}
&\underset{A}{\text{maximize}} \sum_i \sum_j (h(\hat{r}_{ij}) + h(\hat{r}_{kj}))s_j\alpha_{ij} \\
&\text{s.t. } \mathbf{1}^T A \leq \mathbf{1}, 0 \leq \alpha_{ij} \leq 1
\end{aligned} \tag{5.22}$$

The solution of Equation (5.22) can be obtained by assigning job j to freelancer that gives the most total surplus, namely,

$$A_{ij} = \begin{cases} 1, & \text{if } i = \max\{h(\hat{r}_{i'j}) + h(\hat{r}_{kj})\}_{i'=1}^m \\ 0, & \text{otherwise} \end{cases} \tag{5.23}$$

Equation 5.23 is actually a specification of the direct non-probabilistic framework in Eq.(5.2). Furthermore, this can be viewed as a surplus-augmented version of the traditional CF-based personalized recommendation algorithms, which will be discussed in the following together with the previous specifications.

5.2.4 Remarks

It is worthwhile to compare and contrast our framework with some traditional recommendation algorithms.

In the case of unlimited quantity where $M_j = \infty$, the quantity constraint $\mathbf{1}^T \int Qp(Q)dQ \leq M$ in Eq.(5.3) can be removed and we obtain an unconstrained optimization function, just as shown in Eq.(5.10). In this case, the total surplus related to each consumer is independent from those of the others, and the optimal allocations for each consumer is independently isolated from each other. In the E-commerce application for example, the allocation for a given consumer i can be obtained with the following equation:

$$\begin{aligned} \underset{\Theta'}{\text{minimize}} & - \underbrace{\sum_{i=1}^M \sum_{j \in B_i} \sum_{q=0}^{\infty} \frac{\lambda_{ij}^q e^{-\lambda_{ij}}}{q!} (\hat{a}_{ij} \ln(1+q) - c_j q)}_{\text{total surplus regularization}} \\ & + \eta \sum_{i=1}^M \sum_{j \in B_i} (\lambda_{ij} - q_{ij})^2 \end{aligned} \quad (5.24)$$

This is similar to traditional Personalized Recommender System (PRS) [132] algorithms, where we consider the preferences of each targeted user and aim to provide the most relevant recommendations. In fact, Equation 5.24 can be interpreted as *Collaborative Filtering* (CF) on purchasing quantity with total surplus term as regularization, as opposed to conventional CF where the regularization term is L2 norm as below,

$$\underset{\Theta'}{\text{minimize}} \sum_{i=1}^M \sum_{j \in B_i} (\lambda_{ij} - q_{ij})^2 + \underbrace{\eta \|\Theta'\|^2}_{\text{L2 regularization}} \quad (5.25)$$

As total surplus regularization considers personalized utility on products and it implicitly enforces constraints to the quantity assignment λ_{ij} through the law of zero surplus for the last unit, it is much better than generic $L2$ regularization term. This is the main reason why the TSM based method as Equation 5.24 outperforms CF baseline algorithm as Equation 5.25. We acknowledge that the TSM algorithm might no longer be superior if it is compared to stronger baseline algorithm, e.g. a baseline algorithm considering product features, temporal factors and product repurchase cycle. If this is the case, our TSM based method also needs to be improved. For example, one of the biggest improvements might be changing the modeling of personalized utility $U_{ij}(Q_{ij})$ to more flexible utility function. For example, we can adopt Cobb-Douglas function which allows different curve shapes for different products. Such change can more accurately model purchase quantity λ_{ij} than KRP utility function used in this chapter.

Similarly for online freelancing application denoted in Eq.(5.22), we see that for a given target job j , the employer-job rating $h(\hat{r}_{kj})$ (predicted by CF) and the hourly salary s_j would be known values. As a result, the greedy weight $(h(\hat{r}_{ij}) + h(\hat{r}_{kj})) s_j$ will only depend on the freelancer i . In this sense, we are actually assigning the job j to the freelancer i of the maximized $h(\hat{r}_{ij}) s_j$. This is actually a generalization of CF-based algorithms that recommend job j to the freelancer i of the maximum predicted rating \hat{r}_{ij} , where the only difference is that we further take the hourly salary s_j into consideration for a maximized total surplus that is measured on a basis of money.

Another interesting yet intuitive conclusion from the existence of a non-infinity solution to Eq.(5.24) is that, a larger quantity of products that the producers sell is not

necessarily preferred by the system, although we assume the quantities that producers can supply are unlimited. This results from the diminishing marginal utility experienced by consumers.

However, when the constraint on quantity exists, the consumer surpluses are correlated with each other, so that the allocation matrix that gains a globally maximized total surplus does not necessarily imply a maximized surplus for each consumer or producer.

5.3 Results

In this section, we take our framework to the data, and perform the traditional tasks of purchase prediction and personalized recommendation, as well as the new task of total surplus maximization. We first present a more detailed description of the E-commerce application, which we think is one of the most representative and easy-to-understand application scenarios that match the economic theories. Then we briefly sketch results on P2P lending and online freelancing applications, to illustrate the scope of our framework.

5.3.1 E-commerce Dataset Description

We adopt the consumer purchasing records dataset from Shop.com² for model evaluation, because an important information source leveraged in our framework is the quantity of product that a consumer purchases in each transaction, which is absent

²<http://www.shop.com>

in many of the public datasets. In the Shop.com dataset, however, we have both the product price information and the quantity that a consumer purchased in each record.

To avoid the problem of cold-start [95, 185], and to focus on our key research target of total surplus maximization, we select those consumers and products with at least five purchasing records, which is a frequently adopted pre-processing method in previous work [95, 93, 158]. Some statistics of our dataset are summarized in Table 5.2.

Table 5.2: Statistics of the Shop.com dataset.

#Consumers	#Products	#Purchasing Records	Density	Train/Test
34,099	42,691	400,215	0.03%	75%/25%

We see that the dataset is extremely sparse with a density of only 0.03%, which is similar to previously seen recommendation tasks. Furthermore, we randomly select 75% of the transactions from each consumer to construct the training set for model learning, and the rest 25% are used for testing. These amount to roughly 100,000 purchasing records by 34,000 consumers on 30,000 products in the testing dataset.

5.3.2 Parameter Selection and Estimation

The personalized KPR utility function $U_{ij}(q)$ indicated in Eq.(5.8) is parameterized solely by parameter a_{ij} , and the estimation of a_{ij} boils down to the inference of consumer and product biases and latent factors by optimizing Eq.(5.5).

In the estimates, we set the hyper-parameter K (i.e., the dimensions of \vec{x}_i and \vec{y}_j in Equation (5.5)), λ involved in Eq.(5.5) and η in Eq.(5.10) based on cross validation

Table 5.3: Evaluation on Conversion Rate (CR@N) and Total Surplus (TS@N) for Top-N recommendation, where TSM^* stands for our TSM approach with regularization coefficient $\eta = *$ in Eq.(5.10).

N	5				
Method	CF	$TSM^{0.1}$	TSM^1	TSM^5	TSM^{10}
CR (%)	0.10	0.10	0.10	0.30	0.30
TS (\$)	33.05	1009.45	1009.45	422.01	24.48
N	10				
Method	CF	$TSM^{0.1}$	TSM^1	TSM^5	TSM^{10}
CR (%)	0.10	0.10	0.10	0.30	0.30
TS (\$)	57.89	2278.36	2208.50	807.56	213.45
N	20				
Method	CF	$TSM^{0.1}$	TSM^1	TSM^5	TSM^{10}
CR (%)	0.20	0.30	0.40	0.60	0.50
TS (\$)	98.09	2892.03	3135.35	1137.89	676.65

on the training dataset, and they are usually set as $K = 20$, $\lambda = 0.05$ and $\eta = 5$, unless we tune these parameters to investigate their influences on model performance.

Once the estimated \hat{a}_{ij} 's are obtained with Eq.(5.5), we are able to evaluate the utility $U_{ij}(q)$ of an arbitrary consumer-product pair, which allows us to learn the average allocation quantities λ_{ij} in Eq.(5.10). Recognizing that λ_{ij} is consumer-product specific similar to a_{ij} , we once again parameterize it in a CF manner with $\lambda_{ij} = \alpha' + \beta'_i + \gamma'_j + \vec{x}'_i \vec{y}'_j$, and thus λ_{ij} can be estimated as a intermedia parameter by gradient descend on $\Theta' = \{\alpha', \beta'_i, \gamma'_j, \vec{x}'_i, \vec{y}'_j\}$.

For simplicity, we set the cost $c_j = 0.5P_j$ for all the products in the dataset, where P_j is the price of a product j . The cost ratio 0.5 is an average estimation based on the surveys of 100 producers from different product categories. In fact, we evaluated our framework using various costs and find that the performance of our framework is not sensitive to different cost ratios in a reasonable range.

Please note that when the regularization term η in Eq.(5.10) is set sufficiently large, the effect of total surplus component will vanish and the equation turns into a mere CF problem to predict q_{ij} , which serves a baseline algorithm in our later comparative study.

The procedure ends up with the estimated values of λ_{ij} for any given consumer-product pair in our dataset. As suggested by Eq.(5.12), a product recommendation list is thus provided to consumer i by ranking the products in descending order of λ_{ij} . For easy reference, the values of the involved hyper parameters are shown in Table 5.4.

Table 5.4: Summary of parameters. The number of latent factors K and the CF regularization coefficient λ are identified by cross validation, and are fixed throughout the reported results; η varies so as to examine its influence; c_j is the marginal cost of product j .

#Latent factors K	λ in Eq.(5.5)	η in Eq.(5.10)	c_j in Eq.(5.10)
20	0.05	5	$0.5P_j$

5.3.3 Purchase Prediction and Recommendation

We investigated the performance of our TSM framework for the task of personalized purchase prediction and recommendation. For performance comparison, we adopt the widely used CF algorithm in Eq.(2.4) and (2.6) to predict the purchasing quantities directly, which are integer values ranging from 1 to 20. For fair comparison, the hyper-parameters K and λ are set the same as those in Table 5.4.

Similar to our TSM framework, once the predicted quantities are obtained, we construct the top- N recommendation list for a consumer from the testing set in descending order of the quantities. We adopt the measure Conversion-Rate@ N (CR@ N) for performance evaluation on top- N recommendation, which is a typical metric widely adopted in real-world E-commerce systems [33].

For a given number of testing consumers and the recommendation lists of length N for each of them, CR@ N is the percentage of lists that ‘hit’ the purchase records in testing set for the target consumer. In our exercise, N runs from 1 up to 100. For each consumer in the testing set, there are as many as 30,000 candidate products for recommendation, and all the candidate products are present in the training dataset.

For computational efficiency in evaluation, we randomly select 1000 users to evaluate average CR at each time, and the CR performance of 30 testing rounds are averaged to provide the final evaluation results.

We use *Collaborative Filtering* (CF), which is described in Equation 5.25, as the baseline algorithm. Please refer to Section 5.2.4 for the discussion about the difference and connection between TSM framework and the baseline algorithm. The results for CF and our TSM framework with different choices of regularization coefficient $\eta =$

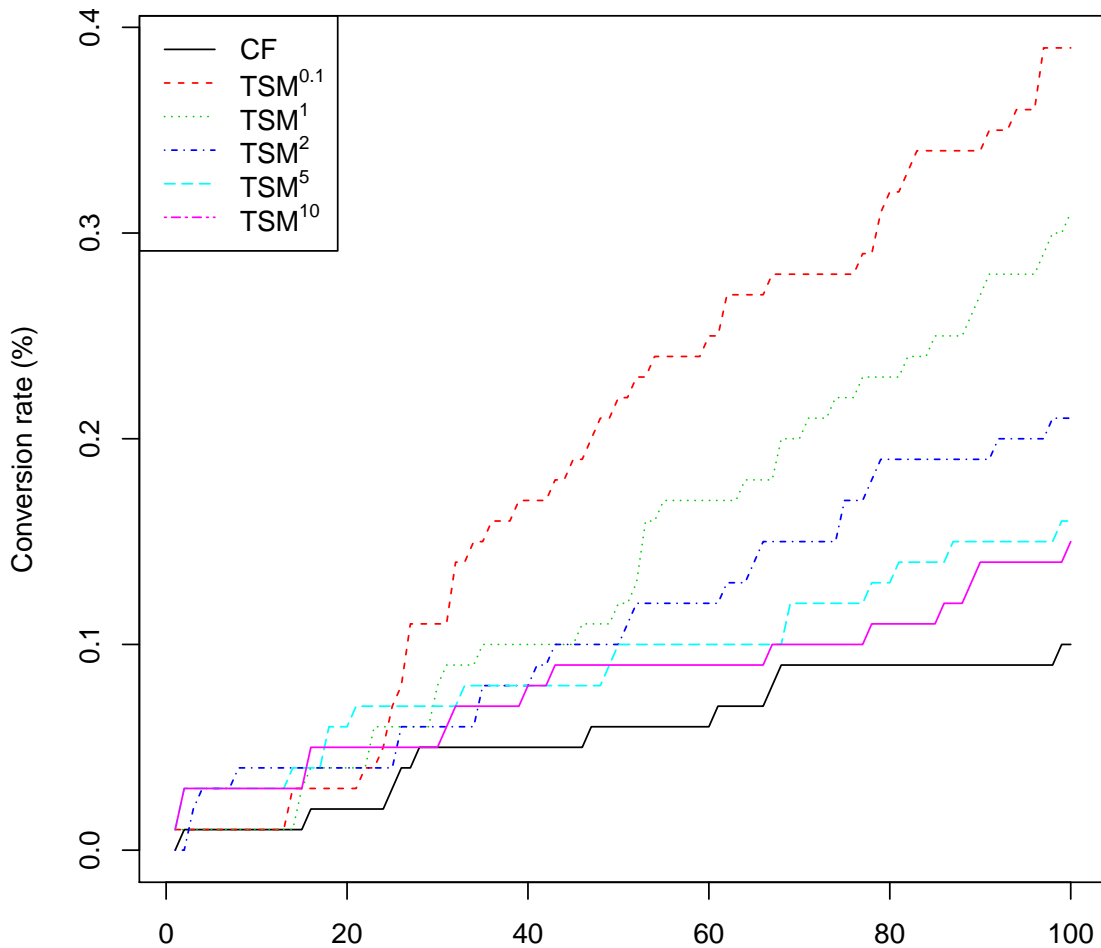


Figure 5.2: Comparison of the recommendation performance for CF and TSM ^{η} . The y-axis is the conversion rate, and the x-axis is the length N of each recommendation list.

0.1, 1, 5, 10 with recommendation length $N = 5, 10, 20$ are presented in Table 5.3. More complete results for N from 1 to 100 can be seen in Figure 5.2. The bolded improvements in Table 5.3 are significant at a 0.05 level.

The results show that our TSM framework outperforms CF for most choices of regularization coefficient η and recommendation length N . An interesting observation is that the performance of TSM $^\eta$ generally degrades with the increase of η on relatively long recommendation lists, all the way towards the performance given by the baseline algorithm of *Collaborative Filtering* (CF). This is actually reasonable as stated before, because when $\eta \rightarrow \infty$, TSM literally degenerates to CF and its performance will also converge to CF. This observation further emphasizes the importance of our surplus maximization component, and it suggests that maximizing with total surplus could be beneficial to the consumer experience on personalized recommendations.

Besides, the results also suggest that the choice of η should not be too small either, which would dismiss the quantity guidance of the observed purchases, especially for top precisions in shorter recommendation lists. One possible reason can be that without the quantity guidance, λ_{ij} would mostly depend on the personalized KPR utility U_{ij} . As KPR utility function is rather limited in terms of shape flexibility, it could fail to describe the actual consumer utility for some products. We in fact confirmed this by predicting the purchase quantities using the constraints in Eq.(5.4) directly, and the predictions turned out rather inaccurate with larger *root mean squared error* (RMSE) than that by CF. In summary, η influences the performance by balancing the importance between total surplus and quantity guidance, and it should be properly

selected in practical applications.

5.3.4 Evaluation on Total Surplus

In this section, we closely examine the performance of our framework under the *total surplus* metric, which is a core notion of this work. The evaluation is carried out based on the recommendation results from the above section. Similar to the Top-N conversion rate, we are interested in calculating the *accumulated total surplus* of a Top-N recommendation list for each user, which is defined as,

$$TS@N = \frac{1}{M} \sum_{i=1}^M \sum_{j \in \Pi_{i,N}} (\hat{a}_{ij} \ln(1 + \lambda_{ij}) - c_j \lambda_{ij}) \quad (5.26)$$

where i and M are the index and the total number of testing consumers, respectively, N is the length of recommendation list, and $\Pi_{i,N}$ is the length- N personalized recommendation list for the i -th consumer.

Similarly, the results of $TS@N$ are reported in Table 5.3, and a full scope report under comprehensive choices of N can be found in Figure 5.3.

It can be seen from the results that our TSM approach consistently outperforms the CF method. This result is actually not surprising because our TSM framework is by nature able to maximize the total surplus by Eq.(5.10). Besides, we find that the smaller η is, the more total surplus our TSM approach gains. This observation on the influence of η further verifies the effects of the surplus maximization component and the quantity guidance in Eq.(5.10).

More interestingly, when combining this result with that on recommendation

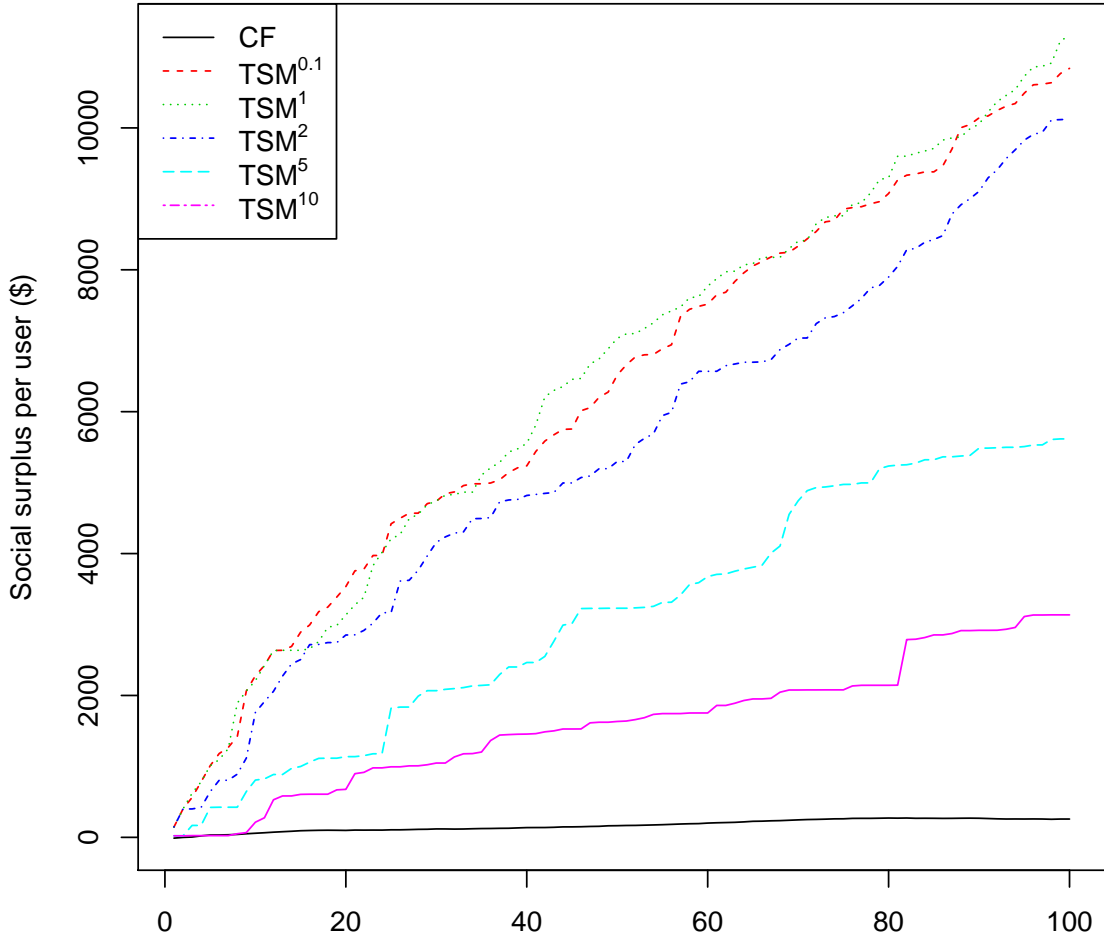


Figure 5.3: Comparison on Total Surplus (TS) for CF and TSM ^{η} . Note that TS@N means the total social surplus of the top N recommendations when they are all accepted by the corresponding consumer.

in the previous section, we find that our TSM framework can achieve decent results in terms of both total surplus and conversion rate when η is properly set. This is exciting because our framework is able to benefit the social good on total surplus, and at the same time improves the consumer experience in personalized recommendations.

5.3.5 P2P Lending Networks

To investigate the performance on Peer-to-Peer loan networks, we use the datasets from the well-known P2P lending website Prosper³[32]. Beginning in the third quarter in 2009, Prosper introduced an automatic bidding mechanism that bids the listings (i.e., loan requests) on behalf of the lenders automatically once a listing is created. However, as we intend to investigate the behavior of consumers and producers in an economic system, we prefer the decisions be made directly by the consumers, instead of indirectly by the algorithms. As a result, we adopt those listing and bidding records before this mechanism is launched, which covers the period from November 9th 2005 to May 8th 2009.

As we do not consider risk control in our current model, we select those successfully funded listings whose status are not *Defaulted*, *Cancelled* or *Charge-off* from the dataset, because these listings are meant to be ruled out from the system by the intelligent risk control mechanisms. Finally, our dataset involves those funded listings of the status *Current*, *Late*, *Payoff in Progress*, or *Paid*, which correspond to 46,680 listings, 1,814,503 bids, and a total amount of \$157,845,684 fundings. Some statistics of these records are summarized in Table 5.5.

To calculate the total surplus reached by an arbitrary allocation $Q = [Q_{ij}]_{m \times n}$, we take the yearly average bank deposit interest rate $\hat{r} = 0.01$ as the risk-free interest

³<http://www.prosper.com>

Table 5.5: Statistics of the selected Prosper dataset, where ‘rate’ represents the interest rate of a loan.

#Listings	#Lenders	#Bids	TotalAmount
46,680	49,631	1,814,503	\$157,845,684
MinimumRate	MaximumRate	AverageRate	Amount/Listing
0.0001	0.4975	0.1662	\$3,381.44

rate, and the TS for P2P lending can be calculated as:

$$TS_{P2P} = \sum_i \sum_j Q_{ij}(r_j^{max} - \hat{r}) \quad (5.27)$$

Based on this, the results on total surplus for the actual allocations (Actual)

and our Total Surplus Maximization (TSM) framework are shown as follows:

Table 5.6: Results on total surplus with and without our Total Surplus Maximization (TSM) framework.

	TS(\$)	TS/Listing(\$)	TS/capita(\$)
Actual	25,174,131	539.29	0.1595
TSM	33,838,364	724.90	0.2144

The estimates indicate that the TSM framework achieves 34.42% higher total and per listing/capita surplus, from \$0.16 per capita to \$0.21 per capita, which is a major improvement in efficiency for the online lending systems. Based on two-tailed t -test on the large amount of listings, the improvements are significant at a 0.01 level.

Please note that the TSM algorithm would not have improved over actual total

surplus if all the lists are fully funded. This can be easily recognized by reorganizing the right hand side of Equation 5.27 as,

$$TS_{P2P} = \sum_j M'_j (r_j^{max} - \hat{r}) \quad (5.28)$$

where $M'_j = \sum_i Q_{ij}$ is the amount of money invested to list j . Clearly, given any fixed amount of money, it should be invested to the most profitable list, namely, the list with the highest $r_j^{max} - \hat{r}$, and the remaining money goes to the second most profitable list, third most profitable list... This process continues until all money is invested. If all lists are fully funded, the total surplus by Equation 5.27 is the same for any assignment of Q . For the data used in the experiment, we verified that some of the lists indeed are not fully invested. We must admit that our current TSM based approach for P2P problem is very limited and can hardly be adopted in practice. The purpose of this algorithm is just to illustrate how TSM might be applied to P2P problem.

The improvement on total surplus is not surprising because our framework intends to achieve a maximized surplus among all the possible allocations. However, we should further verify that our allocations are acceptable to the lenders in practice. As a result, we calculate the Percentage of Paid (PoP) listings among all the funded listings in our dataset, which indicates the safety factor of a funding allocation.

Results show that the PoP among all the listings in our selected dataset is 69.37%, while the PoP among the funded listings of our TSM allocation is 73.32%, which is no lower than the actual PoP. This suggests that our TSM framework is able to improve efficiency without impairing the safety of the system.

5.3.6 Online Freelancing

We used the dataset from Zhubajie⁴(ZBJ) for empirical verification of online freelancing applications. ZBJ is a well-known Chinese online marketplace website that includes online jobs across various categories. Each employment record includes the employer, freelancer, and job IDs, hourly salary, and employer-job and freelancer-job ratings, which are integers ranging from 0 to 5. Some of the basic statistics of the dataset we collected are summarized in Table 5.7.

Table 5.7: Some key statistics of the ZBJ dataset.

#Employers	#Freelancers	#Jobs	AverageSalary
40,228	46,856	296,453	¥21.68/hr
#Employer Ratings	#Freelancer Ratings	Average Emp- loyer Rating	Average Free- lancer Rating
276,103	241,638	2.336	2.405

Similar to our E-commerce application, we make job recommendations to freelancers based on the allocation matrix produced by our framework, then verify the performance on this task. To do so, we take all the freelancer-job ratings, and conduct personalized recommendations based on Collaborative Filtering (CF). In CF, a job j is assigned to freelancer i who has the highest predicted rating \hat{r}_{ij} , while in our Total Surplus Maximization (TSM) framework, it is assigned to the freelancer where $Q_{ij} = 1$

⁴<http://www.zbj.com>

according to Eq.(5.23).

We conduct five-fold cross-validation for both methods, and we still adopt the Conversion Rate (CR) for performance evaluation, which is the percentage of properly assigned jobs in the testing dataset. Results of TSM and CF methods are presented in Table 5.8, under different choices of the number of latent factors K used for rating prediction (see Eq.(2.4)).

Table 5.8: Conversion rate on job recommendation.

K	5	10	20	30	40	50
CF(%)	0.165	0.216	0.244	0.258	0.262	0.266
TSM(%)	0.384	0.421	0.453	0.486	0.507	0.512

Results show that our TSM framework gains consistently better performance on conversion rate for job recommendation. The improvements are significant at 0.01 level for all choices of latent factors K . According to the discussions in Section 5.2.4, the improvement comes from the inherent consideration of salary rate in our model, which implies that the salary could be an extremely important factor when freelancers seek for jobs. Besides, we see that the results tend to be stable when $K \geq 40$ for both methods, which means that a dimensionality of 40 could be sufficiently enough to describe the factors considered by freelancers.

We further calculate the total surplus for the allocations given by CF and TSM under different choices of K 's. Once an arbitrary allocation $Q = [Q_{ij}]_{m \times n}$ is realized in

practice, we obtain the total surplus as:

$$TS_{Fr} = \sum_i \sum_j (h(\hat{r}_{ij}) + h(\hat{r}_{kj})) s_j Q_{ij} \quad (5.29)$$

We calculate the total surplus for each of the five testing folds, where there are 59,291 job allocations on average in each fold. Finally, the averaged total surplus among the five folds are shown in Table 5.9, where the surplus is measured in CNY (¥) and ‘m’ is for ‘million’.

Table 5.9: Total surplus of online freelancing job allocations under typical choices of latent factor K .

K	5	10	20	30	ActualAllocation
CF(¥)	1.562m	1.758m	1.824m	1.860m	2,593,618
TSM(¥)	3.235m	3.862m	4.270m	4.336m	

The improvements on total surplus are significant at 0.001 level for all choices of K . We see that our TSM framework consistently gains more surplus than CF. It even leads to more surplus than the actual surplus of the testing dataset. The TSM framework gains a total surplus of ¥73.13/job on a job-level when $K = 30$, while that for the CF approach and the actual allocation are ¥31.37/job and ¥43.74/job, respectively.

The fact that the total surplus of the actual allocation is less than that gained by our TSM approach implies the failure of market equilibrium, which is frequently observed by economists in the research of antitrust and market regulations. For online freelancing as an example, this comes from the problem of information asymmetry between freelancers and employers, because it is impossible for freelancers to browse

millions of jobs to make a final decision. This further stresses the importance of personalized recommendation techniques in service allocation, which help to push appropriate jobs to freelancers, to overcome the problem of information overload.

When putting the evaluation results on total surplus and recommendation together, we find it extremely exciting because our TSM framework leads to better market efficiency than even the practical market of the system, while at the same time benefiting the freelancers with more acceptable job recommendations. This means that our allocation solution may well be applied in practice for a better online markets compared with currently adopted recommendation techniques.

5.4 Related Work

In mainstream economics, economic surplus [56, 45, 16], also known as total welfare or Marshallian surplus [107] (named after Alfred Marshall), refers to three closely related quantities: consumer surplus, producer surplus, and social/total surplus, where social surplus is the sum of surpluses experienced by both consumers and producers. The research of surplus has had quite a long history in the progress of economical theories, dating back to as early as the 19th century with the initial understandings of Surplus Values [108, 109], when the gigantic increase in wealth and population brought by the First and Second Industrial Revolution drove economists to investigate the nature of economical increase [137].

In modern economics, the concept of social surplus has been widely adopted

by economists for economic system analysis and mechanism design, usually as a direct measure of *social good* to benefit the good of our human society [56, 111, 20]. However, although the Web has formed itself as a virtual society by continuously integrating the human activities from offline to online, the research community has seldom investigated the surplus nature of the Web as a social system.

In fact, a large number of Web-based services can be formalized as consumer-producer interaction systems, including the most commonly used E-commerce websites [104, 144], online financing [102, 32], crowd-sourcing systems [49, 26], and even social networks [135, 74], where consumers consume normal goods, financial products, free-lancing jobs, or information from corresponding producers.

These applications raise the practical problem of matching services from producers to consumers. Perhaps the most closely related tasks for such matching processes are Personalized Recommendation [53, 132, 81] and Search [23, 105, 61], which feed the implicit or explicit needs of the users with recommendations and search results.

However, current approaches for such tasks mainly focus on the benefits of one side without explicitly modeling the benefits of the Web system as a whole. For example, the widely adopted Collaborative Filtering (CF) [53, 83, 159, 143, 130, 141, 182] techniques for personalized recommendation inherently focus on the maximization of consumer satisfaction based on consumer preferences. Although the satisfaction of consumers intuitively benefits the surplus of producers by improving the potential of user clicks, there is no direct guarantee that such a single-side oriented modeling can benefit both sides.

In this work, however, we view the Web as a virtual society and propose to maximize the social surplus directly, based on well-developed and widely-accepted economic concepts and conclusions, which, to the best of our knowledge, is the first time to do so in the context of web-based applications.

5.5 Discussion

The P2P lending modeling is based on two assumptions: 1) the borrower and lenders arrive at the same time and the match or money allocation is performed between all borrowers and lenders. 2) the opportunity cost is a flat risk-free interest rate for all lenders. In the following, we will briefly discuss some ideas on relaxing such assumptions so that our method can be applied to real world problem.

5.5.1 Temporal Factor

Before the discussion, we would like to recap the process of the P2P lending problem. In a typical P2P lending network such as Prosper, a loan list gets funded through the following steps,

1. The borrower posts a loan request, specifying the amount, the maximum acceptable interest rate and the expiration date.
2. Lenders who are interested in the loan start to bid on the loan. The bid includes the amount of money the lender want to lend and the acceptable interest rate (lower than the borrower's interest rate).

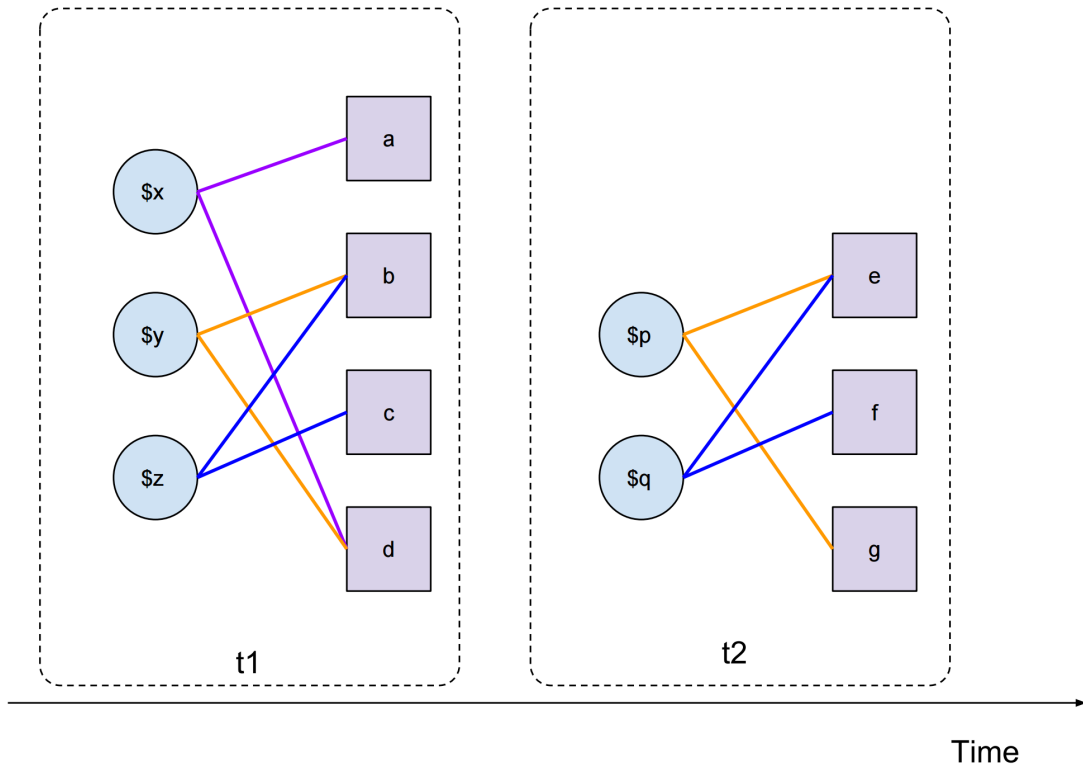


Figure 5.4: P2P lending problem considering temporal factor. In the graph, t_1 and t_2 denote two different time points. At each time point, a set of active loan requests and their potential lenders are identified. The relationship between loans and lenders can be described by a bipartite graph in which the nodes are loans and lenders and an edge is placed between a loan and lender if the lender is likely to invest the loan.

3. The bidding continues until the loan is fully funded by the bids or time expires for funding the loan. The interest rate of the loan is set as the highest of interest rates of the bids⁵.

The above process implies several important facts. First, loan requests are posted at different time and the lenders join the system at different time, meaning the loans are seen by different groups of lenders. Second, after seeing a loan request, lenders who are

⁵<http://www.lendacademy.com/comparing-returns-on-prosper-before-and-after-the-auction-model-ended/>

interested in the loan bid at different time, meaning their bidding amount and interest rate is affected by the existing biddings. Third, at any moment, a lender usually faces a set of loan requests and need to decide allocating his or her money among the requests. Considering these facts, we need to modify our current model in the following ways,

- Given a time point, identify the active loan requests and for each loan request, find a set of potential lenders. Active loan requests refer to the unexpired and partially funded requests. Identifying potential lenders for a given loan request can be accomplished by using standard recommendation algorithm such as collaborative filtering. We ended up with a set of active loans and a set of lenders. As shown in Figure 5.4, the link between a loan and a lender means the lender might invest the loan. Having the links is not enough. Our objective is to figure out the amount of money is attached to each link/investment so that the total surplus is maximized. It is worth noting that compared to our current treatment of P2P problem which involve all loans and lenders, the TSM at a particular time point such as t_1 involves only a few number of loans and lenders and it can be solved efficiently.
- When bidding a request, the lender needs to decide the interest rate strategically. The decision depends on several factors. The first important factor is lender's opportunity cost. We will discuss more on this factor in the next section. Beside opportunity cost, another important factor is estimating the interest rate other lenders will bid – clearly, the lender can win the bidding by set a very low interest rate, but the gain is much less; or the lender can gain more by specifying a higher

rate, but the chance of winning the bid will be smaller. An interesting research question would be what is the lender's optimal bidding strategy and how such bidding/auction process is related to other well-known action mechanisms such as second-price auction.

5.5.2 Opportunity Cost

We assume that all lenders have the same opportunity cost as risk-free bank saving. This assumption is quite restricted and should be relaxed for real world application. In this section, we discuss possible alternatives for opportunity cost modeling. From lender's perspective, choosing which list to lend is essentially an investment problem. When it comes to investment, there are two important factors: *risk* and *return*. The investor, lender here, is characterized by *risk aversion*. The objective of investment is about how to achieve good return while keeping the risk minimum. This objective is usually achieved by diversifying an investment portfolio[62]. In the P2P lending problem, each lender has a certain amount of money and he or she can split the money into multiple loan listings, and for each listing, the risk and return needs to be accessed. Though a full integration of investment theory will benefit our modeling, we choose to focus on obtaining a better way than risk-free bank saving to model opportunity cost at individual level.

Of course, the opportunity cost differs lender by lender. The difference is attributed to factors such as experience and risk aversion. For example, more experienced lenders are more likely to gain more than less experienced lenders; lenders who are less

risk sensitive tend to invest high return listings and gain more than risk sensitive lenders who prefer low return and low risk listings. If we explicitly express all these factors in the model, it is quite a challenge. Instead, we take a model-less approach – deriving each lender’s opportunity cost from the lender’s investment history. There are a wide range of ways to do this, from simple averaging to more sophisticated machine learning techniques. As an elaborate modeling of opportunity cost might deserve a full scoped paper, we here just study the simplistic modeling – averaging the lender’s historical returns for each lender.

We inherit the same symbols used in Section 5.2.2. For each lender i , the corresponding opportunity cost at time t is,

$$\hat{r}_{it} = \frac{\sum_{t'=t-W}^{t-1} L'_{it'}}{\sum_{t'=t-W}^{t-1} L_{it'}} - 1 \quad (5.30)$$

where t denotes t -th lending event for the lender, W is size of history window, $L_{it'}$ and $L'_{it'}$ are the lender investment and return at time t' , respectively. The purpose of W is to capture the situation where the lender’s opportunity cost changes overtime, e.g. getting more and more experienced through investments. W is a global parameter and can be decided by cross validation. Equation 5.30 predicts the lender’s opportunity cost (expected interested rate) as the average of previous investment return.

We are able to obtain each lender’s opportunity cost at time t . In order to perform the matching between lenders and listings, we need to predict the final interest rate r_j for listing j . In Section 5.2.2, we did not bother to calculate r_j explicitly, as it is safe to assume r_j is always above \hat{r} – bank saving. However, for personalized

opportunity cost \hat{r}_{it} , we should no longer adopt the same assumption. We propose the following matching criteria. Assume r_j is available, we consider the expected return of loan j as,

$$\bar{r}_j = r_j \times \text{Probability of the borrower paying the loan} \quad (5.31)$$

Lender i is a matching candidate of loan j only when the expected return \bar{r}_j is no smaller than the lender i 's opportunity cost by Equation 5.30. It is essential to evaluate the probability of borrower paying the loan, or in other words, the riskiness of the borrower. Otherwise, if we simply assume the borrower will pay the loan, the expected return for the lender might be less than the promised return unless the lender is insured. To adopt the expected return based matching strategy, we need to model both the r_j and the borrower's riskiness and we leave this as our future work.

5.6 Summary

Most existing literature on recommender systems focuses on developing new algorithms for standard evaluation metric such as RMSE, conversion rate or click through rate. There is little research on some fundamental questions, such as what metrics should be used to evaluate recommender systems and to what extent do the metrics reflect the goals of users, producers, platform providers, and the overall Web economy.

This chapter is our first step towards finding principled answers to these questions based on established economic theory. Considering a recommender system as an information agent to support two-sided matching tasks, we introduce established eco-

conomic surplus theory into recommender systems and meld it with recent data-driven algorithmic approaches. Our proposed Total Surplus Maximization framework integrates the goals of users and suppliers, which can be a good metric to optimize for platform providers as it better reflects the overall economic value of the online system. We have illustrated how to realize this framework for different recommendation systems. The empirical results for several sets of industry data demonstrated the effectiveness of the proposed framework.

This paper focuses on the broadest metric of efficiency, or maximization of total surplus, and this inherent principle is not restricted to recommendation tasks that we primarily investigated, but applicable to the whole research effort of web intelligence for social good. In the future, we will also examine performance metrics about its two major components: producer surplus and consumer surplus. We can also try the ideas on new datasets, compare different functional form and specifications of utilities and profits. We will implement these ideas both in static (one-time) and dynamic (multi-period/session/page) recommendation or search settings, and evaluate with real users to see the short term and long term impact of the total surplus based framework.

Please note that maximizing total surplus is only one of the important problems economics concerns about. Total surplus maximization, or market efficiency, is always phrased as creating a bigger pie[56]. Another closely related and equally important problem is how to divide the pie. Given a fixed total surplus as Equation 2.18, the consumer and producer divide the total surplus by Equation 2.16 and 2.17. Price is the magic factor that determines how the pie is divided. It is our interest to explore how

to balance the consumer surplus and producer surplus in the future.

5.7 Acknowledgment

This chapter is based on the WWW16 conference paper[186]. The paper is the result of a joint effort of Yongfeng Zhang, Qi Zhao⁶, Yi Zhang and Daniel Friedman. The specific contribution of each co-author is as follows: Yongfeng Zhang initiated the paper, formulated the basics of the TSM framework and evaluated the framework on the Prosper and ZBJ datasets. He also wrote the majority of the paper, including formalization and the experiment sections he was in charge of. Qi is a major contributor to the theoretical formalization. He also implemented and evaluated the E-commerce model specification and wrote the corresponding sections. Yi offered a lot of useful suggestions on experimental design and evaluation. She also helped polish the writing and shared her thoughts on further improvement. Daniel examined the economic notions of the paper and ensured they were used properly. He proposed to integrate “the law of zero surplus for the last unit” as the constraints for the E-commerce model specification, which is particularly important for personalized utility estimation. He also contributed significantly to the writing.

⁶the first two authors contribute equally

Chapter 6

Conclusion and Future Work

In this chapter, we conclude this dissertation by recapping our contributions, revealing the limitations of our methods and discussing several directions for future work.

6.1 Contribution

Product recommendation has become increasingly important in order to meet consumers' need to find the right products from the massive amount of online product information. Recommender systems have attracted attention from both academia and industry and a number of algorithms have been developed in the past decades. Recommender systems have a wide range of applications, including movie, music, news and E-commerce recommendation. Existing recommendation algorithms tend to formalize the recommendation task as a rating prediction and learn individual characteristics by fitting a collaborative filtering-based model. However, we argue that existing rec-

ommendation algorithms are insufficient for product recommendation for three main reasons. First, price needs to be considered explicitly for product recommendation, as it is a key factor determining consumers' shopping decisions. However, as price has little importance for movie, news and music applications, it is usually ignored. Second, for product recommendation, existing evaluation metrics such as RMSE might not reflect how consumer or producer benefit from the recommendation. In economics, consumers and producers are motivated to maximize their own benefit and the benefit has concrete economic significance. It is worth considering economics based metrics for evaluation. Third, existing algorithms usually recommend based on optimizing one side's benefit. However, we argue that a recommender system should be viewed as an ecosystem. In order to keep the ecosystem operating healthily, it is necessary to take both sides' benefit into consideration. The main contributions of the dissertation are summarized in the following sections.

6.1.1 General contributions

The major contributions of this dissertation are introducing economics-based principles into product recommendation and building recommendation frameworks by combining machine learning and economics theory. The proposed work comprises three major components:

- **A framework emphasizing producer benefit.** In a market, as producers are profit driven, it is their nature to do their best to get the most profit from selling. In economics and marketing study, price discrimination, especially personalized

pricing, has been known to be an important marketing strategy for the sellers to capture more consumer surplus[163]. WTP elicitation is considered to be the key of implementing personalized pricing. As consumers are heterogeneous, it is a nontrivial task to elicit WTP on the individual basis. In this dissertation we are motivated to introduce personalized pricing to product recommendation by devising a WTP elicitation mechanism that is applicable to the E-commerce scale.

- **A framework emphasizing consumer benefit.** In economics, consumers are utility-maximization driven, that is, at any moment, they seek to purchase the set of products that give the most desirability given their limitations on their expenditure. Consumer choice theory has been well studied in economics literature, but to the best of our knowledge, it is relatively unstudied in the E-commerce setting. It is well known that products are related in certain way, and the inter-product relationship determines the utility of multiple products as a whole. In this dissertation, we are motivated to embed product recommendation in a consumer utility maximization framework. Our main contributions include a theoretical approach of deriving the utility function of two products and a data-driven approach to learn the representation of inter-product relationships.
- **A framework emphasizing platform benefit.** Recommender systems have long been used as a means of optimizing one side's benefit. In this dissertation, we propose to view recommender systems as an ecosystem and the objective of

recommendation is maximizing the total benefits of all parties through recommendation. We are mainly inspired by matching theory developed in economics [122, 44, 46]. Our contributions in this direction include introducing total surplus as an optimization objective to product recommendation and proposing a generic recommendation framework for several typical applications, including E-commerce, P2P lending and online freelancing.

6.1.2 Specific contributions

Each of the above components represents a high leveled view of our contributions to product recommendation. In particular, we develop specific techniques to achieve the objective defined by each component. We implement the respective techniques and demonstrate their effectiveness by evaluating them on real world datasets. We summarize the significance of the key techniques as below:

- We propose an incentive-compatible mechanism for eliciting WTP in the E-commerce setting. WTP elicitation is considered essential for dynamic pricing (or first degree price discrimination) in economics and a number of methods have been developed for this task. Though existing methods have been shown to be quite successful in some application scenarios, they face challenges when being applied to the E-commerce setting. In this dissertation, we propose a WTP elicitation mechanism for the E-commerce setting. The major contribution of our proposed mechanism is its applicability in E-commerce setting. In addition, our mechanism can complement existing WTP elicitation methods and could become a useful reference

for marketing practitioners.

- We propose to introduce personalized promotion to E-commerce recommender systems and implement a machine learning based method that can predict individual's WTP. Further-more, we propose a principled way to find the optimal price by solving Equation 3.9. We proposed to evaluate the performance of recommendation by producer's profit and demonstrate the effectiveness of personalized promotion.
- In Chapter 4, we propose a general method to derive the utility function for any given *marginal rate of substitution* (MRS). Our method is mathematically sound and generic. More importantly, it provides important guidance for us to find the desired utility function. For example, the well known *constant elasticity of substitution* (CES) utility function is a natural result of our method when an exponential function is chosen for MRS. It is possible for us to obtain other interesting utility function alternatives by examining different choices of MRS function.
- We propose to recommend by maximizing multi-product utility for consumers. We adopt multi-product utility for both model learning and evaluation. For a fair comparison, our approach is compared to other baseline recommendation algorithms under top-K precision and recall metric defined in Section 2.2.3. The empirical results on two real-world E-commerce datasets show that our approach outperforms baseline recommendation algorithms.

- In Equation 5.8, we show model personalized utility through the widely used collaborative filtering technique. We also explore specific formalization for E-commerce, P2P lending and freelancing marketing, as shown in Equation 5.10, Equation 5.16 and Equation 5.22. Though our formalizations sometimes make restrictive assumptions, they capture the most important characteristics of each application and can serve as good basis for future extension.
- We propose a Total Surplus Maximization (TSM) framework for recommendation. We evaluate our proposed framework in real datasets. The experimental results demonstrate that TSM-based recommendation performs better than standard recommendation techniques.

6.2 Limitations and Future Work

Although this dissertation demonstrates how economic principles can benefit product recommendation, we believe there is still a lot to do in order to make our methods perform better in practical settings. Here we focus on several major limitations of our methods and discuss how they could be resolved in future work.

6.2.1 On WTP elicitation

In Chapter 3, we propose to elicit consumers' WTP through BDM auction. Though we are able to collect meaningful data through educating users about the mechanism of BDM and the optimum strategy, direct application of BDM to real world setting

is still limited in several ways. First, BDM is not intuitive and might be hard to understand for a general audience without proper training. In our experiments, a significant number of users failed the quiz even after the training. Second, it is not clear how BDM will be accepted in real E-commerce websites. In our experiments, the users are from a crowdsourcing platform and are motivated to finish the experiment for the task bonus. Real E-commerce consumers might have very different motivations and objectives, and whether they will accept BDM is unknown. Third, in Chapter 3, we estimate personalized promotion and predict the seller's profit generated from personalized promotion, but our profit prediction is not derived from real recommendation results. In order to get more convincing results, it is necessary to add a recommendation step after the auction step. To demonstrate the effectiveness of personalize promotion, the evaluation is then about comparing the respective profit from recommendation with and without personalized promotion.

In future work, it would be interesting to explore other WTP elicitation methods that do not need consumers' explicit action such as participating in a BDM auction. Ideally, we expect the elicitation process to naturally fit into a typical E-commerce shopping process without intruding on the consumer's shopping experience. For example, we can present the items in an adaptive manner, that is, the system learns the consumer's preference based on consumer's previous feedback (e.g. click, browse) and adjusts the displayed items based on the learned results. Such adaptive process can help us to get a more and more refined profile about the consumer and hence achieve a more accurate estimation of that consumer's WTP.

Another interesting question is how the personalized pricing should be implemented. Showing different discount rates or prices of the same product to different people might cause problems, as personalized pricing has reportedly caused issues [125]. Thus, it is worth more in-depth discussion on how personalization should be implemented.

6.2.2 On multi-product utility modeling

In Chapter 4, we adopt CES utility function as a representation of two products' utility. CES has some nice properties. For example, it can easily degenerate into perfect substitutes, perfect complements and Cobb-Douglas by changing its elasticity of substitution (ES) variable. However, on the flip side, CES has its limitations - it is actually difficult to model perfect complements as it requires that the ES variable approach zero. Such a requirement cannot be achieved in practice. It is worth exploring other utility function alternatives. Finding a better utility function will contribute to both the recommendation community and economics community.

In Section 4.2.5, we would need to generate negative candidate sets in order to construct the RUM choice model. As the product space is usually very large, it is not possible (and probably not necessary) to consider all candidate sets given the budget constraint. In Chapter 4, we simply generate the negative samples by varying the quantity of products observed in the transaction. In the future, we might want to expand the sampling space by including other products which are relevant but not observed in the transaction. For example, in [114], the authors propose to classify

product relationship links using information from an Amazon.com dataset. It is possible for us to use their product relation graph as a guidance for negative candidate sets sampling.

Our multi-product utility modeling in Equation 4.8 is based on pairwise utility terms. This is also a limitation of our approach, as in practice it is common to have more than two products co-occurring frequently. For example, pumpkins, candy and costumes are purchased together at Halloween, so given pumpkins and candy, the next thing a consumer considers might be costumes. Our current multi-utility modeling, however, might not capture this, as it might end up recommending a cucumber or chocolate as these co-occur frequently with pumpkin and candy. Therefore, it is necessary to go beyond the pairwise utility basis.

Our proposed MPUM is based on an ordinal utility function. As we have seen in Section 2.3.1, ordinal utility is sufficient for representing consumer preferences. In Chapter 3, we show how personalized promotion at the single product basis can help sellers to gain higher profit. It would be useful to be able to estimate the cardinal utility of multiple products at the individual basis, as this will allow us to personalize price for a product bundle. In future work, we can transform ordinal utility to cardinal utility by integrating more constraints, e.g. *law of zero surplus for last unit* [54] which is used in Equation 5.4.

6.2.3 On personalized utility modeling and total surplus maximization

The E-commerce application of our TSM framework as described in Chapter 5 uses KPR utility function to represent how utility changes with the consumption quantity at the individual level. Though KPR has diminishing marginal utility property, its curve shape lacks flexibility and cannot provide a product specific diminishing rate. For example, the curve shapes for perishable products such as milk should be different from those durable products such as laptop computers. It is worth exploring other utility functions such as the Cobb-Douglas utility of the form $U(q_j) = a_{ij}q_j^{\gamma_j}$. If more flexible utility function is adopted, there are more model parameters to fit. This eventually requires us to integrate more information in order to get reliable estimation. For example, it is possible to incorporate product meta information, especially category information, to constrain the model parameters.

For the P2P application of TSM framework, we specify the lender's opportunity cost as a risk free investment such as bank savings. However, the estimation of a lender's opportunity cost could be much more complicated, depending on the lender's personal characteristics. For example, experienced lenders are more likely to gain more profit from investing than less experienced lenders, so they have higher opportunity costs. Another important aspect of investing is risk, which is not considered in our model. Risk aversion is also a personal characteristics and should be accounted for in the investment opportunity evaluation. For example, some lenders might be more aggressive and it is better to match them to high return and high risk loans. Generally, considering both

risk and return jointly is an interesting topic in the future work of TSM.

In our TSM framework, the product price is not manipulated. As we learn from our personalized promotion work in Chapter 3, price is a major influence on consumer’s purchase decision, or it influences how products and consumers are matched. An interesting research question would be: can we manipulate the prices of the products in the system so that maximum total surplus can be achieved? We are aware that multi-product pricing has been used as a marketing tool for revenue maximization[67, 68, 5]. In future work, we are interested in studying how multi-product pricing can be adopted for total surplus maximization and how it will impact the platform.

Bibliography

- [1] Jack Abrams. A new method for testing pricing decisions. *The Journal of Marketing*, pages 6–9, 1964.
- [2] Adam Smith. An Inquiry into the Nature and Causes of the Wealth of Nations. *Strahan and Cadell, London*, 1776.
- [3] Gediminas Adomavicius, Bamshad Mobasher, Francesco Ricci, and Alex Tuzhilin. Context-Aware Recommender Systems. *Recommender systems handbook*, pages 217–253, 2011.

- [4] Y. Kwon, Adomavicius, G. Toward more diverse recommendations: Item re-ranking methods for recommender systems. In *Proc. of the 19th Workshop on Information Technologies and Systems*, pages 657–666. ACM, 2009.
- [5] Gagan Aggarwal, Tomás Feder, Rajeev Motwani, and An Zhu. Algorithms for multi-product pricing. In *Automata, Languages and Programming*, pages 72–83. Springer, 2004.
- [6] Rakesh Agrawal, Sreenivas Gollapudi, Alan Halverson, and Samuel Ieong. Diversifying search results. In *Proceedings of the Second ACM International Conference on Web Search and Data Mining*, pages 5–14. ACM, 2009.
- [7] Hirotugu Akaike. A New Look at the Statistical Model Identification. *IEEE Transactions on Automatic Control*, 19(6):716–723, 1974.
- [8] Greg M Allenby, Neeraj Arora, and James L Ginter. Incorporating prior knowledge into the analysis of conjoint studies. *Journal of Marketing Research*, pages 152–162, 1995.
- [9] Greg M Allenby, Neeraj Arora, and James L Ginter. On the heterogeneity of demand. *Journal of Marketing Research*, pages 384–389, 1998.
- [10] Jon Scott Armstrong. *Principles of forecasting: a handbook for researchers and practitioners*, volume 30. Springer Science & Business Media, 2001.
- [11] Neeraj Arora, Greg M Allenby, and James L Ginter. A hierarchical bayes model of primary and secondary demand. *Marketing Science*, 17(1):29–44, 1998.

- [12] Neeraj Arora and Joel Huber. Improving parameter estimates and model prediction by aggregate customization in choice experiments. *Journal of Consumer Research*, 28(2):273–283, 2001.
- [13] Tom Ault and Yiming Yang. knn, rocchio and metrics for information filtering at trec-10. In *TREC*, 2001.
- [14] Klaus Backhaus, Jrg Becker, Daniel Beverungen, Margarethe Frohs, Oliver Muller, Matthias Weddeling, Ralf Knackstedt, and Michael Steiner. Enabling individualized recommendations and dynamic pricing of value-added services through willingness-to-pay data. In *Electronic Markets*, 2010.
- [15] Linas Baltrunas and Xavier Amatriain. Towards Time Dependant Recommendation based on Implicit Feedback. *CARs*, 2009.
- [16] Paul A. Baran. Economic Progress and Economic Surplus. *Science & Society*, pages 289–317, 1953.
- [17] Paul A. Baran. Monopoly Capital. *NYU Press*, 1966.
- [18] Justin Basilico and Thomas Hofmann. A joint framework for collaborative and content filtering. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 550–551. ACM, 2004.
- [19] Gordon M Becker, Morris H DeGroot, and Jacob Marschak. Measuring utility by a single-response sequential method. *Behavioral science*, 9(3):226–232, 1964.

- [20] Tilman Borghers, Daniel Krahmer, and Roland Strausz. An Introduction to the Theory of Mechanism Design. *Oxford University Press*, 2015.
- [21] George E. P. Box, Gwilym M. Jenkins, and Gregory C. Reinsel. Time Series Analysis: Forecasting and Control. *John Wiley & Sons*, 2013.
- [22] Christoph Breidert, Michael Hahsler, and Thomas Reutterer. A review of methods for measuring willingness-to-pay. *Innovative Marketing*, 2(4):8–32, 2006.
- [23] Sergey Brin and Lawrence Page. Reprint of: The Anatomy of a Large-Scale Hypertextual Web Search Engine. *Computer Networks*, 56(18):3825–3833, 2012.
- [24] Erik Brynjolfsson, Yu (Jeffrey) Hu, and Michael D. Smith. Research commentary—long tails vs. superstars: The effect of information technology on product variety and sales concentration patterns. *Info. Sys. Research*, 21(4):736–747, December 2010.
- [25] Randolph E Bucklin and Venkataraman Srinivasan. Determining interbrand substitutability through survey measurement of consumer preference structures. *Journal of Marketing Research*, pages 58–71, 1991.
- [26] Michael Buhrmester, Tracy Kwang, and Samuel D. Gosling. Amazon’s Mechanical Turk: a New Source of Inexpensive, yet High-quality, Data? *Perspectives on psychological science*, 6(1):3–5, 2011.
- [27] Kenneth P. Burnham and David R. Anderson. Model Selection and Multimodel Inference: A Practical Information-Theoretic Approach. *Springer*, 2002.

- [28] Pedro G. Campos, Fernando Dez, and Ivan Cantador. Time-aware Recommender Systems: A Comprehensive Survey and Analysis of Existing Evaluation Protocols. *User modeling & user-adapted interaction*, 24:67–119, 2014.
- [29] John Canny. Collaborative filtering with privacy via factor analysis. In *SIGIR*, pages 238–245, 2002.
- [30] Timothy N Cason and Charles R Plott. Misconceptions and game form recognition of the bdm method: challenges to theories of revealed preference and framing. 2014.
- [31] Pablo Castells, Jun Wang, Ruben Lara, and Dell Zhang. Workshop on novelty and diversity in recommender systems. 2011.
- [32] Simla Ceyhan, Xiaolin Shi, and Jure Leskovec. Dynamics of Bidding in a P2P Lending Service: Effects of Herding and Predicting Loan Success. *WWW*, pages 547–556, 2011.
- [33] Dave Chaffey, Fiona Ellis-Chadwick, Richard Mayer, and Kevin Johnston. Internet Marketing: Strategy, Implementation and Practice. *Pearson Education*, 2009.
- [34] Jia Chen, Qin Jin, Shiwan Zhao, Shenghua Bao, Li Zhang, Zhong Su, and Yong Yu. Does product recommendation meet its waterloo in unexplored categories?: No, price comes to help. In *Proceedings of the 37th international ACM SIGIR*

- conference on Research & development in information retrieval*, pages 667–676. ACM, 2014.
- [35] Ting Chen, Weili Han, Haidong Wang, Yixun Zhou, Bin Xu, and Binyu Zang. Content Recommendation System based on Private Dynamic User Profiling. *ICMLC*, 2007.
- [36] Wei Chen, Wynne Hsu, and Mongli Lee. Modeling User’s Receptiveness Over Time for Recommendation. *SIGIR*, pages 373–382, 2013.
- [37] Alexander Chernev. Reverse pricing and online price elicitation strategies in consumer choice. *Journal of Consumer Psychology*, 13(1):51–62, 2003.
- [38] Yun Chi, Shenghuo Zhu, Yihong Gong, and Yi Zhang. Probabilistic polyadic factorization and its application to personalized recommendation. In *Proceedings of the 17th ACM conference on Information and knowledge management*, pages 941–950. ACM, 2008.
- [39] Hyunyoung Choi and Hal Varian. Predicting the Present with Google Trends. *Economic Record*, 88(s1):2–9, 2012.
- [40] Konstantinos Christidis, Dimitris Apostolou, and Gregoris Mentzas. Exploring customer preferences with probabilistic topic models. In *ECML*, 2010.
- [41] Wei Chu and SeungTaek Park. Personalized Recommendation on Dynamic Content Using Predictive Bilinear Models. *WWW*, pages 691–700, 2009.

- [42] Tuck Siong Chung, Roland T Rust, and Michel Wedel. My mobile music: An adaptive personalization system for digital audio players. *Marketing Science*, 28(1):52–68, 2009.
- [43] James S. Coleman and Thomas J. Fararo. Rational Choice Theory. *Nueva York: Sage*, 1992.
- [44] Economic Sciences Prize Committee et al. Stable matching: Theory, evidence, and practical design-the sveriges riksbank prize in economic sciences in memory of alfred nobel 2012: Scientific background. Technical report, Tech. rep. The Nobel Foundation, Stockholm, Sweden, 2012.
- [45] John Martin Currie, John A. Murphy, and Andrew Schmitz. The Concept of Economic Surplus and its Use in Economic Analysis. *The Economic Journal*, pages 741–799, 1971.
- [46] PA Diamond, DT Mortensen, and Ch A Pissarides. Compiled by the economic sciences prize committee of the royal swedish academy of sciences-scientific background on the sveriges riksbank prize in economic sciences in memory of alfred nobel 2010. markets with search friction.
- [47] Min Ding. An incentive-aligned mechanism for conjoint analysis. *Journal of Marketing Research*, pages 214–223, 2007.
- [48] Xiaowen Ding, Bing Liu, and Philip S. Yu. A Holistic Lexicon Based Approach to Opinion Mining. *WSDM*, 2008.

- [49] Anhai Doan, Raghu Ramakrishnan, and Alon Y. Halevy. Crowdsourcing Systems on the World-Wide Web. *Communications of the ACM*, 54:86–96, 2011.
- [50] Pedro Domingos and Matt Richardson. Mining the network value of customers. In *Proceedings of the seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 57–66. ACM, 2001.
- [51] Songting Dong, Min Ding, and Joel Huber. A simple mechanism to incentive-align conjoint experiments. *International Journal of Research in Marketing*, 27(1):25–32, 2010.
- [52] Gideon Dror, Noam Koenigstein, and Yehuda Koren. Yahoo! music recommendations: Modeling music ratings with temporal dynamics and item taxonomy. *RecSys*, 2011.
- [53] Michael D. Ekstrand, John T. Riedl, and Joseph A. Konstan. Collaborative Filtering Recommender Systems. *Foundations and Trends in Human-Computer Interaction*, 4(2):81–173, 2011.
- [54] Daniel Friedman and Skovics Jzsef. Tractable Consumer Choice. *Theory and Decision*, pages 1–26, 2013.
- [55] Andre Gabor, Clive WJ Granger, and Anthony P Sowter. Real and hypothetical shop situations in market research. *Journal of Marketing Research*, pages 355–359, 1970.

- [56] Joshua Gans, Stephen King, and N. Gregory Mankiw. Principles of Microeconomics. *Cengage Learning*, 2011.
- [57] Zeno Gantner, Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. MyMediaLite: A Free Recommender System Library. *RecSys*, 2011.
- [58] Zeno Gantner, Steffen Rendle, and Lars Schmidt-Thieme. Factorization Models for Context-/Time-Aware Movie Recommendations. *CAMRa*, pages 14–19, 2010.
- [59] Anna Gatzoura and Miquel Sanchez-Marre. A case-based recommendation approach for market basket data. *Intelligent Systems, IEEE*, 30(1):20–27, 2015.
- [60] Susan Gauch, Mirco Speretta, Aravind Chandramouli, and Alessandro Micarelli. User Profiles for Personalized Information Access. *The Adaptive Web*, pages 54–89, 2007.
- [61] M Rami Ghorab, Dong Zhou, Alexander O’Connor, and Vincent Wade. Personalised Information Retrieval: Survey and Classification. *User Modeling and User-Adapted Interaction*, 23(4):381–443, 2013.
- [62] William N Goetzmann. *An introduction to investment theory*. William N. Goetzmann., 1996.
- [63] Jennifer Golbeck. *Generating predictive movie recommendations from trust in social networks*. Springer, 2006.
- [64] David Goldberg, David Nichols, Brian M Oki, and Douglas Terry. Using collab-

- orative filtering to weave an information tapestry. *Communications of the ACM*, 35(12):61–70, 1992.
- [65] Ramanathan Guha, Ravi Kumar, Prabhakar Raghavan, and Andrew Tomkins. Propagation of trust and distrust. In *Proceedings of the 13th international conference on World Wide Web*, pages 403–412. ACM, 2004.
- [66] Sunil Gupta. Impact of sales promotions on when, what, and how much to buy. *Journal of Marketing research*, pages 342–355, 1988.
- [67] Ward Hanson and R Kipp Martin. Optimal bundle pricing. *Management Science*, 36(2):155–174, 1990.
- [68] John R Hauser and Patricia Simmie. Profit maximizing perceptual positions: An integrated theory for the selection of product features and price. *Management Science*, 27(1):33–56, 1981.
- [69] Jonathan L Herlocker, Joseph A Konstan, Al Borchers, and John Riedl. An algorithmic framework for performing collaborative filtering. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 230–237. ACM, 1999.
- [70] Thomas Hofmann and Jan Puzicha. Latent class models for collaborative filtering. In *IJCAI*, volume 99, pages 688–693, 1999.
- [71] Neil Hurley and Mi Zhang. Novelty and diversity in top-n recommendation–

- analysis and evaluation. *Transactions on Internet Technology (TOIT)*, 10(4):14, 2011.
- [72] Mohsen Jamali and Martin Ester. Using a trust network to improve top-n recommendation. In *RecSys '09: Proceedings of the third ACM conference on Recommender systems*, pages 181–188, New York, NY, USA, 2009. ACM.
- [73] Toshihiro Kamishima and Shotaro Akaho. Personalized pricing recommender system: Multi-stage epsilon-greedy approach. In *Proceedings of the 2nd International Workshop on Information Heterogeneity and Fusion in Recommender Systems, HetRec '11*, pages 57–64, New York, NY, USA, 2011. ACM.
- [74] Andreas M. Kaplan and Michael Haenlein. Users of the World, Unite! The Challenges and Opportunities of Social Media. *Business Horizons*, 53(1):59–68, 2010.
- [75] Alexandros Karatzoglou, Xavier Amatriain, Linas Baltrunas, and Nuria Oliver. Multiverse Recommendation: N-dimensional Tensor Factorization for Context-aware Collaborative Filtering. *RecSys*, pages 79–86, 2010.
- [76] Henry Kautz, Bart Selman, and Mehul Shah. Referral web: combining social networks and collaborative filtering. *Communications of the ACM*, 40(3):63–65, 1997.
- [77] Simon Kemp. Digital, Social & Mobile in 2015. *We Are Social*, 2015.
- [78] David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the spread of influence

- through a social network. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 137–146. ACM, 2003.
- [79] Nobuhiro Kiyotaki and Randall Wright. A search-theoretic approach to monetary economics. *The American Economic Review*, pages 63–77, 1993.
- [80] Joseph A Konstan, Bradley N Miller, David Maltz, Jonathan L Herlocker, Lee R Gordon, and John Riedl. Grouplens: applying collaborative filtering to usenet news. *Communications of the ACM*, 40(3):77–87, 1997.
- [81] Joseph A. Konstan and John Riedl. Recommender Systems: from Algorithms to User Experience. *User Modeling and User-Adapted Interaction*, 22(1-2):101–123, 2012.
- [82] Ioannis Konstas, Vassilios Stathopoulos, and Joemon M Jose. On social networks and collaborative recommendation. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 195–202. ACM, 2009.
- [83] Y. Koren, R. Bell, and C. Volinsky. Matrix Factorization Techniques for Recommender Systems. *Computer*, 2009.
- [84] Yehuda Koren. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 426–434. ACM, 2008.

- [85] Yehuda Koren. Collaborative Filtering with Temporal Dynamics. *KDD*, pages 447–455, 2009.
- [86] Yehuda Koren. Collaborative filtering with temporal dynamics. *Communications of the ACM*, 53(4):89–97, 2010.
- [87] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, (8):30–37, 2009.
- [88] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.
- [89] Gueorgi Kossinets, Jon Kleinberg, and Duncan Watts. The structure of information pathways in a social communication network. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 435–443. ACM, 2008.
- [90] D. Lee and H. Seung. Learning the parts of objects with nonnegative matrix factorization. *Nature*, 401(6755):788–791, 1999.
- [91] D. D. Lee and H. S. Seung. Algorithms for Non-negative Matrix Factorization. *NIPS*, pages 556–562, 2001.
- [92] Daniel Lemire and Anna Maclachlan. Slope one predictors for online rating-based collaborative filtering. In *SDM*, volume 5, pages 1–5. SIAM, 2005.
- [93] Asher Levi, Osnat Mokryn, Christophe Diot, and Nina Taft. Finding a Needle in

- a Haystack of Reviews: Cold Start Context-Based Recommender System. *RecSys*, pages 115–122, 2012.
- [94] David D Lewis. Applying support vector machines to the trec-2001 batch filtering and routing tasks. In *TREC*, 2001.
- [95] Blerina Lika, Kostas Kolomvatsos, and Stathes Hadjiefthymiades. Facing the Cold Start Problem in Recommender Systems. *Expert Systems with Applications*, 41(4):2065–2073, 2014.
- [96] Y. J. Lim and Y. W. Teh. Variational Bayesian approach to movie rating prediction. In *Proceedings of KDD Cup and Workshop*, 2007.
- [97] Greg Linden, Brent Smith, and Jeremy York. Amazon. com recommendations: Item-to-item collaborative filtering. *Internet Computing, IEEE*, 7(1):76–80, 2003.
- [98] Pasquale Lops, Marco De Gemmis, and Giovanni Semeraro. Content-based recommender systems: State of the art and trends. In *Recommender systems handbook*. Springer, 2011.
- [99] Jordan J Louviere and George Woodworth. Design and analysis of simulated consumer choice or allocation experiments: an approach based on aggregate data. *Journal of marketing research*, pages 350–367, 1983.
- [100] Yue Lu, Malu Castellanos, Umeshwar Dayal, and ChengXiang Zhai. Automatic construction of a context-aware sentiment lexicon: An optimization approach. *WWW*, 2011.

- [101] Zhengdong Lu, Deepak Agarwal, and Inderjit Dhillon. A Spatio Temporal Approach to Collaborative Filtering. *RecSys*, 2009.
- [102] Chunyu Luo, Hui Xiong, Wenjun Zhou, Yanhong Guo, and Guishi Deng. Enhancing Investment Decisions in P2P Lending :An Investor Composition Perspective. *KDD*, pages 292–300, 2011.
- [103] Hao Ma, Irwin King, and Michael R. Lyu. Learning to recommend with social trust ensemble. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 203–210. ACM, 2009.
- [104] B. Mahadevan. Business Models for Internet-based E-commerce. *California Management Review*, 42(4):55–69, 2000.
- [105] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schtze. Introduction to Information Retrieval. *Cambridge: Cambridge university press*, 1, 2008.
- [106] Yves Marbeau. What value pricing research today. *Journal of the Market Research Society*, 29(2):153–182, 1987.
- [107] Alfred Marshall and Claude William Guillebaud. Principles of Economics. *World Scientific*, 1961.
- [108] Karl Marx. *Wages, Prices and Value Added*. 1865.
- [109] Karl Marx, G. A. Bonner, and Emile Burns. Theories of Surplus Value. *Capital*, Volume IV, 1863.

- [110] Andreu Mas-Colell, Michael Dennis Whinston, Jerry R Green, et al. *Microeconomic theory*, volume 1. Oxford university press New York, 1995.
- [111] Eric S. Maskin. Mechanism Design: How to Implement Social Goals. *The American Economic Review*, pages 567–576, 2008.
- [112] Paolo Massa and Bobby Bhattacharjee. Using trust in recommender systems: an experimental analysis. In *Trust Management*, pages 221–235. Springer, 2004.
- [113] Mohamed Massoud and Mohamed Abo-Rizka. A conceptual model of personalized pricing recommender system based on customer online behavior. 2012.
- [114] J. J. McAuley, R. Pandey, and J. Leskovec. Inferring networks of substitutable and complementary products. In *KDD*, 2015.
- [115] Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton van den Hengel. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 43–52. ACM, 2015.
- [116] Daniel McFadden. Econometric models for probabilistic choice among products. *Journal of Business*, pages S13–S29, 1980.
- [117] Klaus M Miller, Reto Hofstetter, Harley Krohmer, and Z John Zhang. How should consumers’ willingness to pay be measured? an empirical comparison of state-of-the-art approaches. *Journal of Marketing Research*, 48(1):172–184, 2011.

- [118] BPS Murthi and Sumit Sarkar. The role of the management sciences in research on personalization. *Management Science*, 49(10):1344–1362, 2003.
- [119] Kenta Oku, Shinsuke Nakajima, Jun Miyazaki, Shunsuke Uemura, and Hirokazu Kato. A Recommendation Method Considering Users’ Time Series Contexts. *ICUIMC*, 2009.
- [120] Arkadiusz Paterek. Improving regularized singular value decomposition for collaborative filtering. In *Proceedings of KDD Cup and Workshop*, 2007.
- [121] M. J. Pazzani and D. Billsus. Content-Based Recommendation Systems. *The Adaptive Web LNCS*, pages 325–341, 2007.
- [122] Christopher A Pissarides. *Equilibrium unemployment theory*. MIT press, 2000.
- [123] Ana Maria Popescu and Oren Etzioni. Extracting Product Features and Opinions from Reviews. *EMNLP*, 2005.
- [124] Pearl Pu, Li Chen, and Rong Hu. A user-centric evaluation framework for recommender systems. In *RecSys*, pages 157–164. ACM, 2011.
- [125] Anita Ramasastry. Web sites change prices based on customers’ habits. <http://www.cnn.com/2005/LAW/06/24/ramasastry.website.prices/>, 2005. [Online; accessed 02-21-2016].
- [126] Steffen Rendle. Factorization machines. In *Data Mining (ICDM), 2010 IEEE 10th International Conference on*, pages 995–1000. IEEE, 2010.

- [127] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pages 452–461. AUAI Press, 2009.
- [128] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt Thieme. BPR: Bayesian Personalized Ranking from Implicit Feedback. *UAI*, 2009.
- [129] Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. Factorizing personalized markov chains for next-basket recommendation. In *Proceedings of the 19th international conference on World wide web*, pages 811–820. ACM, 2010.
- [130] Jasson D. M. Rennie and Nathan Srebro. Fast maximum margin matrix factorization for collaborative prediction. *ICML*, pages 713–719, 2005.
- [131] Paul Resnick, Neophytos Iacovou, Mitesh Suchak, Peter Bergstrom, and John Riedl. GroupLens: an open architecture for collaborative filtering of netnews. In *Proceedings of the 1994 ACM conference on Computer supported cooperative work*, pages 175–186. ACM, 1994.
- [132] Francesco Ricci, Lior Rokach, and Bracha Shapira. Introduction to Recommender Systems Handbook. *Springer US*, 2011.
- [133] Elaine Rich. User modeling via stereotypes*. *Cognitive science*, 3(4):329–354, 1979.

- [134] Manuel C. Rios, Campbell R McConnell, and Stanley L Brue. *Economics: Principles, problems, and policies*. McGraw-Hill, 2013.
- [135] George Ritzer and Nathan Jurgenson. Production, Consumption, Prosumption: The Nature of Capitalism in the Age of the Digital Prosumer. *Journal of Consumer Culture*, 10(1):13–36, 2010.
- [136] Stephen E Robertson and Ian Soboroff. The trec 2002 filtering track report. In *TREC*, volume 2002, page 5, 2002.
- [137] Murray Rothbard. An Austrian Perspective on the History of Economic Thought Volume II: Classical Economics. 1995.
- [138] Paat Rusmevichientong, Shenghuo Zhu, and David Selinger. Identifying early buyers from purchase data. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 671–677. ACM, 2004.
- [139] Gary J Russell and Ruth N Bolton. Implications of market structure for elasticity structure. *Journal of Marketing Research*, pages 229–241, 1988.
- [140] Gary J Russell and Ann Petersen. Analysis of cross category dependence in market basket selection. *Journal of Retailing*, 76(3):367–392, 2000.
- [141] R. Salakhutdinov and A. Mnih. Bayesian probabilistic matrix factorization using markov chain monte carlo. *ICML*, pages 880–887, 2008.

- [142] Ruslan Salakhutdinov, Andriy Mnih, and Geoffrey Hinton. Restricted boltzmann machines for collaborative filtering. In *Proceedings of the 24th international conference on Machine learning*, pages 791–798. ACM, 2007.
- [143] B. Sarwar, G. Karypis, et al. Item-based collaborative filtering recommendation algorithms. *WWW*, 2001.
- [144] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. Analysis of Recommendation Algorithms for E-commerce. In *Proceedings of the 2nd ACM conference on Electronic Commerce (EC)*, pages 158–167. ACM, 2000.
- [145] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. Application of dimensionality reduction in recommender system-a case study. Technical report, DTIC Document, 2000.
- [146] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*, pages 285–295. ACM, 2001.
- [147] John Shawe-Taylor, Nicola Cancedda, Nicolo Cesa-Bianchi, Alex Conconi, Claudio Gentile, Cyril Goutte, Thore Graepel, Yaoyong Li, and Jean-Michel Renders. Kernel methods for document filtering. 2002.
- [148] Yue Shi, Martha Larson, and Alan Hanjalic. Mining Mood-specific Movie Similarity with Matrix Factorization for Context-aware Recommendation. *CAMRa*, pages 34–40, 2010.

- [149] Yue Shi, Xiaoxue Zhao, Jun Wang, Martha Larson, and Alan Hanjalic. Adaptive diversification of recommendation results via latent factor portfolio. In *SIGIR*, SIGIR '12, pages 175–184, New York, NY, USA, 2012. ACM.
- [150] Allan D Shocker, Barry L Bayus, and Namwoon Kim. Product complements and substitutes in the real world: The relevance of other products. *Journal of Marketing*, 68(1):28–40, 2004.
- [151] Robert H. Shumway and David S. Stoffer. Time Series Analysis and Its Application. *Springer*, 2010.
- [152] Luo Si and Rong Jin. Flexible mixture model for collaborative filtering. In *ICML*, volume 3, pages 704–711, 2003.
- [153] Xiaodan Song, Yun Chi, Koji Hino, and Belle L Tseng. Information flow modeling based on diffusion rate for prediction and ranking. In *Proceedings of the 16th international conference on World Wide Web*, pages 191–200. ACM, 2007.
- [154] Xiaodan Song, Belle L Tseng, Ching-Yung Lin, and Ming-Ting Sun. Personalized recommendation driven by information flow. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 509–516. ACM, 2006.
- [155] N. Srebro and T. Jaakkola. Weighted low-rank approximations. *ICML*, 2003.
- [156] Nathan Srebro, Tommi Jaakkola, et al. Weighted low-rank approximations. In *ICML*, volume 3, pages 720–727, 2003.

- [157] Roy G Stout. Developing data to estimate price-quantity relationships. *The Journal of Marketing*, pages 34–36, 1969.
- [158] Xiaoyuan Su and Taghi M. Khoshgoftaar. A Survey of Collaborative Filtering Techniques. *Advances in AI*, 4, 2009.
- [159] G. Takacs, I. Pitaszy, B. Nemeth, and D. Tikk. Investigation of Various Matrix Factorization Methods for Large Recommender Systems. *Proc. ICDM*, 2008.
- [160] Duyu Tang, Furu Wei, Bing Qin, Ming Zhou, and Ting Liu. Building Large-Scale Twitter-Specific Sentiment Lexicon: A Representation Learning Approach. *COLING*, pages 172–182, 2014.
- [161] Sarah K Tyler and Yi Zhang. Open domain recommendation: Social networks and collaborative filtering. In *Advanced Data Mining and Applications*, pages 330–341. Springer, 2008.
- [162] Carmen Vaca, Amin Mantrach, Alejandro Jaimes, and Marco Saerens. A Time-based Collective Factorization for Topic Discovery and Monitoring in News. *WWW*, 2014.
- [163] Hal R Varian. Price discrimination. *Handbook of industrial organization*, 1:597–654, 1989.
- [164] Hal R Varian and WW Norton. *Microeconomic analysis*, volume 2. Norton New York, 1992.

- [165] Hal R Varian and Jack Repcheck. *Intermediate microeconomics: a modern approach*, volume 6. WW Norton & Company New York, NY, 2010.
- [166] William Vickrey. Counterspeculation, auctions, and competitive sealed tenders. *The Journal of finance*, 16(1):8–37, 1961.
- [167] Frank E Walter, Stefano Battiston, and Frank Schweitzer. Personalised and dynamic trust in social networks. In *Proceedings of the third ACM conference on Recommender systems*, pages 197–204. ACM, 2009.
- [168] J. Wang and Y. Zhang. Utilizing marginal net utility for recommendation in e-commerce. In *SIGIR 2011*. ACM.
- [169] Jian Wang and Yi Zhang. Utilizing marginal net utility for recommendation in e-commerce. In *SIGIR*, pages 1003–1012. ACM, 2011.
- [170] Jian Wang and Yi Zhang. Is It Time For a Career Switch? *WWW*, pages 1377–1387, 2013.
- [171] Jian Wang and Yi Zhang. Opportunity Models for E-commerce Recommendation: Right Product, Right Time. *SIGIR*, pages 303–312, 2013.
- [172] Pengfei Wang, Jiafeng Guo, and Yanyan Lan. Modeling retail transaction data for personalized shopping recommendation. In *CIKM*, pages 1979–1982, 2014.
- [173] Pengfei Wang, Jiafeng Guo, Yanyan Lan, Jun Xu, Shengxian Wan, and Xueqi Cheng. Learning hierarchical representation model for nextbasket recommendation. In *SIGIR*, 2015.

- [174] Yao Wu and Martin Ester. FLAME: A Probabilistic Model Combining Aspect Based Opinion Mining and Collaborative Filtering. *WSDM*, pages 199–208, 2015.
- [175] Liang Xiang, Quan Yuan, Shiwan Zhao, Li Chen, Xiatian Zhang, Qing Yang, and Jimeng Sun. Temporal Recommendation on Graphs via Long- and Short-term Preference Fusion. *KDD*, pages 723–731, 2010.
- [176] Danqing Xu, Yiqun Liu, Min Zhang, Shaoping Ma, Anqi Cui, and Liyun Ru. Predicting Epidemic Tendency through Search Behavior Analysis. *IJCAI*, pages 2361–2366, 2011.
- [177] Yiming Yang, Shinjae Yoo, Jian Zhang, and Bryan Kisiel. Robustness of adaptive filtering methods in a cross-benchmark evaluation. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 98–105. ACM, 2005.
- [178] Ainur Yessenalina, Yisong Yue, et al. Multi-level structured models for document-level sentiment classification. *EMNLP*, pages 1046–1056, 2010.
- [179] Cong Yu, Laks Lakshmanan, and Sihem Amer-Yahia. It takes variety to make a world: diversification in recommender systems. In *Proceedings of the 12th international conference on extending database technology: Advances in database technology*, pages 368–378. ACM, 2009.
- [180] Quan Yuan, Gao Cong, Zongyang Ma, Aixin Sun, and Nadia Magnenat Thal-

- mann. Time-aware Point-of-interest Recommendation. *SIGIR*, pages 363–372, 2013.
- [181] Yi Zhang. Using bayesian priors to combine classifiers for adaptive filtering. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 345–352. ACM, 2004.
- [182] Yongfeng Zhang, Guokun Lai, Min Zhang, Yi Zhang, Yiqun Liu, and Shaoping Ma. Explicit Factor Models for Explainable Recommendation based on Phrase-level Sentiment Analysis. In *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 83–92. ACM, 2014.
- [183] Yongfeng Zhang, Guokun Lai, Min Zhang, Yi Zhang, Yiqun Liu, and Shaoping Ma. Explicit Factor Models for Explainable Recommendation based on Phrase-level Sentiment Analysis. *SIGIR*, pages 83–92, 2014.
- [184] Yongfeng Zhang, Haochen Zhang, Min Zhang, Yiqun Liu, et al. Do Users Rate or Review? Boost Phrase-level Sentiment Labeling with Review-level Sentiment Classification. *SIGIR*, pages 1027–1030, 2014.
- [185] Yongfeng Zhang, Min Zhang, Yi Zhang, Yiqun Liu, and Shaoping Ma. Understanding the Sparsity: Augmented Matrix Factorization with Sampled Constraints on Unobservables. *CIKM*, 2014.
- [186] Yongfeng Zhang, Qi Zhao, Yi Zhang, Daniel Friedman, and Min Zhang. Web in-