CHOICE MODELING AND RECOMMENDATION OPTIMIZATION IN PRESENCE OF CONTEXT EFFECTS

BY

REZA YOUSEFI MARAGHEH

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Industrial Engineering
in the Graduate College of the
University of Illinois Urbana-Champaign, 2021

Urbana, Illinois

Doctoral Committee:

Professor Xin Chen, Chair
Professor Sridhar Seshdari
Professor S. Rasoul Etesami
Professor Yuan Zhou
Dr. Kannan Achan
Dr. Jason Cho

Abstract

Random Utility choice models are supervised learning tools that can be used to estimate the choice behavior of customers facing multiple options. This is accomplished through assigning a utility value to each option and deriving a choice probability for each option. In the presence of context effects, the utility perceived from individual options is not fixed and depends on other options that are offered beside them. While context effects are well explored in the marketing and psychology literature, very little work has been done on incorporating these effects in revenue management systems and product recommendation modules. In this thesis, we propose three sets of machine learning models in order to capture these effects in different settings with different input data structures. For these settings, we also study combinatorial problems concerned with finding the optimal set of products to offer to the customer including (i) assortment optimization problem or reward maximization problem, (ii) click through rate optimization problem, and (iii) customer surplus optimization problem.

The first model we propose is a random utility discrete choice model which captures context effects in sparse choice/click data sets and under single-choice outcome assumption. In the proposed model, the perceived utilities from products are dependent on the whole choice set recommended to the customer, and choice probabilities have Multinomial Logistic Regression-type structure. We show the prediction power of this model by testing it on a relevant real data set and prove the NP-hardness of the assortment optimization problem under the proposed model. Several polynomially solvable special cases of the model are identified that also perform well in our empirical validation for our data set. We obtain some easily verifiable conditions

for the monotonicity and submodularity of the assortment optimization objective in order to provide some approximation guarantees.

Second, we propose a utility based listwise logistic regression model, which is applicable in estimating the context effects in dense data sets with a multi-choice outcome assumption. We show the predictive and descriptive power of this model through an extensive empirical study on real click data sets chosen from diverse categories of products. We prove the NP-hardness of the Assortment Optimization Problem (AOP) under the general CL model, and show that when some specific types of contextual interactions are dominant in the data, the AOP is tractable.

Third, we propose a featurized choice model, in order to capture context effects when the input data is featurized. We study the top-K retrieval problem which focuses on finding K relevant products/documents for a given query. We train a featurized estimator that can measure the context effects among the objects through mapping their features to contextual interaction terms by using an underlying neural net structure. We empirically validate the estimator on a real data set and prove the NP-hardness of the top-K retrieval problem for the proposed model. For all three sets of models, to circumvent NP-hardness, we design heuristic algorithms and test their efficiency through extensive numerical studies.

Different models proposed in this thesis and the relevant empirical studies, as well as the recommendation optimization results, shed more light on the contextual behavioral patterns observed in customers' choice behavior in e-commerce platforms, and how to further optimize the recommender systems by considering these patterns. To the best of our knowledge, this thesis is the first systematic study and its findings can help in designing operational recommender systems that capture complex contextual patterns in large scale data sets.

To my wife, Ati, for always being there for me,

and,

To my parents, for their endless love, support and encouragement.

Acknowledgments

First of all, I would like to express my sincere regards to my supervisor, Professor Xin Chen for all his guidance and support. I owe him a lifetime of gratitude for his dedication in teaching me critical thinking, problem solving, and conducting meaningful research will me make me. His high standard has pushed me to think of the correct problems to tackle and without his guidance this thesis would not have been possible.

I wish to express my gratitude to my thesis committee members Professor Sridhar Seshdari, Professor S. Rasoul Etesami, Professor Yuan Zhou, Dr. Kannan Achan, and Dr. Jason Cho for their time, valuable comments and suggestions. I would like to thank Professor James Davis, and Professor Alexandra Chronopoulou for all their constructive feedback for my reasearch.

I also would like to specifically thank Dr. Jason Cho in recognizing my research and helping me patiently during my last year of study. I should also thank Dr. Kannan Achan, Luyi Ma, Sushant Kumar, Chuwanwei Ruan, and Venugopal Mani, and all the members of the Personalization team at Walmart Labs for all their support and help in providing data, tools of research, and funding during the last year of my Ph.D. research. Their help was critical in making my research more problem oriented and applicable.

I am blessed to have had the opportunity to attend so many great courses given by distinguished professors in the University of Illinois at Urbana-Champaign. I would like to thank Prof. Ruoyu Sun, Prof. Yuan Zhou, and Prof. James M. Davis, and many others for their excellent teaching and inspiring material.

I am thankful to the faculty and staff of the Department of Industrial and Enter-

prise Systems Engineering at University of Illinois. I am very thankful to Professor Ramavarapu S. Sreenivas, Holly Kizer, Lauren M. Redman, Professor Qiong Wang, and Harry Wildblood for their continuous support and help during my studies.

I would like to extend my sincere gratitude to my labmates and classmates, Ebrahim Arian, Jafar Abbaszadeh Chekan, Roshanak Khaleghi, Dedy Suryadi, Juan Xu, Yifan Hu, Qi Zhao, Tiancheng Zhao, Menglong Li, Albert E. Patterson, Reza Soleymanifar, Peter McGlaughlin, Sagnik Das, Jialin Song, Massi Amrouche, and Arun Raman for all making my Ph.D. studies more joyful.

Finally, I would like to thank my dear wife, Atyeh, for all her unconditional support and presence. Atyeh who has been there for me throughout the good and bad days of the past few years and was always there to encourage me in spite of having her own Ph.D. work to handle. I am grateful to my parents for their endless love and support. Thank you both for inspiring me to become an engineer and a researcher; and to pursue all these years of school and work. My dearest sister, Shabnam and her newborn son, dear Mehrsam deserve my wholehearted thanks as well.

Contents

List of Tables ix

List of Figures xi

Chapter 1 Introduction 1
1.1 Discrete Choice Modeling and Context Effects 2
1.2 Multi-choice Modeling frameworks 5
1.3 Top-K Retrieval Task and Context Effects 7
1.4 Contributions and Outline 8

Chapter 2 Literature Review 12
2.1 Discrete Choice Modeling 13
2.2 Multi-Choice Modeling 15
2.3 Ranking Models and Information Retrieval 16

Chapter 3 Discrete Choice Modeling and Assortment Optimization in the presence of Context Effects 19
3.1 Modeling 20
3.2 Parameter Estimation 26
3.3 Empirical Studies 31
3.4 Polynomially Solvable Special Cases of the CMNL Model 38
3.5 Approximation Algorithms, Conditions for Monotonicity and Submodularity 42
3.6 Heuristics and Numerical Study 45

Chapter 4 Multi-choice Modeling with Context Effects 53
4.1 Model Description 54
4.2 Empirical Validation 60
4.3 Assortment Optimization and Click Through Rate Optimization Problems 69

Chapter 5 Top-K Retrieval Problem with Featurized Contextual Model . . . 78
5.1 Model and Estimation . . . 79
5.2 Top-K Retrieval . . . 84
5.3 Experiments . . . 87

Chapter 6 Conclusion and Future Work . . . 92

Appendix A . . . 97

Appendix B . . . 112

Appendix C . . . 116
C.1 Proofs . . . 116
C.2 Note on the equivalence of SMP to maximizing the binary polynomial problem . . . 118
C.3 More on Numerical Results . . . 118

Bibliography . . . 120

List of Tables

3.1 Special Cases of the CMNL Model 31
3.2 AIC and BIC scores for the CMNL model, its special cases, and the benchmark DCMs 34
3.3 Average of cross entropy and number of rejections of CMNL model, special cases of the CMNL, and the benchmark DCMs in Test Sets for difference train-test splits 37
3.4 Mean optimality gap percentage of AOP with an underlying CMNL model for different heuristic with instant sizes 5-40. 49
3.5 Median\Mean optimality gap percentage of AOP with an underlying CMNL model for different Heuristic with instant sizes 50-100. . 51
3.6 Mean CPU-time for different heuristic for solving AOP with an underlying CMNL model with different instant sizes. 52

4.1 Special Cases of the CL Model 61
4.2 Average percent improvement in log-likelihood ($\hat{L}$), AIC, BIC, Cross-Entropy(CE), MRR, and NDCG scores of different models w.r.t the Point-wise Logistic Regression (PW-logReg) Model. As MRR and NDCG are normalized scores we also report the average value of the scores for the models inside parenthesis. 66
4.3 Mean\Median optimality gap percentage for different heuristic solving the AOP with underlying CL model with instant sizes 30-100. . . 76
4.4 Mean CPU time spent under each heuristics when solving the AOP with an underlying CL model for each instance sizes. 77

5.1 **Algorithm 1** SGD Algorithm for CMNL-Net. 81

5.2 MRR, NDCG, CE scores for different models and % improvement when compared to the benchmark models. The first two columns show the scores for Top-1, Top-3 ListMLE. Last three rows show the percent improvement of contextual models w.r.t. best benchmark model. 88

C.1 Numerical results on average surplus for each recommendation algorithms and problem size 119
C.2 Numerical results on average CTR for each recommendation algorithms and problem size 119
C.3 Numerical results on average expected reward for each recommendation algorithms and problem size 119

List of Figures

3.1 An example of recommended items in Customers-also-considered section 33

4.1 Example for illustrating the similarity effect. (a) Items recommended in product recommendation module, (b) Second item is replaced with a replica of first item in order to illustrate the similarity effect. 57

4.2 Some of the items offered in the Customers also bought these products can be considered complementary for each other. 58

4.3 The recommendation modules (each row) that can be recommended in https://www.walmart.com/grocery/. The interactions are either positive (due to some degree of complementarity) or zero (as some module pairs are independent). 59

4.4 An example for the recommended items in the Customer- also-considered module of the item page. The recommended products recommend alternative substitutes for the anchor item of the page. . 63

5.1 Percent improvement in Surplus, CTR, and Expected Reward of CMNL-Net model and its sub-models w.r.t. to the best of the benchmark models 90

Chapter 1

Introduction

1.1 Discrete Choice Modeling and Context Effects

Predicting choice outcomes plays a major role in modeling customers' behavior as well as designing and implementing of revenue management systems. For instance, product recommendation modules in e-commerce platforms can benefit from increased capability in predicting customers' click/purchase behavior.[1] If the models can predict customers' behavior better, they can offer a more appropriate set of products to the customer and end up increasing indices which are relevant to customer satisfaction, Click Through Rate (CTR), and revenue related metrics.

Discrete Choice Models (DCMs) are machine learning models that try to predict the behavior of users/customers when facing multiple options but with a single-choice outcome. Note that the person who decides to choose from multiple options can choose just one of the options (which we refer to as "single-choice outcome") or can choose more than one option (which we refer to as "multi-choice outcome"). Single-choice outcome is expected in some industries like lodging, or in general merchandise where customers end up buying just one of the alternatives. However, Multi-choice outcome can happen in some other cases, for instance, when buying groceries or household essentials. Most of the choice models used in revenue management systems are DCMs and the integration of DCMs in revenue management problems was an important step in modeling demand dependencies of substitutable products.

Most of the DCMs have random utility based justifications. In other words, under these models, we can associate a utility to each option which is random. This utility tries to approximate how much the users/customers like any given option, and to capture the variability of choice decisions, they assume the utility to be random. DCMs with Random Utility justification enjoy nice interpretations and explainability. Also, some of the revenue/recommendation optimization problems are well-explored under these set of models as apposed to other frameworks for choice prediction like Factorization machines (see Train 2009 and Gallego et al. 2019 for more on these models).

[1] Through out this thesis, we use terms "selection", "click", and "purchase" interchangeably unless we explicitly state otherwise. Also, the terms "customer" and "user"; as well as the terms "product" and "item" are used interchangeably.

In addition, the DCMs used in these settings, like the famous Multinomial Logit (MNL) model, were mostly derived from the "Theory of Rational Choice" (ToRC) and its corresponding underlying assumptions. According to these underlying assumptions, the perceived utility that customers get from individual products is merely dependent on the product's own features. Thus, although the demand for a given product may vary in case of availability/unavailability of other substitutable or complementary products, the perceived utility from the product–or how much the customers like the product– does not vary and is not dependent on the presence or absence of other items (see Luce 1959 for more details on ToRC).

However, in many instances in the marketing and cognitive science literature, researchers have shown that this independence of utility from availability/unavailability of other items is not consistent with the choice observations and is in contrast with the so-called "context effects" (Bettman et al. 1998, Huber et al. 1982, Rooderkerk et al. 2011, Simonson 1989, Tversky 1972, Tversky and Simonson 1993). Context effects refer to the changes in the perception about the preferability of a given alternative that depends on the presence or absence of other options besides the given alternative (Trueblood et al. 2013). For example, according to the ToRC, the relative preference of pairs of products is independent of the presence of other products. This is called the Independence of Irrelevant Alternative property (IIA). However, it has been well-documented that this is not the case in practice (Berlyne 1960, Borle et al. 2005, Kahn 1995, Kahn and Lehmann 1991, Kalyanam et al. 2007).

In addition, the ToRC implies that the choice share of an individual item should not increase when adding other new items to the offered set. However, this property is shown to be violated in real-world examples. For instance, Simonson and Tversky (1992) observed the choice behavior of customers when offering Williams-Sonoma bread making devices. Sales of a device were almost doubled after the introduction of a more expensive but similar item and the new device obtained negligible sale share. This phenomenon is called the "Halo effect" in the psychology literature (Thorndike 1920). For more on the Halo effect in a choice setting see Feng et al. (2018).

Another implication of the ToRC is known as the "betweenness inequality". Ac-

cording to this assumption, the introduction of new top-level items hits the market share of middle-level options more than the low-level options. Again, in some real-world examples, exactly the opposite occurs (Rooderkerk et al. 2011, Tversky and Simonson 1993).

By observing the above inconsistencies of choice models based on ToRC, researchers of marketing and cognitive science began to consider the context effects. When assuming the presence of context effects in a model, it is implied that the choice behavior is partly determined by the context that is provided by the available choice options; and the utilities of products are created in a choice context rather than recalled or inferred exogenously (Bettman et al. 1998).

The marketing and cognitive science literature provided ample evidence for the existence of three important types of context effects as follows:

- **The Attraction Effect or Decoy Effect Huber et al. (1982):** A "decoy item" (also known as "asymmetrically dominated item") is a product which is dominated feature-wise by just one of the items, the "target item", of the choice set and not the others. For instance, the decoy item can be more expensive than the target item, even though it has lower quality-related features. According to this effect, adding a decoy item will increase the perceived preference of the target item, i.e. target item will look more preferable in the presence of the decoy item.

- **The compromise Effect Simonson (1989):** It is referred to the phenomenon when an item gains a higher share in terms of probability of purchase if it is the middle option in a choice set. In particular, in presence of this effect, when adding an extreme option (very high-level or basic product), the share of the middle-level options will increase. See Wernerfelt (1995) for more on intuition and examples.

- **The Similarity Effect Tversky (1972):** Adding an item will hurt the market share and perceived preference of the options which are similar to it more than the dissimilar options. The famous "Red Bus; Blue Bus" example can be

justified with the similarity effect: Adding a red bus, will hurt the probability share of the "similar" blue bus more than the other non-similar options for transportation.

The context effects are well explored in marketing and psychology literature. However, in the revenue management literature, there is much less work that incorporates these effects in modeling choice outcomes or demand substitution patterns.

On chapter 3 we propose a discrete choice model that can potentially capture context effects and study some of the relevant revenue management problems for this model.

1.2 Multi-choice Modeling frameworks

As mentioned, most of the models which are used to predict choice in revenue management systems are Discrete Choice Models (DCM) with single-choice outcome assumption. However, there are some business settings that single choice outcome assumption is no longer valid. For instance, when modeling grocery purchase decisions, we should note that each customer may end up buying more than one item with a high probability.

To incorporate multi-choice outcome assumption, some modeling solutions (mostly extensions of DCMs) are proposed in marketing and cognitive sciences literature, and very recently, implementing multi-choice models became popular research topic in operations management field. The modeling approaches which these revenue management problems are based on, can be divided into three categories.

One approach is the use of so-called Multivariate (MV) models, including Multivariate Logit (MVL) and Multivariate Multinomial Logistic (MVMNL) (See Seetharaman et al. 2005, Aurier and Mejia 2014 for more details on these models). In this approach bundles of products are considered as choices. MV-type models are in fact the extensions of simpler models like PointWise Logistic Regression (PW-LogReg) (Liu 2011), under which it is assumed that the utility of a bundle is the sum of

the utilities of products in the bundle. MVs extend PW-LogReg by adjusting the utilities of other items when customers "choose" more than one item.

The second approach is based on Random Utility models in a ranking framework, which allows choosing several top products in the rankings. This approach includes calculating the probability of all rankings of products that are consistent with the purchases/clicked bundle (see Xia et al. 2008 for the ranking model version of Multinomial Logistic Regression model), which are know to be complex to analyse.

In the third approach to model multi-purchase/click proposed recently by Gallego and Wang (2019), customers determine a utility threshold and pick whatever products having utilities larger than the threshold, while specifying a constraint on the number of products to pick. Focusing on justifying the proposed model, Gallego and Wang (2019) do not provide a rigorous estimation procedure that is tractable for large data.

MV-type models adjust the utility of products when a co-purchase happens, while the mentioned ranking and threshold utility models consider a fixed utility for products. However, it it has been well-established that even impressing (showing) an item may lead to significant changes in the perceived utility from items, due to the presence of context effects (See Yousefi Maragheh et al. 2020a).

Note that in a multi-choice setting, we can have contextual interactions between both complementary and substitute products in a multi click/purchase setting, however, the nature of the effects and underlying reasons will be different. For complementary products, when we offer them beside each other, this may influence the perceived utility of individual items positively as customers consider buying each one of the products more when they complement each other properly, which is referred as **Complementarity Effect** (see Manchanda et al. 1999).

The nature of interactions among the substitute products is more diverse and can be divided into three mentioned types types: (i) **Similarity Effect**, (ii) **Compromise Effect**, and (iii) **Attraction/Decoy Effect** as well as the (iv) **Synergistic Halo Effect** which is referred to cognitive biases that can be created when the customers associate one feature of a product like brand with quality while avoiding probe of other features (Thorndike 1920). Feng et al. (2018) mentions an example

of camera options, two of which are from the same brand (Cannon). If one of the Cannon cameras has a very high average customer rating, this can make customers optimistic about the brand Cannon in general; and thus, having a positive effect on the utility of the other Cannon camera.

None of the above mentioned three modeling approaches can capture some or all of the contextual interaction types. In chapter 4, we propose an extension of logistic regression model that can be used to incorporate context effects in multi-choice settings.

1.3 Top-K Retrieval Task and Context Effects

In addition the importance of considering context effects when devising revenue management systems, context effects may affect the efficiencies of Information Retrieval (IR) tasks.

One of the core tasks in IR is to obtain all relevant objects to any given query[2]. However, in some cases, instead of retrieving all the relevant objects, one may focus on retrieving K most relevant objects, or the Top-K retrieval task. In some IR applications like user homepage recommendation modules in e-commerce K is explicitly given and the recommendation module allows for only presenting at most K items. In some other applications like web search engines, K maybe implicitly inferred from user behavior as studies show that users tend to focus on selecting from first few retrieved results. [3]

To retrieve top-K relevant objects, one natural approach is to rank all the relevant objects to a given query, and then select the objects in the first K positions. Under this approach, ranking can be done with one of the Learning To Rank (LTR) models. Typically, LTR models are trained on user feedback data via minimizing a

[2]Examples of <query, object> could be <search query, document> for a web search engine, or <query item, recommendation> in e-Commerce. We use terms objects and documents interchangeably throughout this paper

[3]iProspect Search Engine User Behavior Study, April 2006, http://www.iprospect.com/; According this user study, 62% of search engine users only click on the results within the first page, and 90% of users click on the results within the first three pages.

loss function that reflects the discrepancies between the predicted and observed user behavior in the training data. The models are used to find a relevance score for each object, and thus, for Top-K retrieval one can simply choose the Top-K scores. This approach may work well when the scores of objects do not depend on other objects presented besides it.

However, this may not work in the presence of context effects. In other words, in a given IR setting, the score or CTR of a given object can be a function of other objects besides it in some application areas, and in presence of context effects, users compare the objects before clicking. In the case of Top-K retrieval, the user's behavior w.r.t. a given presented object, may be a function of other $K-1$ presented objects. Because of this, in the presence of context effects, using the typical LTR models to find the scores of objects and then, selecting the Top-K scores may lead to inefficient recommendations. To the best our knowledge there is no paper about top-K retrieval which explicitly considers all kinds of context effects.

1.4 Contributions and Outline

In this thesis, we propose three types of choice models in order to capture context effects in different choice settings and under different data inputs and study the relevant recommendation problems that arise in those settings. These models can be considered as one of the very first models that both capture context effects and can potentially be operational in product recommendation systems. For every set of models that we present in this thesis we empirically validate the models using real data sets and also study the relevant recommendation optimisation problems that arise in the settings that the model is valid.

In Chapter 3, a Utility-based DCM is proposed that can potentially approximate the above context effects on the perceived attractiveness of items under a single-choice outcome and in sparse choice data sets. In particular, an extension of the MNL model is developed which captures the effect of eliminating other products from the offered assortment on the perceived utility of a given product. In other

words, in the model, the utilities of presented products are linearly dependent on the presence and absence of other products, and the perceived utility of a given product changes if any of the other products become unavailable. This effect on the perceived utility can be both positive and negative, making the model flexible to be used under different scenarios or types of context effects. We call this model "Contextual MNL" or CMNL.

We provide some remarks on the structure of the log-likelihood function under the CMNL model and specify easily verifiable conditions for identifiablity of the contextual parameters. In addition to this, we empirically test the prediction performance of the CMNL model as well as its descriptive power on a relevant real data set by comparing it to several widely used choice models. Our results show that for our data set, considering context effects in choice modeling significantly enhances predictive and descriptive scores for this setting and data set. In addition to the empirical analysis of data, we develop and analyze the Assortment Optimization Problem (AOP) and an important instance of it, the Click Through Rate Optimization Problem (CTROP), building upon the CMNL model. We prove the NP-hardness of the AOP and CTROP under the CMNL. Furthermore, we identify some important special cases of the CMNL that are tractable. For instance, we show in the presence of one decoy item and when the attraction effect is dominant among the products, the AOP and CTROP are polynomially solvable. Also, we show tractability of AOP and CTROP when items are similar. Remarkably, these special cases provide high prediction performance on our real data set, making them effective for practical implementations in assortment design and product recommendation system.

We also derive some easily verifiable sufficient conditions for monotonicity and submodularity of some special cases with NP-hard CTROP and AOP, which allows us to provide some approximation guarantees. Finally, we propose a heuristic method for solving the AOP under the general CMNL model, and prove its effectiveness in finding near-optimal solutions through testing it on randomly generated instances.

In Chapter 4, we propose a new multi-choice model that can estimate the contextual interactions among the products in dense impression/click data sets. Specifically, we extend the PW-LogReg model to include the contextual interactions induced by

the presence of other products. We call this extension the "Contextual Logit" model. In our model, a customer considers selecting each of the products by comparing the utility of two options: "Selecting" and "Not Selecting" that product. The utility of each of these options can be a function of the product itself as well as the other products offered in the assortment. The other products affect the utilities through contextual interactions and in our model we assume a general interaction term to capture different types of contextual effects from complementary and substitute products. The selection probabilities are determined by using a logistic regression proportional to the utilities.

We consider different special cases of the general CL model, which can be used to model different types of interactions. Considering these cases allows us to systematically check for the type of contextual interactions which are dominant in a given data set; which consequently help in modeling the selection behavior of customers better and understanding how the presence of products in an assortment affects the each other's CTR or purchase rate.

We conduct a comprehensive empirical analysis on 70 real impression/click data sets selected from a diverse set of categories of products, and test the performance of the CL model and its special cases versus some of the benchmark models considered in recent research works. Our results indicate that using CL model and some special cases can lead to significant improvement over the prediction scores.

We consider the AOP as well as the CTROP under the CL model and show their NP-hardness. We also show these problems are polynomially solvable under some of the special cases, which provide good empirical performance in our data sets. These results show the double advantages of (i) good data fit and (ii) tractable AOP for some of the special cases over the benchmark models, in our data sets.

We show that revenue ordered assortments can perform arbitrarily bad in AOP under the general case of CL model and that the AOP's objective is not submodular. We also devise efficient heuristics, and validate their performance through a numerical study over randomly generated instances of the AOP.

In this Chapter 5, we consider the problem of retrieving Top-K objects from a superset of relevant objects when the users' behavior is affected by context effects.

Specifically, we consider a featurized extension of the CMNL model proposed in Chapter 3 (see Yousefi Maragheh et al. 2020a) to make the model applicable in IR settings. Under this model, the score of objects can be affected by other objects through interaction terms. We extend the model by assuming an underlying neural net structure that maps the features of pairs of items to contextual interaction terms among the objects. We call the model CMNL-Net.

After using this model for estimation of context effects, we investigate the Top-K retrieval problem under three different combinatorial objectives: (i) Surplus Maximization Problem, where we aim to find a subset of K objects that has the highest sum of scores, (ii) CTR Maximization, (iii) Expected Reward Maximization. We show that Top-K retrieval problem is NP-hard under all three of the objectives. However, to deal with the NP-hardness we propose binary swapping algorithms to provide high quality solutions. We test and confirm the high performance of our estimation procedure and presented swapping algorithm on real and synthetic data sets.

Chapter 2

Literature Review

In this chapter, we review some of the relevant papers to the different modeling frameworks we use. First, we review some of the well-known and relevant DCMs that are used to model choice behavior with single choice outcome assumption. Then, we proceed to reviewing the relevant modeling frameworks in marketing and cognitive sciences literature and some results from the operations management papers in multi-choice settings. Finally we review research in information retrieval literature.

2.1 Discrete Choice Modeling

Modeling choice behaviors is still an active research area in revenue management (Gallego et al. 2019). Among the proposed models, the Multinomial Logit (MNL) choice model, which is proposed by McFadden (1974), and the Basic Attraction Model (BAM) in general (which was developed axiomatically by Luce 1959) has received significant attention in modeling choice behaviors and demand dependencies among researchers. These models are derived based on the ToRC. In the BAM (and MNL), the probability of the purchase of an item is proportional to its fixed attraction value; and if an item becomes unavailable, its demand will be recaptured by other products. This capture rate of each product is also proportional to its fixed attraction value. Thus, it cannot model and measure the context effects. However, under this model, some of the revenue management problems including the AOP are tractable due to its simple structure (Talluri and Van Ryzin 2004).

The General Attraction Model (GAM) is proposed in Gallego et al. (2014) with the following motivation: The MNL model is too optimistic about the recapture rate of the demand of the unavailable items by the available items. To solve this, the GAM considers a higher probability than the MNL for the customers to leave without any purchase if more items become unavailable. In other words, the GAM assumes an increase in the perceived attractiveness of the option of "buying nothing" when an item is removed from the offered set. However, under GAM, as the perceived attractiveness of all the other items is fixed, it still suffers from the IIA property. In addition, under the GAM, it is impossible for an item to have its purchase probability

increased when new items are introduced to the offered set. Also, none of the three context effects mentioned in Section 1 are captured with GAM.

To avoid the IIA property, the Nested Logit Model (NLM) is proposed by Williams (1977) . In NLM the customers first choose a nest; then, within that nest, they choose a product. The NLM has the theme of the MNL model but in hierarchical framework: the choice of nest and the product can be assumed to be made under the MNL models. In NLM, there is a dissimilarity parameter for each nest which measures the degree of dissimilarity of products within that nest. The NLM with dissimilarity parameters larger than one can potentially model the Halo effect among the products of the same nest (Davis et al. 2014), i.e., the introduction of an item to a nest may increase purchase probability of other items. However, still, the utility of the products are independent of the assortment offered within the nests and the NLM does not incorporate the context effects.

The Mixed MNL model (MMNL), introduced in Cardell and Dunbar (1980) and Boyd and Mellman (1980), is an extension of the MNL model. In MMNL, we can have more than one customer type and each type perceives different attractiveness toward the items. This model also has an important modeling characteristic: it can approximate any discrete choice model derived from random utility maximization (McFadden and Train 2000). We will use the MMNL model as benchmark in the empirical validation section and compare its prediction power with our proposed model.

Recently, Wang (2018), following Orhun (2009), proposed a choice model, which explicitly incorporates the context dependency through defining reference points along attribute (e.g. price) dimensions. These reference points are functions of the choice set offered and the paper explores the joint assortment and pricing problem under its proposed model.

However, reference-dependent models are not able to capture all types of context effects mentioned above (Rooderkerk et al. 2011). For instance, they are not efficient in detecting the similarity effects or pair-wise decoy-target effects. In addition, these models are justified based on the theory of loss aversion which states that the customers care about unfavorable comparisons among available options more than the

favorable ones (Orhun 2009); and disadvantages affect the choice decision process more than the advantages (see Tversky and Kahneman 1991, Tversky and Simonson 1993). But, associating the loss aversion to the context effects is questioned in Trueblood et al. (2013).

A different model that can capture the synergies between the pairs of the products is proposed in Lo and Topaloglu (2019). In their model, the attraction values (not the utilities) of product-pairs can increase in additive manner. They do not validate the model empirically on a real or synthetic data set. In addition, their model does not have RUM justification and only approximates the positive effects on the attraction value. Blanchet et al. (2016) approximate utility based DCMs by assuming that substitution of one product by another in case of stock out follows a Markov chain structure.

There are other choice models, mostly in psychology and cognitive science literature, that try to capture the contextual patterns in choice (see Orhun 2009, Roodenkerk et al. 2011, Tversky and Simonson 1993 as examples). But, they are too complicated model-wise to be used in the problems like AOP of CTROP.

Our proposed CMNL model, based on the Random Utility (RU) theory postulates that a customer's utility depends on the offered assortment which allows us to approximate different kinds of pair-wise context effects. Also, we derive both complexity and approximation results for the AOP and CTROP under this model.

2.2 Multi-Choice Modeling

Now, we review the relevant papers in multi-choice settings in the choice modeling and the operations management fields.

Very recently, some papers study the AOP for models that allow multi-choice outcome. Tulabandhula et al. (2020) consider the so-called the BundleMVL model, with an underlying MVL structure where the size of the selected bundle is endogenously given. Lyu et al. (2021) extends this study to Multi-variate MNL model, where a customer selects according to MNL from sub-categories of products.

Gallego and Wang (2019) propose the Threshold Utility Model (TUM), where the customers pick the products with utilities larger than a threshold value, while restricting the size of the selected bundle. They show the NP-hardness of the AOP under TUM.

Immorlica et al. (2021) study the AOP for vertically differentiated products with rankings known to both customers and the firm and obtain some complexity results. Feldman et al. (2020) consider an extension of the MNL model where the customers can pick bundles with sizes less than a given number. Similar to works done by Xia et al. (2008) and Xia et al. (2009a), they compute the probabilities of a given ranking of products with an underlying utility structure similar to MNL. They show the NP-hardness of the AOP under this model and derive some results on approximating the optimal solution.

Note that none of the mentioned multi-choice models are capable of measuring the contextual interactions like similarity, compromise, or decoy effects on the utilities of items. These effects play a dominant role in some data sets as confirmed in our empirical studies. In addition, to capture the substitution induced by other products, it is necessary for these models to restrict the bundle size that is chosen by customers from a given offered assortment. As without this restriction, the number of model's parameters, or time needed to compute the choice selection probabilities can be exponential (see Tulabandhula et al. 2020, Feldman et al. 2020).

To the best of our knowledge, there is no paper studying AOP in multi-choice setting while capturing the contextual interaction among the items, which is what we set to do in Chapter 4.

2.3 Ranking Models and Information Retrieval

In this section, we review the relevant papers in IR settings. The learning-to-rank (LTR) methods with listwise learning objectives and their top-K variants for top-K retrieval are closely related to our work. Under the listwise methods, the entire list of objects in every query are considered as an instance of training. For instance,

ListNet, ListMLE as well as their top-K extensions proposed by Cao et al. (2007), Xia et al. (2008), and Xia et al. (2009b) respectively, use the list of objects to construct probability distribution over all the possible rankings of objects. These ranking models calculate an underlying score for each object which is merely dependent on its features and hence fail to incorporate context effects. Another related listwise approach is BoltzRank which incorporates the pair-wise dependency of documents in the relevance scoring function and learns the optimal ranking via a energy-based likelihood function Volkovs and Zemel (2009). However like the other list-wise models, its objective is to come up with the best ranking for a set of objects, rather than retrieving the best set for the Top-K retrieval problem.

Our method presented in Chapter 5 is different from these list-wise approaches as it estimates the choice probability of choosing each of K objects rather than estimating a probability distribution over the list of possible rankings. We focus on choice probability since we can utilize it later in optimizing metrics like CTR and Expected Reward of Top-K items.

In addition, our methods trains a neural net to approximate the contextual interactions among the products. Neural networks have proven to be generalizable and to be very efficient in estimation and prediction in diverse settings (see Aggarwal et al. 2018 and Tang et al. 2007). In IR settings, models like ListNet and ListMLE use neural networks to train their non-contextual score parameters and following their approach we select neural net to estimate the contextual parameters in our model.

Another point we would like to mention is the difference of context effects and contextual information. Many previous works have covered how contextual information (eg., time, place, user social network) improve existing recommender systems and search engine Adomavicius and Tuzhilin (2015), Xiang et al. (2010). While context effects and contextual information share similar terminology, there are noticeable difference between the two. Contextual informations are not induced by objects themselves, and are well studied whereas contextual effects rely on interaction between given set of documents or recommendations.

Recently, Multinomial Logit Model (MNL) and Non-parametric Choice model (first introduced in McFadden 1974 and Farias et al. 2013 respectively), are used

in the domain of e-commerce to capture substitution effect between products Feldman et al. (2018), Mao et al. (2020). Substitution effect is referred to the replacement of a less desirable object by the user (product in the case of e-commerce) as result of not offering a more desirable object. Note that under the substitution effect the inferred scores of an object still depends merely on the features of that object itself and thus, Top-K retrieval is trivial when optimizing on objectives like CTR.

Chapter 3

Discrete Choice Modeling and Assortment Optimization in the presence of Context Effects

In this chapter, we introduce a new choice model that can be used to capture the context effects in sparse choice data sets with an underlying single-choice outcome assumption. We empirically validate this model and study problems like assortment optimization and click through rate optimization for the model. The materials of this chapter are based on Yousefi Maragheh et al. (2020a) and Yousefi Maragheh et al. (2020b).

3.1 Modeling

Consider a collection $\mathcal{N} = \{1, \dots, N\}$ of products. Each member of this collection can be potentially offered for sales in a store. Each time a customer arrives to the store, a subset S of these products is offered to that customer (we can denote subset S with a binary vector $X = (x_1, x_2, \dots, x_N)$, where $x_i = 1$ if $i \in S$ and $x_i = 0$ otherwise.).

After offering the products, the customer decides either to buy one of them or to leave the store without purchasing. We can consider the no-purchase option as one of the products and assign a label zero to it. In this way, the set of choice options for the customer is $S \cup \{0\}$.

While our proposed choice model is a RUM choice model, the utility of an individual item depends on the set offered to the customers. In our model, like some of the RUM choice models (including MNL), the perceived utility of an item is modeled to consist of two parts: a deterministic part (denoted by f_i for an item i), and an additive random noise (denoted by ϵ_i).

To approximate and measure the context effects, we assume that the deterministic part of the utility of each product is linearly dependent on the offered set. Conceptually, we decompose the deterministic part of the customer's utility for item i in two components: *(i)* the *baseline utility* (i.e. the utility when all items are offered) for item i, denoted by μ_i, $i = 1, \dots, N$; and *(ii)* the effect of the absence of item j on the utility for item i, denoted by α_{ji}, $\forall i, j = 1, \dots, N$, with $\alpha_{ii} = 0$. Specifically,

the expected perceived utility of item i among the offered subset $\mathcal{S}$ is given by:

$$f_i^{\mathcal{S}} = \mu_i + \sum_{j \notin \mathcal{S}} \alpha_{ji}, \qquad i \in \mathcal{S}. \tag{3.1}$$

In equation (3.1), μ_i is a parameter that is merely dependent on product i's attributes and thus refereed to as the "baseline utility." On the other hand, α_{ji} can be dependent on the interaction between both of the products i and j and is able to capture the types of context effects discussed in the previous sections. Defining α_{ji} enables the model to explicitly account for the contextual effect of item j on item i, and thus, allows us to measure the change in perceived utility of item i as a result of the absence/presence of item j. Note that this change in the perceived utility can be due to any type of the aforementioned context effects. We call the matrix $[\alpha_{ji}]_{i,j \in \mathcal{N}}$ the Context Matrix.

Similar to the MNL model, we assume ϵ_is, $i \in \mathcal{N} \cup \{0\}$ are independent and identically distributed according to a zero-mean Gumbel distributions with the same scale parameter in the CMNL model. In addition, we normalize the utility of the no-purchase alternative to be 0 i.e. $f_0^{\mathcal{S}} = 0$.

With the above utility structure, we can obtain a closed-form probability structure for the choices which is dependent on $\mathcal{S}$. Denote the probability of choosing item j given that subset $\mathcal{S}$ is offered by $P_j(\mathcal{S})$. With a similar argument to Luce (1959), we get:

$$\begin{aligned} P_j(\mathcal{S}) &= \begin{cases} \frac{\exp\left(\mu_j + \sum_{i \notin S} \alpha_{ij}\right)}{1 + \sum_{l \in S} \exp\left(\mu_l + \sum_{i \notin S} \alpha_{il}\right)} & \text{if } j \in \mathcal{S}, \\ 0 & \text{if } j \notin \mathcal{S}, \end{cases} \\ P_0(\mathcal{S}) &= \left[1 + \sum_{l \in S} \exp\left(\mu_l + \sum_{i \notin S} \alpha_{il}\right)\right]^{-1}, \end{aligned} \tag{3.2}$$

which also can be written in terms of the binary availability vector ($x_i = 1$ if $i \in \mathcal{S}$,

$x_i = 0$ o.w.) as follows:

$$
\begin{aligned}
P_j(x) &= \frac{x_j \exp\left(\mu_j + \sum_{i=1}^{N} (1 - x_i)\, \alpha_{ij}\right)}{1 + \sum_{l=1}^{N} x_l \exp\left(\mu_l + \sum_{i=1}^{N} (1 - x_i)\, \alpha_{il}\right)}, \\
P_0(x) &= \left[1 + \sum_{l=1}^{N} x_l \exp\Big(\mu_l + \sum_{i=1}^{N} \Big(1 - x_i\Big)\alpha_{il}\Big)\right]^{-1}.
\end{aligned}
\tag{3.3}
$$

Note that this probability structure reduces to the MNL model when $\alpha_{ji} = 0$, $\forall i, j \in \mathcal{N}$.

3.1.1 Special Cases of the CMNL Model

The CMNL can potentially model different types of context effects. In this subsection, we mention special cases of the model that can be associated with these effects.

Similarity Effect and Red Bus; Blue Bus Paradox:

When two items, with indices i_1 and i_2, are similar, we can expect that they have similar contextual effects on the other items and $\alpha_{i_1 j} \approx \alpha_{i_2 j}$, $\forall j \in \mathcal{N}$. In an extreme case, when the attributes of all (or most) the products are similar to each other, we can expect to have $\alpha_{ji} \approx \alpha_i$, $\forall i, j \in \mathcal{N}$. We will refer to this special case as model $S2$, since the contextual effect of α_{ji} is dependent on the 2^{nd} index. We can also consider a special case of $S2$ where $\alpha_i = \theta$, $\forall i \in \mathcal{N}$, which we will be referring to as model "$C\theta$". In this case, all the contextual effects among the pairs of the products are equal. This may become useful when trying to approximate the CMNL in small data sets with smaller number of parameters and avoid overfitting the data.

We can justify the "Red Bus; Blue Bus" paradox (see Gallego et al. 2014), using the CMNL model. This paradox is normally used to explain the IIA property and how MNL model is suffering from it. According to this paradox, a person has to decide between using a car or taking either a red or blue bus as her transportation mode. It is assumed that both of the buses are exactly the same except for the color

(i.e. they have sufficient capacity, they leave at the same time, travel with the same speed etc.). Let the utility of driving the car be u_c and utilities of taking blue bus and red bus be u_{bb} and u_{rb} respectively. We can expect $u_{bb} = u_{rb}$ as they are similar. Under the MNL, with $u_0 = 0$, the probability of driving a car when there is only blue bus with ample capacity is $\exp(u_c)/(1+\exp(u_c)+\exp(u_b))$ and adding a red bus decreases this probability to $\exp(u_c)/(1+\exp(u_c)+2\exp(u_b))$. However, since both buses have ample capacity, one would expect the probability does not change. We can address this paradox using the CMNL model and setting sum of the attraction values of the busses when both of them are available, $\exp(\mu_{bb})+\exp(\mu_{rb})$, equals the attraction value of one bus when only one of them is available, $\exp(\mu_{bb}+\alpha_{rb,bb})$ or $\exp(\mu_{rb}+\alpha_{bb,rb})$:

$$\exp(\mu_{bb})+\exp(\mu_{rb}) = \exp(\mu_{bb}+\alpha_{rb,bb}) = \exp(\mu_{rb}+\alpha_{bb,rb}),$$

which results in the following pairwise context effect parameters (by setting $\mu_b = \mu_{bb} = \mu_{rb}$):

$$\begin{cases} \alpha_{bb,rb} = \alpha_{rb,bb} = \ln(2\exp(\mu_b)) - \mu_b, \\ \alpha_{rb,c} = \alpha_{bb,c} = \alpha_{c,rb} = \alpha_{c,bb} = 0. \end{cases}$$

With the above parameters, we will not see a decrease in probability of choosing a car when adding an additional bus.

Attraction/Decoy and Compromise Effects:

If an item, i_1, is perceived as a decoy for a target item i_2, by assuming $\alpha_{i_1,i_2} \ll 0$, the CMNL model may capture the positive effect of adding item i_1 beside item i_2.

Also, if the compromise effect is the dominant type of context effect among a given set of products, we expect adding a luxury item i_3 may boost the perceived utility of a lower option i_4 and by assuming $\alpha_{i_3,i_4} < 0$, the CMNL model may approximate this when the variation in the offered assortment is limited.

In addition to the above examples, consider an stable market that customers are aware of all the product options and their features and we may be interested in evaluating the effect of adding a new item, i, to this market. In this case, the CMNL model with only the i^{th} row of Context Matrix being non-zero can help us

approximate the effect of item i on other items. We refer to this special case as P_i in the rest of the paper.

Synergistic and Antagonistic Products:

The CMNL model may describe the synergistic or antagonistic effect of products on each other. By synergy, we mean the positive effect that offering one product may have on the other. For instance, a decoy item has a positive effect on a target item. The effect of products on each other can be antagonistic, like the effects that red bus and blue bus have on each other. Synergy or antagony, i.e. positive or negative effect on the perceived attractiveness of products, can exist due to many other reasons (see Feng et al. 2018 and Rooderkerk et al. 2011 for more examples). Under the CMNL model setting $\alpha_{ji} < 0$ or $\alpha_{ji} > 0$ can potentially describe the synergistic or antagonistic effect of offering item j beside item i on the utility of item i. We will denote the synergistic version of each model by adding $-Syn$ in the rest of this chapter.

3.1.2 Click Through Rate Maximization and Assortment Optimization Problems

Given the CMNL model, we are interested in the Assortment Optimization Problem, i.e., finding the subset of products (assortment) which has the highest expected revenue among all the possible assortments. For a fixed assortment S (with binary vector equivalent of x), let us denote the expected revenue obtained by offering subset S of products by $r(S, \mu, A)$, when the underlying parameters are specified by vector $\mu = (\mu_1, \ldots, \mu_N)$ and context matrix $A = [\alpha_{ji}]_{j,i\in\mathcal{N}}$.

The AOP is the problem concerned with finding the set S which maximizes $r(S, \mu, A)$. Formally, the AOP can be formulated as the following optimization problem:

$$\max_{x\in\{0,1\}^N} \sum_{i=1}^{N} x_i r_i P_i(x), \tag{3.4}$$

where $P_i(x)$ is defined in (3.3) and r_i is the revenue obtained by selling item i, $\forall i \in \mathcal{N}$.

Click Through Rate Optimization Problem is an important instance of the AOP where $r_i = 1$, $\forall i \in \mathcal{N}$. In this instance, we maximize the probability of purchase from the offered set S, or the Click Through Rate (CTR) of S, which we denote by $M(S, \mu, A)$. This is equivalent to minimizing the chance of loosing the customer to the no-purchase alternative. The following formulates the CTROP:

$$\max_{x \in \{0,1\}^N} \sum_{i \in \mathcal{N}} x_i P_i(x). \tag{3.5}$$

In online shopping platforms, one of the widely considered goals is to maximize the CTR of the recommended items (or ads) to the customer and there is a considerable amount of evidence that optimizing CTR help in maximizing the revenues of the online recommendation platforms and search engines (Richardson et al. 2007). Problem (3.5) can be interpreted as the problem of optimizing the Market Share of the company as well. The market share is an important index since higher market share can lead to economies of scale, market power, strategic advantages over the rival companies, and even higher rate of return of investment (Buzzell et al. 1975).

In the following theorem, we will show that the CTROP is an NP-hard problem. To show this (see Appendix A for the proof), we prove the NP-hardness of a special case of the CMNL with $\alpha_{ji} = \alpha_j$, $i, j \in \mathcal{N}$. We will refer to this special case as model $S1$, since the contextual effect of α_{ji} is dependent on the 1^{st} index . For all the theorems, propositions, and lemmas, the proofs not presented in the main body of the paper can be found in the Appendix A.

Theorem 3.1. *When the underlying probability structure is CMNL, the CTROP* (3.5) *is NP-hard.*

An immediate corollary from the above theorem is the NP-hardness of the AOP under the CMNL model.

3.2 Parameter Estimation

The goal in this section is to describe an estimation procedure and discuss identifiability conditions for the model parameters μ_j, α_{ij}, for $i, j = 1, \ldots, N$. First, let us define the matrix containing all parameters to estimate, Θ, as follows:

$$\Theta := \begin{bmatrix} \mu_1 & \mu_2 & \ldots & \mu_{N-1} & \mu_N \\ 0 & \alpha_{12} & \ldots & \alpha_{1,N-1} & \alpha_{1N} \\ \alpha_{21} & 0 & \ddots & \ldots & \vdots \\ \alpha_{31} & \alpha_{32} & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & 0 & \alpha_{N-1,N} \\ \alpha_{N1} & \alpha_{N2} & \ldots & \alpha_{N,N-1} & 0 \end{bmatrix} = \begin{bmatrix} \boldsymbol{\mu} \\ \mathbb{A} \end{bmatrix}, \tag{3.6}$$

where $\boldsymbol{\mu}$ is the vector containing μ_j, $j = 1, \ldots, N$ and $\mathbb{A} = [\alpha_{ij}]$, with $i, j = 1, \ldots, N$.

3.2.1 Likelihood Function

In this subsection we define the log-likelihood function for a given impression/choice data. Note that this likelihood function can be used to obtain the maximum likelihood estimators. In order to define the likelihood function for some given data with varying availability of items, consider a collection of $\mathcal{N} = \{1, \ldots, N\}$ of products to be sold over M browsing sessions (periods)[1]. Also, denote by $\mathcal{S}_m \subset \mathcal{N}$ the set of items offered (impressed) to the customer in a given period $m \in M$ which can be equivalently shown as a binary vector that is dependent on the period m: $x^{(m)} = \{x_1^{(m)}, x_2^{(m)}, \ldots, x_N^{(m)}\}$ where $x_i^{(m)} = 1$ if $i \in \mathcal{S}_m$ and $x_i^{(m)} = 0$ o.w. We assume that, due to limited availability or strategic decisions for example, not all products are available on a given period $m \in M$, which implies that for different periods the assortment sets $\mathcal{S}_m$ may be different.

Our observation during each period m includes the offered set and the choice decisions; i.e. binary variable associated with choice of different products in period

[1]We use periods and browsing session interchangeably in the rest of this thesis.

m, denoted by vector $z^{(m)} = \{z_i^{(m)}\}_{i\in\mathcal{N}}$, with $z_i^{(m)} = 1$ iff i is chosen in browsing session m, and $z_i^{(m)} = 0$ o.w. If $i \notin \mathcal{S}_m$ then trivially $z_i^{(m)} = 0$. The aggregate number of times that item i is chosen in all periods is simply denoted by $z_i = \sum_{m\in M} z_i^{(m)}$.

By having the above notation, the likelihood function can be represented as follows:

$$L(\mu, \mathbb{A}) = \prod_{m=1}^{M} \left(P_0(\mathcal{S}_m)\right)^{z_0^{(m)}} \left(\prod_{j\in\mathcal{S}_m} \left(P_j(\mathcal{S}_m)\right)^{z_j^{(m)}}\right). \tag{3.7}$$

If we take the logarithm and plug-in the expression (3.2), for the probabilities $P_j(\mathcal{S}_m)$ we obtain the following log-likelihood:

$$\begin{aligned}
\ell(\mu, \mathbb{A}) &:= \log L(\mu, \mathbb{A}) \\
&= \sum_{m=1}^{M} \left(z_0^{(m)} \ \log P_0(\mathcal{S}_m) + \sum_{j\in S_m} z_j^{(m)} \log P_j(\mathcal{S}_m) \right) \\
&= \sum_{m=1}^{M} \Bigg\{ - \kappa^{(m)} \log\Big(1 + \sum_{k\in S_m} \exp\Big(\mu_k + \sum_{i\notin S_m} \alpha_{ik}\Big)\Big) \\
&\qquad\qquad + \sum_{j\in S_m} z_j^{(m)} \Big(\mu_j + \sum_{i\notin S_m} \alpha_{ij}\Big) \Bigg\},
\end{aligned} \tag{3.8}$$

where we set $\kappa^{(m)} := \sum_{j\in S_m \bigcup\{0\}} z_j^{(m)}$. Taking the derivatives with respect to the parameters,

$$\frac{\partial \ell(\mu, \mathbb{A})}{\partial \mu_p} = 0, \quad \frac{\partial \ell(\mu, \mathbb{A})}{\partial \alpha_{qp}} = 0, \ \ \forall q, p = 1, \ldots, N, \ \ q \neq p,$$

we obtain a complicated system of equations. For example, when taking derivative w.r.t parameter μ_p, $p = 1, \ldots, N$ we get:

$$\sum_{m=1}^{M} \Big\{ z_p^{(m)} (1 + \sum_{k\in\mathcal{S}_m} \exp(\mu_k + \sum_{i\notin\mathcal{S}_m} \alpha_{ik})) - \kappa^{(m)} x_p^{(m)} \exp(\mu_p + \sum_{i\notin\mathcal{S}_m} \alpha_{ip}) \Big\} = 0 \tag{3.9}$$

As one would expect, (3.9) heavily depends on the structure of the offered sets $\mathcal{S}_m$,

$m \in M$. In order to better understand the dependence of the likelihood on the offered sets, we define an intermediate matrix $\mathbb{Q}$ that specifies the relationship between the parameters and the offered sets. Specifically, the elements of the $\mathbb{Q}$-matrix indicate the presence or absence of an item in the offered set in period m, with $q_{mj} = 1$, if $j \in \mathcal{S}_m$ and 0 if $j \notin \mathcal{S}_m$:

$$\mathbb{Q} := \begin{bmatrix} q_{11} & q_{12} & \cdots & q_{1N} \\ q_{21} & q_{22} & \cdots & q_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ q_{M1} & q_{M2} & \cdots & q_{MN} \end{bmatrix}. \tag{3.10}$$

Therefore, the derivative of the log-likelihood w.r.t. μ_p can be written as:

$$\ell^*(^-, \mathbb{A}) = \sum_{m=1}^{M} \Bigg\{ z_p^{(m)} \Bigg(1 + \sum_{k=1}^{N} q_{mk} \exp\Big(\mu_k \ + \sum_{i=1}^{N} \alpha_{ik} \, (1 - q_{mi}) \Big) \Bigg) \\ - \kappa^{(m)} q_{mp} \exp\Big(\mu_p + \sum_{i=1}^{N} \alpha_{ip} \, (1 - q_{mi}) \Big) \Bigg\} = 0.$$

3.2.2 Identifiability Conditions

Due to the nature of the data and the structure of the $\mathbb{Q}$-matrix, it is not always possible to uniquely identify all the model parameters. The most natural example is when, for example, item i is always in the offer set. Then, we are not able to observe its effect on the other items when missing, which is quantified via parameter $\alpha_{ij}, j = 1, \ldots, i-1, i+1, \ldots, N$ and i fixed . This, of course, does not mean that there is not an effect, but our data do not allow us to estimate it.

However, in practice, as it is also confirmed by our empirical validation, this is not an important restraining issue. The actual data used for training in e-commerce platform present a very diverse impression set, and the identifiablity conditions are generally satisfied in practice. We present this section for the sake of complicity.

Mathematically, identifiability is defined as follows:

Definition 3.1. *A set of parameters Θ is identifiable if the following holds:*

$$P_j^{(m)} = P\left(Z = z^{(m)}|\mathbb{Q}, \Theta\right) = P_j^{(m)} = P\left(Z = z^{(m)}|\mathbb{Q}, \bar{\Theta}\right) \Leftrightarrow \Theta = \bar{\Theta}, \tag{3.11}$$

where $z^{(m)}$ are the data in a period m, Θ is the parameter matrix and $\mathbb{Q}$ is the matrix that indicates the presence/absence of an item in the offer set.

The goal in this section is not to attack this question in its greater generality. However, we want to provide *sufficient identifiability conditions* that will guarantee the uniqueness of the solution of (3.11). Therefore, consider the two following cases:

(C1) Assume that the $\mathbb{Q}$-matrix takes the following form after row-swapping:

$$\mathbb{Q} = \begin{bmatrix} \mathbb{1} \\ \mathbb{1} \\ \mathbb{1} - \mathbb{I} \\ \mathbb{1} - \mathbb{I} \\ \mathbb{Q}' \end{bmatrix}.$$

where $\mathbb{1}$ is a matrix which all elements of it are one; $\mathbb{I}$ is the identity matrix; and $\mathbb{Q}'$ has a similar structure as its upper rows of $\mathbb{Q}$, i.e. combination of matrices of type $\mathbb{1}$ and $\mathbb{1} - \mathbb{I}$.

(C2) Suppose $\mathbb{Q}$ has the following structure after row-swapping:

$$\mathbb{Q} = \begin{bmatrix} \mathbb{1} \\ \mathbb{1} \\ \mathbb{U} \\ \mathbb{U} \\ \mathbb{Q}' \end{bmatrix},$$

where $\mathbb{1}$ is as defined previously; $\mathbb{U}$ is an upper-triangular matrix of ones and $\mathbb{Q}'$ has a similar structure as upper rows of $\mathbb{Q}$, i.e. combination of matrices of type $\mathbb{1}$ and $\mathbb{U}$.

Remarks: Condition **(C1)** is slightly stronger than it is necessary, but we need to guarantee that each subset of products appears at least twice so that we have enough information to identify the model parameters. Condition **(C1)** is interpreted as having periods where all items are offered and periods where only one item is missing. In addition, every item must be missing in at least one period. Condition **(C2)** is interpreted as having periods where all items are offered and then periods where sequentially items are not in the presented offer sets. For example, that would mean that after period m_1, item i is out-of stock, so it will not be offered in the subsequent period either.

Theorem 3.2. *Under the CMNL model with utility and probability structure defined in* (3.1) *and* (3.2) *respectively, if condition* ***(C1)*** *is satisfied, then* Θ *is identifiable.*

The above theorem gives us the conditions on the structure of the data for identifiability of all of the parameters. In practice, we may be interested in estimating a subset of parameters. For example, consider the case that we restrict the parameter space such that $\alpha_{ij} = 0$ when $i \leq j$. Interestingly, when products never appear again in the offered subset after being out of stock the likelihood function of the data only include α_{ij} parameters with $i < j$ (note that we may need to relabel the products). This creates a specific structure of the $\mathbb{Q}$ matrix which allows us to obtain the following sufficient conditions for identifiability:

Theorem 3.3. *Under the CMNL model with utility and probability structure defined in* (3.1) *and* (3.2) *respectively, when the matrix* $\mathbb{A}$ *is upper triangular and the* $\mathbb{Q}$ *matrix satisfies Condition* ***(C2)****, then the model is identifiable.*

A more general criterion for parameter identifiability has been introduced in the literature, that of *local identifiability*. According to it, in a local region of the parameter space, there is a unique θ_0 that fits some specified body of data. More formally,

Definition 3.2. *A set of parameters* Θ *is locally identifiable, if there exists a neighborhood* $\tilde{\Theta} \subset \Theta$ *such that* (3.11) *is satisfied.*

Proposition 3.1. *Under the CMNL model, when the $\mathbb{Q}$ matrix satisfies Condition **(C2)**, then the model is locally identifiable.*

In practice, this implies that by using a subset of the data that satisfies the desirable structure of the $\mathbb{Q}$ matrix, we are able to uniquely estimate the model parameters.

3.3 Empirical Studies

In this section, we test the performance of our proposed model and some of its special cases on a real data set. We compare the CMNL model and several special cases with four benchmark models: the MNL, the GAM, the MMNL with two customer types, and MMNL with three customer types; denoted by "MNL", "GAM", "MMNL2", and "MMNL3" respectively in the tables included in this section. For better reference, we present a short description of the CMNL model and its special cases in Table 3.1.

Table 3.1: Special Cases of the CMNL Model

The Model	Parameter Restriction	Short Description or Interpretation
CMNL-Syn	$\alpha_{ji} \leq 0$	CMNL with only synergistic effects
S1	$\alpha_{ji} = \alpha_j$	Special case with NP-hard AOP, an approximation of the full model
S1-Syn	$\alpha_{ji} = \alpha_j \leq 0$	Synergistic version of S1
S2	$\alpha_{ji} = \alpha_i$	Can work well when items are similar, utility being a linear function of cardinality of assortment, cardinality affecting different items with different slopes
Cθ	$\alpha_{ji} = \theta$	Can work well when items are similar, utility being a linear function of cardinality of assortment, cardinality having same effect on the utility of all items
P_i	$\alpha_{jl} = 0, \forall j \neq i$	Only item i is has effect on other items' utilities, item i is a decoy item

For comparing the models, first, we compute the AIC and BIC scores for each of

them. These scores are likelihood-based criteria that penalize the models for having more number of parameters. For testing the prediction capability of the models, we randomly divide the data into training and testing parts and obtain the Maximum Likelihood Estimators (MLE) of the models in training sets. Then, we compare the log-likelihood of the test set for each model. In addition, in order to see how well the models predict the choice probability vectors, we perform a χ^2 test for multinomial fit in the test sets and for each of the assortments sets. Using these tests are common practice in the literature (see Read and Cressie 2012, Wang 2018, and Zhang and Sabuncu 2018 for more examples and description). The presented results illustrate the superiority of our choice models in predicting and describing choice outcomes in our data set.

3.3.1 The Data

For empirical validation, the click and impression data for four of the highly clicked items in the "Tent" category are obtained from the data for user interaction with one of the product recommender system modules of Walmart.com. More specifically, we obtain the impressed (shown) as well as clicked items in the "Customers also considered" module under one of the item pages. "Customers also considered" module in every item page of Walmart.com tries to recommend products that can be a substitute for each other. See Figure 3.1 for an example of this module(This figure is presented as an illustration of the module and the presented items in this figure are not the items of our data set).

The data includes 14,237 sessions of impression and click data for these popular items from March 20^{th}, 2020 till July 31^{st}, 2020. In each session, a subset of the products is offered to a customer and she either clicks on the items from the offered assortment or leaves the item page without clicking. Note that the cardinality of the assortment can be different depending on the size of the screen of the device and the platform that customer uses (PC, Walmart app, Web-mobile).

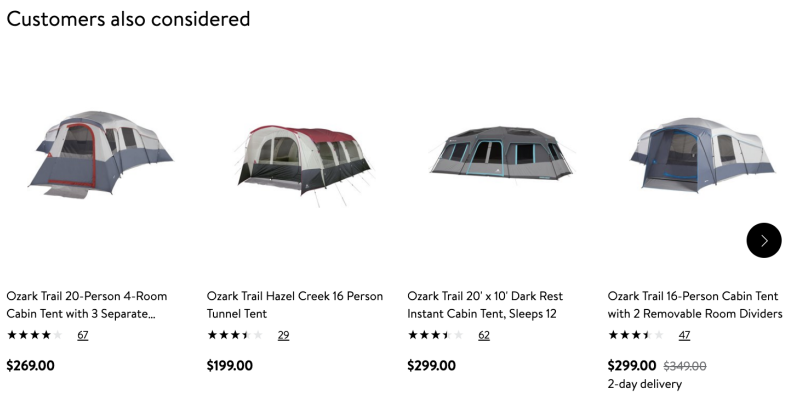

Figure 3.1: An example of recommended items in Customers-also-considered section

3.3.2 Testing the Goodness of Fit

For all the models, the MLE Estimates are computed. Then, using these estimates, the log-likelihood of the data is computed using the Branch-And-Reduce Optimization Navigator (BARON) solver version 20.10.16 (Sahinidis 2017, Tawarmalani and Sahinidis 2005) under Pyomo package in Python (Hart et al. 2011, 2017) with default terminating conditions on a PC with an 2.4 GHz 8-Core Intel Core i9 processor and 64 GB 2667 MHz DDR4 memory operating on macOS (BARON is a computational system for finding the global solution of non-linear non-convex programs). In all models, the attraction value of one of the "no-click" is normalized to 1 (or equivalently the utility is normalized to 0) to get unique MLEs.

As we fit different models with different number of parameters, measuring their descriptive power based on the log-likelihood score is not fair and informative. Because of this, we compare the models based on their Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC) scores defined as follows (Burnham and Anderson 2002):

$$\begin{aligned} \text{AIC} &= -2\log\hat{L} + 2d, \\ \text{BIC} &= -2\log\hat{L} + d\log T, \end{aligned} \tag{3.12}$$

where $\hat{L}$ is the maximum of the likelihood function under the model of consideration, d is the number of parameters to be estimated and T is the number of transactions of the data (14,237 for this data set). The AIC and the BIC are penalized log-likelihood criteria and they are increasing in the number of parameters of a model. Note that by definition a better model has lower AIC and BIC scores. To have a clear and concise presentation, we report the best score among the "P_i" models (P_3 for our data set). The results are in Table 3.2.

Table 3.2: AIC and BIC scores for the CMNL model, its special cases, and the benchmark DCMs

The Model	AIC	vs. MNL	BIC	vs. MNL
MNL	19,707.88		19,738.13	
GAM	19,715.88	-0.04%	19,776.39	-0.19%
MMNL2	18,942.56	3.88%	19,010.64	3.69%
MMNL3	18,952.56	3.83%	19,058.45	3.44%
CMNL	**18,705.01**	**5.09%**	**18,803.34**	**4.74%**
CMNL-Syn	19,725.88	-0.09%	19,824.20	-0.44%
S1	18,787.09	4.67%	18,840.04	4.55%
S1-Syn	19,713.88	-0.03%	19,766.82	-0.15%
S2	**18,709.54**	**5.07%**	**18,770.05**	**4.90%**
Cθ	18,802.06	4.60%	18,839.88	4.55%
P_3	19032.97	0.03%	19153.98	0.03%

Note that the AIC and BIC scores are likelihood-related scores that are in logarithmic scales and the difference between scores of two models denotes how much one model is better on a logarithmic scale. Thus, even small differences in the scores can be interpreted as a relatively large difference when investigating the likelihood of the data under a given set of models.

From the above tables, we note that the CMNL has the best in the AIC score and $S2$ performs the best in the BIC score. BIC score penalizes the extra number of parameters more than the AIC score. For this reason, models with a smaller number of parameters tend to perform better in this score.

We can divide the above models into (i) models with polynomially solvable AOPs and (ii) NP-hard AOPs. "MNL" (Talluri and Van Ryzin 2004), "GAM" (Davis

et al. 2013), as well as special cases S2, Cθ, and P$_i$ have polynomially solvable AOP (Section 5), while "MMNL2" and "MMNL3" (Rusmevichientong et al. 2014), CMNL, and the special case S1 have NP-hard AOPs. Note that $S2$ has the best BIC while having a polynomially solvable AOP. Also, Cθ performs better than all the benchmark models in both scores while just having one more parameter than the MNL model.

As will be confirmed in comparing other predictive scores of the models, generally S2 (alongside Cθ) is working well for this data set. According to interpretation of special cases presented in Section 4.1, we expect S2 and Cθ to work well in data sets with similar products and interestingly most of the products of this data set are similar to each other. More specifically, three out of four of these products have similar (within 10-15%) features (average rating, size/price of the tent that can be observed in the "Customers also considered" module). We think this may be the reason for the high performance of S2. This observation for this data set may be extrapolated to other data that have similar products.

3.3.3 Testing Prediction Accuracy

In this subsection, we compare the prediction accuracy of different models. For this purpose, we randomly split the data into training and testing subsets with equal sizes. A total of generate 50 random splits are considered.

To evaluate each model's prediction performance, we obtain the models' MLEs for each training set and perform the tests described in the following.

Cross Entropy Score or log-Likelihoods of the Test sets: After fitting all the models in different training sets and finding the MLE estimates for the models, we first calculate the log-likelihood of the associated testing set under different models. This is called cross entropy score in the Machine Learning literature and it is normally used as a performance metric in likelihood-based models (Zhang and Sabuncu 2018). This metric gives the likelihood of the click patterns in the test set given the estimated value of parameters from the training set for each model. A likelier model means that the model gives us a better prediction about the click outcomes and has a higher

cross entropy score. Again like the AIC and BIC scores this score is a log-scale score, and a small increase in the cross entropy when comparing with the benchmark models can be inferred as a significant improvement in the prediction performance. The results of this score are reported in Table 3.3.

χ^2 Test for Multinomial Fit: In addition to the cross entropy score, we report the prediction performance of the models in the χ^2 test for multinomial fit (Read and Cressie 2012). Using these MLEs for different models obtained from training sets, we calculate the predicted probabilities of purchase of products for different of impressed(shown) assortments, and then contrast them to the observed purchase shares of the products in the testing sets by calculating the χ^2 test statistic for different models.

In our data, there are only seven types of impression sets and thus, in each train-test splits, and for each model, we perform seven χ^2 tests of multinomial fit. In each test, the null hypothesis is clicked probabilities being generated according to the underlying model. If the predicted probabilities by a model are significantly different than the actually observed click probabilities, then we reject the null hypothesis. The statistical significance of the test is reflected in the P-value and we reject the null hypothesis if the P-value is less than 0.05. Table 3.3 contains the average number of rejections for each model in different train-test splits.

According to the results CMNL model as well as S2 obtain the best average cross entropy scores and least average number of rejection. These two models outperform all the benchmark models in this data set and improve the cross entropy score by 5.01% when compared with the baseline "MNL" model. In addition, CMNL model and S2 decrease the average number of rejections by 39.81% and 42.28% respectively compared to MNL model. Other than these two models, Cθ performs relatively well despite having just one extra parameter than the MNL model.

One other observation is the slightly better performance of the "MMNL2" model when compared to the "MMNL3" model. This may happen due to the overfitting in the "MMNL3" model given the training sets' size in our data set and we may need larger data in order to fit a mixed model with three or more classes.

Note that in all of the tests, the GAM performs very similar to the MNL model.

Table 3.3: Average of cross entropy and number of rejections of CMNL model, special cases of the CMNL, and the benchmark DCMs in Test Sets for difference train-test splits

	Cross Entropy	vs. MNL	# Reject	vs. MNL
MNL	-4,931.16		6.48	
GAM	-4,931.15	0.00 %	6.48	0.00%
MMNL2	-4,730.87	4.06%	4.02	37.96%
MMNL3	-4,731.61	4.05%	4.02	37.96%
CMNL	**-4,684.00**	**5.01**%	**3.90**	**39.81**%
CMNL-Syn	-4,931.15	0.0%	6.48	0.00%
S1	-4,702.36	4.64%	4.82	25.62%
S1-Syn	-4,931.15	0.00%	6.48	0.00%
S2	**-4,684.26**	**5.01**%	**3.74**	**42.28**%
Cθ	-4,705.76	4.57%	4.90	24.38%
P_3	-4,758.91	1.97%	6.60	-0.02%

The main incentive for proposing GAM was the idea of "MNL being too optimistic about the recapture rate by other models" (Gallego et al. 2014). In other words, GAM can be justified when we lose more customer fractions to the no-click (no-purchase) alternative than the MNL model according to this justification. In contrast, we observe the other way around in our data analysis, and we see that MNL is in fact pessimistic about the recapture rate. This is also related to the worse performance of the synergistic contextual models than the general contextual models. When fitting the CMNL model, we observe that most of the pairwise α_{ji} parameters are positive and actually adding the items to the assortment generally decreases the utility of the other existing items.

This low performance of the synergistic models and GAM can be further justified by paying attention to the products' features in our data set. As stated, most of the products have similar feature values in our data set. When eliminating an item which is similar to the other items in a given assortment, we expect that this elimination does not decrease the no-click (no-purchase) probability significantly (exactly like the "Red Bus; Blue Bus" example where eliminating the blue bus while retaining the red bus does not change the probability of choosing nothing). In other words,

since eliminating one (or more) of the similar items does not decrease the variety of the assortment very much, or even it may help in eliminating the clutter and helping the customer to decide to click (Boatwright and Nunes 2001, Kalyanam et al. 2007), we may expect the MNL model having pessimistic estimate of the recapture rate for our data set. Also, with these similar features of products, one can hardly justify existence of synergy among the pair of products. This is due to the fact that when adding similar items, we expect they hurt each others' click shares rather than having constructive and positive effects on each other (Tversky 1972).

Another observation we make from the scores is the low performance of the P_i models. As discussed in previous sections, we expect these models to have high performance when having decoy-target type patterns or the context effects caused by one specific item being more significant than context effects caused by other items. However, by checking the product features in our data set, we notice none of them are dominated by the others feature-wise. Consequently, none of them act as decoy item, and hence, we do not expect these special cases to perform well due to decoy effect.

In summary, our empirical result provides strong support for the importance of context effects in predicting the choice outcomes and fitting models.

3.4 Polynomially Solvable Special Cases of the CMNL Model

In this section, we consider AOP and explore the special cases of the CMNL model under which the AOP is tractable. In particular, the AOP (and consequently CTROP) is tractable when (i) the underlying model is "P_i"; (ii) the revenues of items are similar in synergistic version of "$S1$" (Recall that in the previous section we proved the NP-hardness of AOP under "$S1$" when there is no restriction on the revenues); (iii) and the underlying model is S2.

In Section 4, we already have shown the high prediction performance of some of these special cases by testing on a real data set. Thus, some of these cases enjoy both high prediction accuracy when fitting in a real data set as well as the computational

tractability for revenue management applications.

3.4.1 Presence of Attraction Effect

Recall that under the special case P_i, i^{th} item can be an item which has the only significant context effect on others. For instance, it can be a decoy item which is dominated by just the target item to make it look more preferable when compared to other items (Choplin and Hummel 2005). Naturally, in this case, the effect of presence/absence of the decoy item on the utility of other items is much more significant than the effect of presence/absence of other items. We have the following result for the AOP when the underlying model is P_i.

Proposition 3.2. *The AOP (and consequently the CTROP) under "P_i", $i \in \mathcal{N}$, is polynomially solvable.*

In the proof of Proposition 3.2, we show that the AOP can be solved by considering two sub-problems and dependent on $x_i = 0$ or $x_i = 1$, the problem is reduced to the AOP under the MNL model or a slight variation of AOP under the MNL model both of which are polynomially solvable.

3.4.2 Synergistic S1 with Similar Revenues

As stated in Section 2, the AOP is NP-hard under the special case S1 where $\alpha_{ji} = \alpha_j$, $\forall i, j \in \mathcal{N}$. However, in this section, we show that the AOP is tractable when all the contextual effects are synergistic, i.e. $\alpha_j \leq 0$ and when the revenues of products are close enough. The following proposition states a condition under which the optimal solution for the AOP equals $\mathcal{N}$.

Proposition 3.3. *Under S1-Syn, if $r(S, \mu, A) \leq r_i$, $\forall S \in 2^{\mathcal{N}}$ and $i \notin S$, then $\mathcal{N}$ is an optimal assortment.*

The condition of the above proposition is hard to interpret and verify. In the following two lemmas, we provide an easily verifiable condition as a special case.

Lemma 3.1. *Assume* $r_{\max} = \max_{i\in\mathcal{N}} r_i$ *and* $r_{\min} = \min_{i\in\mathcal{N}} r_i$. *Under the S1-Syn, if the revenues of the products are close to each other such that*

$$\frac{r_{\max} - r_{\min}}{r_{\min}} \leq \min_{S\subseteq\mathcal{N}} \frac{P_0(S)}{1 - P_0(S)}, \tag{3.13}$$

then $r(S, \mu, A) \leq r_i$, $\forall S \in 2^{\mathcal{N}}$ *and* $i \notin S$.

The above lemma states that if the revenues of the products are not too different from each other, or if the chance of not purchasing a product is large enough then the condition of Proposition 2 holds. We expect condition (3.13) to hold in some business settings. For instance, in the online advertisement, the offered links can be considered as products. In this case, the advertiser obtains a revenue if a customer clicks on an advertisement link. The revenues obtained from links are normally similar and the probability of clicking any link is small compared to the probability of the not clicking alternative. In competitive markets in general we expect the probability of the no purchase alternative to be high, and if we consider profits instead of revenues, the relative difference is typically low (Han et al. 2019).

Note that in the CTROP, $M(S, \mu, A) \leq \min\{r_i\}_{i\in\mathcal{N}} = 1$, as $M(S, \mu, A)$ is sum of the probabilities. Thus, we have the following corollary.

Corollary 3.1. *Under S1-Syn of the CMNL model, the optimal assortment for the CTROP with no constraint is* $\mathcal{N}$.

Now, in the following lemma, we provide an easily verifiable sufficient condition for $r(S, \mu, A) \leq r_i$, which can be verified with $O(n)$ comparisons.

Lemma 3.2. *Under S1-Syn of the CMNL model, if* $r(\mathcal{N}, \mu, A) \leq \min\{r_i\}_{i\in\mathcal{N}}$, *then* $r(S, \mu, A) \leq r_i$, $\forall S \in 2^{\mathcal{N}}$ *and* $i \notin S$.

One thing which is worth noting is the similarity between S1-Syn and the GAM. Under GAM (Gallego et al. 2014), eliminating an item j from the assortment, increases the perceived attractiveness of the no-purchase alternative by w_j in an additive manner ($1 \longrightarrow 1 + w_j$), and the probability of choosing item $i(\in S)$ under this

model is

$$P_i(S) = \frac{v_i}{1 + \sum_{j \notin S} w_j + \sum_{j \in S} v_j},$$

where v_i is the attraction value of item i. This probability structure is very similar to that of S1-Syn. As presented in the proof of Proposition 3.3 (see Appendix A for the proof) the probability under this special case can be rewritten as follows:

$$P_i(S) = \frac{v_j}{e^{\sum_{j \notin S} -\alpha_j} + \sum_{j \in S} v_j},$$

where $v_j = e^{\mu_j}, \forall j \in \mathcal{N}$. As $\alpha_j \leq 0$ in S1-Syn, eliminating item j from the assortment S increases the attraction value of the no-purchase alternative similar to GAM, but in a multiplicative manner ($1 \longrightarrow 1 \times e^{-\alpha_j}$) and not in an additive manner.

3.4.3 Linear Dependence of Utility on the Cardinality of Assortment Presented S2:

As shown in Section 3, we can justify the similarity effect using a special case of the CMNL model with $\alpha_{ji} = \alpha_i$, $\forall i, j \in \mathcal{N}$, which we refer to as special case S2. This can be related to the case where the pairwise effect of elimination of other items on a given item depend merely on the item itself. In fact, S2 is equivalent to the case where the utility of products are linearly dependent on the cardinality of assortment presented.

Indeed, when $\alpha_{ji} = \alpha_i \ \forall i, j \in \mathcal{N}$, the utility of each item is as follows:

$$\begin{aligned} u_i &= \mu_i + \sum_{j \in \mathcal{N}} \alpha_i (1 - x_j) \\ &= \mu_i + N\alpha_i - \alpha_i \sum_{j \in \mathcal{N}} x_j \qquad \forall i \in \mathcal{N}, \end{aligned}$$

which is linear in $\sum_{j \in \mathcal{N}} x_j$ or the cardinality of assortment. In addition, recall that Cθ is the special case of S2 with $\alpha_i = \theta$, $\forall i, j \in \mathcal{N}$. Similarly, C$\theta$ is related to the case where the utility of the products is linearly dependent on the cardinality of

assortments with the same slope coefficient:

$$u_i = \mu_i + \theta(N - \sum_{j \in \mathcal{N}} x_j).$$

As stated in the following proposition, the AOP when the underlying probability structure is S2 (or Cθ) is polynomially solvable.

Proposition 3.4. *The AOP (and consequently the CTROP) is polynomially solvable under S2 of the CMNL model.*

In the proof of Proposition 3.4, we show that AOP can be solved by considering $\mathcal{N}$ polynomially solvable sub-problems. Each of these sub-problems is a slight variation of the AOP under the MNL model with cardinality constraint.

3.5 Approximation Algorithms, Conditions for Monotonicity and Submodularity

In this section, we provide some conditions under which the objective of the AOP and CTROP is monotone or submodular. These conditions allow us to derive some approximation guarantees for the mentioned problems.

Consider the set of feasible assortments by $\mathcal{F} \subseteq 2^{\mathcal{N}}$. We assume $\mathcal{F}$ is downward closed, i.e., if $S \in \mathcal{F}$, then $T \in \mathcal{F}$, $\forall T \subseteq S$. Note that the feasible region with cardinality constraint $\mathcal{F} = \{S \subseteq \mathcal{N} : |S| \leq k\}$, the feasible region with knapsack constraint $\mathcal{F} = \{S \subseteq \mathcal{N} : \sum_{i \in S} c_i \leq C\}$ for $c_i \geq 0$, and the unconstrained feasible region $\mathcal{F} = \{S \subseteq \mathcal{N}\}$ are all downward closed.

A set function $f : 2^{\mathcal{N}} \to \mathbb{R}$ is monotone if we have the following:

$$f(S) \leq f(T) \qquad \forall S \subseteq T, T \in \mathcal{F}. \tag{3.14}$$

A set function $f : 2^{\mathcal{N}} \to \mathbb{R}$ is submodular if it has the following property:

$$f(T \cup \{i\}) - f(T) \leq f(S \cup \{i\}) - f(S) \qquad \forall S \subseteq T \subseteq \mathcal{N} - \{i\}, T \cup \{i\} \in \mathcal{F}. \quad (3.15)$$

Maximizing a submodular function is an NP-hard problem and algorithms with theoretical guarantees have been proposed in the literature. For example, when $\mathcal{F}$ is given by a cardinality constraint and f is a non-decreasing submodular function, Nemhauser et al. (1978) shows that the subset obtained by greedily adding products has an $(1 - 1/e)$ approximation factor. For the non-negative (not necessarily monotone) submodular objective function, Buchbinder et al. (2015) develops an $1/2$ approximation algorithm.

Thus, if we can show that under some conditions the objective function for the AOP or the CTROP is non-decreasing submodular, then we can derive an $(1 - 1/e)$ approximation by the greedy algorithm. Or, if we merely show the conditions for submodularity then we have a $1/2$ approximation for the problem under these conditions.

3.5.1 Monotonicity

In this subsection, we investigate the conditions under which the the objective functions for the AOP and CTROP are non-decreasing.

Lemma 3.3. *The obejctive of CTROP under the CMNL-Syn model, with* $\alpha_{ji} \leq 0$ $\forall i, j \in \mathcal{N}$ *is monotone.*

The above lemma implies that when all the context effects between the pairs of the products is synergistic, the objective of CTROP becomes monotone. Consequently, the optimal set for CTROP is $\mathcal{N}$ in this case. Turning to AOP, we have the following result regarding the monotonicity of the AOP under S1-Syn:

Lemma 3.4. *When* $\alpha_{ji} = \alpha_j \leq 0$ *in CMNL model, if* $r(S, \mu, A) \leq r_i$, $\forall S \cup \{i\} \in \mathcal{F}$ *and* $\forall i \notin S$*, then the objective of AOP is monotone.*

As stated in Lemma 3.2, the condition $r(S, \mu, A) \leq r_i, \forall S,$ s.t. $S \cup \{i\} \in \mathcal{F}$ can be checked easily. Also, this condition holds when the revenues of products are similar as stated in lemma 3.1.

3.5.2 Submodularity

In this subsection, we derive conditions for submodularity of objective functions of the CTROP.

Submodularity Condition for CTROP under S1-Syn:

Denote $e^{\alpha_i} := A_i$. For synergistic case, as $\alpha_i \leq 0$, $A_i \leq 1$. The following theorem gives the sufficient condition for the submodularity of the objective function in CTROP, and hence, provides a $(1 - 1/e)$ approximation guarantee.

Theorem 3.4. *Denote the the attractiveness of product i under special case S1-Syn, and when set S is offered by $v_i^{(S)}$, i.e. $v_i^{(S)} = e^{\mu_i + \sum_{k \notin S} \alpha_k}$. If $\sum_{i \in S} v_i^{(S)} \geq 1$, $\forall S \in \mathcal{F}$, then the objective of CTROP is submodular, and consequently, adding the items in a greedy manner approximates the optimal solution of CTROP under a cardinality constraint with a $(1 - 1/e)$ approximation factor.*

$\sum_{i \in S} v_i^{(S)} \geq 1$, $\forall S \in \mathcal{F}$ means that the market share of offered items is always larger than the share of the no-purchase option. This is not very restrictive and is likely to hold when the firm which is offering the items is a dominating force in the market.

Submodularity Condition for CTROP under Antagonistic S1:

Under the case where $\alpha_{ji} = \alpha_j \geq 0$, $\forall i, j \in \mathcal{N}$, deriving the submodularity conditions does not lead to any interpretable condition. Instead, we will consider the submodularity of another function. Note that the CTROP under $S1$ can be written as follows:

$$\max_{S \in \mathcal{F}} \frac{(\sum_{k \in S} e^{\mu_k}) e^{\sum_{k \notin S} \alpha_k}}{1 + (\sum_{k \in S} e^{\mu_k}) e^{\sum_{k \notin S} \alpha_k}},$$

which is equivalent to the following problem:

$$\max_{S\in\mathcal{F}} M^{'}(S) = (\sum_{k\in S} e^{\mu_k}) e^{\sum_{k\notin S}\alpha_k}. \tag{3.16}$$

We now derive a condition for the submodularity of function $M^{'}()$, and then, we will state an approximation result for the objective of the CTROP.

Theorem 3.5. *Denote* $\max_{i\in\mathcal{N}} \mu_i = \mu_{\max}$, $\min_{i\in\mathcal{N}} \mu_i = \mu_{\min}$, *and* $\max_{i\in\mathcal{N}} \alpha_i = \alpha_{\max}$ *when the underlying model is S1. If*

$$\log(1 + \frac{2e^{\mu_{\min}}}{ne^{\mu_{\max}}}) \geq \alpha_{\max},$$

then the function $M^{'}()$ *is submodular, and there exists a randomized linear time* $1/3$*-approximation for the CTROP problem.*

The condition of the theorem puts a cap on the positivity of α_j parameters. Thus, if α_j parameters are small enough the submodularity guaranteed and the result of the theorem holds.

3.6 Heuristics and Numerical Study

In this section, we present several heuristics for solving the NP-hard AOP under the general CMNL model and conduct numerical studies to compare their performance. Note that for the sake of generality, we test the performance of these heuristics under the general CMNL as it can be used to model diverse contextual interaction, as apposed to special cases of this model where they focus on specific types of contextual interactions. Also, note that for some of these special cases we already have tractable algorithms to solve AOP, and hence there is no need to resort to heuristics.

First, we start with "Revenue-Ordered" heuristic which starts from the empty set and adds the product with the highest revenue which is not added yet and stop when observing a drop in the expected revenue objective of AOP. "Just Add" heuristic also

starts with the empty set, and in each step, the heuristic adds the product which gives the highest increase in the objective function of the AOP. In contrast, "Just Remove" heuristic starts with the set of the universe of products, $\mathcal{N}$, and removes the item which gives the highest increase in the expected revenue. In each step of the "Max Incremental Increase" heuristic, either one of the already added products is removed or one of the not-added products is added to the set depending on which one gives a higher increase in the expected revenue objective. While "Max Incremental Increase" heuristic allows for just one change in the set of products in each step, the "2-opt Exchange" and "3-opt Exchange" allows for 2 and 3 simultaneous changes in the set of products in each iteration. As the terminating solution of heuristics depends on the initial set that the heuristic's algorithm starts from, we repeat the experiments on heuristics with different starting set and report the results for them as well.

3.6.1 Heuristics

Revenue-Ordered, Just Add, Just Remove:

The "Revenue-Ordered" heuristic tries to find the optimal assortment by *(i)* adding the product with the highest revenue first and *(ii)* continuing by adding the product with the next highest revenue which has not been added, and *(iii)* stopping when seeing a decrease in the objective value. This heuristic finds the optimal assortment for the MNL model (Talluri and Van Ryzin 2004).

In the "Just Add" heuristic, we start with the empty set, $S_1 = \{\}$. Then, we continue by adding just one product in each iteration. Specifically, in each iteration, we add a product to the set of already added products which results in the highest increase in the objective value of the AOP. We stop when the objective value cannot be increased by adding any of the remaining items.

In the "Just Remove" heuristic, we start with the set which includes all the candidate products: $S_1 = \mathcal{N}$. Then, we keep eliminating the items one by one in each iteration, and in each iteration, we remove the item which gives us the highest increase in the objective value. We stop when eliminating none of the products increases the

objective value. In the following tables we denote the "Reveneue-Ordered", "Just Add", and "Just Remove" heuristics with "RO", "JA", and "JR" respectively.

Max Incremental Increase:

In this heuristic (which is denoted by "MII" in the following tables), we start with the empty set, $S_1 = \{\}$. Then, in iteration k, we do the following:

- Denote the obtained set in iteration $k-1$ by S_{k-1}. Generate n different sets of products which is different from S_{k-1} in only one product. In other words, for every $i \in S_{k-1}$, generate set $S_{k-1} - \{i\}$. Also, for every $i \notin S_{k-1}$, generate set $S_{k-1} \cup \{i\}$. In this way, we can get n new subset of products.

- Out of the n generated sets, pick the one which has the highest increase in revenue as S_k. Proceed to step k.

- If all the generated subsets have revenues which are smaller than the revenue of S_{k-1}, output S_{k-1} and terminate.

Max Incremental Increase with Warm Start:

This heuristic is exactly the same as the "MII" heuristic, but with one difference: S_1 is not necessarily the empty set and it can be any given set. Specifically, we tested this heuristic with the start points which were the output of the "RO", "JA", "JR" heuristics. Interestingly, in most instances having a warm start increases the quality of the final solution. We denote the "MII" with "RO", "JA", and "JR" starts respectively with "MII-RO", "MII-JA", and "MII-JR".

Improvement Heuristics: 2-Opt Exchange and 3-Opt Exchange:

In these heuristics, we try to further improve the results of the "MII-RO", "MII-JA", and "MII-JR". Also, these heuristics have the same theme of improvement as the "MII" heuristic but rather provide a larger number of alternatives in each iteration.

In each iteration of the 2-Opt Exchange heuristic, which we denote by "2-Opt" when generating the alternative assortments we change two elements of the availability vector (instead of one element in case of "MII" heuristic). This will give us $O(N^2)$ possible alternatives in each iteration. We pick the one which gives us the

highest increase in the objective value. In the 3-Opt exchange (denoted by "3-Opt"), we change 3 elements of the availability vector which yields in $O(N^3)$ of alternatives in each iteration.

In our implementation of "2-opt" heuristic, in each step, first, we performed a "MII" step, and in each step, in case of no improvement of the objective value, we performed a 2-opt exchange step. Similarly, in the implementation of "3-opt" heuristic, in each step, in case of having no improvement from "MII" and "2-opt" steps we performed a "3-opt" step. This implementation, results in much smaller computation time. Note that because of the heuristics' designs, the "2opt" heuristics will generate solutions which are at least as good as "MII" heuristics; and similarly the "3opt" heuristics will generate solutions which are at least as good as the "2opt" heuristics.

We also consider the quality of the best solutions generated by combination of different heuristics. For instance, we considered a heuristic that runs the "MII", "MII-RO", "MII-JA", "MII-JR" heuristics and reports the best solution found. We will denote this heuristic with "MII-Best". Similarly, "$i$opt-Best" runs "$i$opt-RO", "$i$opt-JA", "$i$opt-JR" and reports the best solution of them for $i = 2, 3$. We refer to these heuristics as "Combined Heuristics".

3.6.2 Numerical Study

To test the efficiency of different heuristics, we consider different problem sizes (problems with different number of products, N) of the AOP and then randomly generate 20 instances for each problem size. Then, we measure the efficiency of the proposed heuristics by comparing the objective value of the solution with the optimal objective value. In all of the randomly generated instances, the parameters and the revenues of products are randomly generated according to uniform distributions $\mu_i \sim U(-2, 0)$, $\alpha_{ji} \sim U(-0.25, 0.25)$, and $r_i \sim U(0, 1)$ respectively.

We generate the output of the different heuristics for each of the instances. The global optimal solution of the instances is found using the BARON solver version 19.12.7. (Tawarmalani and Sahinidis 2005, Sahinidis 2017) on a computer with an

Intel(R) Core i7 CPU 980-3.33 GHz processor and 12 GB RAM operating on 64-bit Windows 10. When running BARON, the best generated solution by different heuristics is inputted to the solver as the starting point. To compare the quality of the generated solutions of each heuristic, we calculate for each instance the optimality gap percentage as follows:

$$\frac{\text{Objective Value Generated by BARON} - \text{Objective of Heuristic}}{\text{Objective Value Generated by BARON}} \times 100$$

The instances with size 40 products or less could be solved to optimality in BARON in a reasonable amount of time (about 95% of the instances are solved in less than 1 hour) with the given starting point. The mean of the optimality gap percentage for different randomly generated instances are reported in Table 3.4. For these instance sizes, the median of the optimality gap percentage for all of the heuristics is 0, which means all the heuristics generate the optimal solution in at least half of the instances of each size.

Table 3.4: Mean optimality gap percentage of AOP with an underlying CMNL model for different heuristic with instant sizes 5-40.

# P	5	10	15	20	25	30	35	40
MII	0.00	0.00	0.00	0.00	0.01	0.00	0.27	0.09
MII-RO	0.00	0.00	0.00	0.00	0.01	0.08	0.37	0.27
MII-JA	0.00	0.00	0.00	0.00	0.01	0.00	0.20	0.03
MII-JR	0.00	0.00	0.06	0.00	0.22	0.23	0.46	0.55
2opt-RO	0.00	0.00	0.00	0.00	0.01	0.08	0.32	0.19
2opt-JA	0.00	0.00	0.00	0.00	0.01	0.00	0.13	0.35
2opt-JR	0.00	0.00	0.00	0.00	0.07	0.03	0.41	0.47
3opt-RO	0.00	0.00	0.00	0.00	0.00	0.08	0.12	0.17
3opt-JA	0.00	0.00	0.00	0.00	0.00	0.00	0.12	0.03
3opt-JR	0.00	0.00	0.00	0.00	0.05	0.00	0.37	0.42
MII-Best	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.02
2opt-Best	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02
3opt-Best	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

As it can be seen in Table 3.4, for 160 instances of size 40 or less, the "3opt-JA"

heuristic performs the best among all of the heuristics (excluding the "Combined Heuristics") with the mean optimality gap percentage of at most 0.12%. It generates optimal solutions for 155 instances. Among the "Combined heuristics", the "3opt-Best" finds the global optimal solution of all the instances, and "2opt-Best" fails to find the optimal solution in only one instance. The "MII-Best" finds the global optimal solution of 157 instances, with a maximum optimality gap percentage of only 0.41%.

The instances with a problem size of 50 products or more can not be solved to global optimality in a reasonable amount of time with our PC. Thus, we input the best generated solution of the heuristics to BARON as the starting point and let BARON run for 3600 seconds. Then, we compute the percentage gap of the best found solution by BARON within this time with generated solutions of heuristics. Table 3.5 reports the median\mean of this percentage gap.

Again, as it can be seen in the Table, the improvements by BARON over the generated solutions of all the heuristics are notably small. Specifically, "2opt-Best" and "3opt-Best" heuristics generate high-quality solutions that in the majority of the instances could not be improved by the BARON global solver in the given time. The median Gap Percentage was 0 for all the instance sizes when comparing the generated solutions by these heuristics with improved solutions for these heuristics. The average gap is at most 0.56% and 0.52% for "2opt-Best" and "3opt-Best" heuristics. In addition, the maximum gap among all the instances is 3.57% for both of these heuristics.

Table 3.6 shows the mean run-times of different heuristics. We do not report the run times for instances with 40 or fewer products as in all of these instances CPU-time consumed was relatively small. The following table confirms that even for large instances, the presented heuristics obtain high-quality solutions in a relatively small amount of time. From Tables 3.5 and 3.6 simultaneously, we note that the quality of generated solutions by "3opt-Best" heuristic is not significantly different than that of "2opt-Best", while "3opt" heuristics, in general, take much longer to execute for the large instances. Thus, "2opt-Best" heuristic provides the right balance between the consumed CPU-time and the quality of generated solutions.

Table 3.5: Median\Mean optimality gap percentage of AOP with an underlying CMNL model for different Heuristic with instant sizes 50-100.

# P	50	60	70	80	90	100
MII	0.00	0.62	0.20	0.25	0.20	0.41
	0.69	0.84	1.08	1.22	1.25	1.34
MII-RO	0.00	0.50	0.20	0.25	0.10	0.39
	0.66	0.82	1.06	1.12	1.04	1.01
MII-JA	0.00	0.53	0.06	0.20	0.04	0.42
	0.25	0.78	0.76	1.04	1.34	1.34
MII-JR	0.00	0.02	0.61	0.30	0.33	0.34
	0.86	1.41	2.99	2.44	1.96	1.78
2opt-RO	0.00	0.03	0.16	0.00	0.01	0.14
	0.55	0.71	0.96	1.01	1.02	0.89
2opt-JA	0.00	0.20	0.03	0.01	0.02	0.31
	0.18	0.72	0.63	0.92	1.32	1.22
2opt-JR	0.00	0.02	0.51	0.26	0.30	0.34
	0.81	1.37	2.93	2.23	1.95	1.77
3opt-RO	0.00	0.00	0.00	0.00	0.00	0.00
	0.51	0.50	0.85	0.77	0.98	0.77
3opt-JA	0.00	0.00	0.00	0.00	0.01	0.14
	0.10	0.58	0.52	0.68	1.32	1.20
3opt-JR	0.00	0.00	0.51	0.00	0.30	0.17
	0.79	1.27	2.86	2.21	1.95	1.30
MII-Best	0.00	0.00	0.00	0.00	0.00	0.17
	0.07	0.08	0.39	0.30	0.29	0.68
2opt-Best	0.00	0.00	0.00	0.00	0.00	0.00
	0.06	0.02	0.33	0.30	0.28	0.56
3opt-Best	0.00	0.00	0.00	0.00	0.00	0.00
	0.06	0.00	0.31	0.23	0.26	0.52

Table 3.6: Mean CPU-time for different heuristic for solving AOP with an underlying CMNL model with different instant sizes.

# P	50	60	70	80	90	100
MII	0.03	0.04	0.07	0.11	0.16	0.38
MII-RO	0.02	0.03	0.06	0.09	0.15	0.34
MII-JA	0.03	0.04	0.07	0.10	0.14	0.31
MII-JR	0.05	0.08	0.11	0.15	0.21	0.42
2opt-RO	0.08	0.12	0.19	0.29	0.41	0.99
2opt-JA	0.08	0.13	0.20	0.30	0.39	0.92
2opt-JR	0.10	0.16	0.25	0.32	0.46	0.98
3opt-RO	0.88	1.69	2.90	4.69	7.21	18.29
3opt-JA	0.90	1.75	2.96	4.69	7.13	18.04
3opt-JR	0.87	1.81	2.93	4.62	7.01	18.10
MII-Best	0.13	0.19	0.31	0.45	0.66	1.46
2opt-Best	0.29	0.45	0.71	1.01	1.42	3.28
3opt-Best	2.68	5.29	8.86	14.11	21.51	54.81

Chapter 4

Multi-choice Modeling with Context Effects

In this chapter, we propose an ML model that can capture contextual interaction among the products in dense choice data sets with a multi-choice outcome assumption. We empirically validate this model through an extensive analysis on 70 impression/click data sets chosen from diverse product categories. We study the relevant recommendation optimization problems including the assortment optimization and click through rate optimization for this model.

4.1 Model Description

We assume a logit-base model with Random Utility framework to capture the decision process of customers when facing multiple options. In our modeling setting, a customer perceives a random utility from each product i when facing a set of offered products by the retailer and makes a decision on choosing or not choosing each item separately, while taking into consideration what other options are offered. We denote the set of available options by set S, where this set is chosen from a superset of products $\mathcal{N}$ by retailer to be offered to the customer. More specifically, when deciding about selecting item i, the customer compares the utility of two options of "choosing item i" and "not choosing item i" (which we denote by i and i^0 respectively). The utilities of the two options depend on S. In this way one can capture the effect of presence of other options available in S on the decision of selecting item i.

4.1.1 General Model

We assume the utilities of the mentioned two options, denoted by $U_i(S)$ and $U_{i^0}(S)$, $\forall i \in S$ respectively, consist of two parts: (i) the deterministic part, denoted by $u_i(S)$ and $u_{i^0}(S)$, and a zero-mean random part ϵ_i and ϵ_{i^0}.

The deterministic part reflects the expected utility perceived from choosing the options i and i^0, which is dependent on S. More specifically, we assume the following

structure:

$$\begin{cases} u_i(S) = \gamma_i + \sum_{j \in S, j \neq i} \beta_{ji} & \forall i \in S, \\ u_{i^0}(S) = \gamma_{i^0} + \sum_{j \in S, j \neq i} \beta_{ji^0} & \forall i \in S, \end{cases} \tag{4.1}$$

where γ_i and γ_{i^0} are intrinsic utilities of choosing or not choosing i, and merely depend on item i itself; β_{ji} and β_{ji^0} capture the effect of item $j \in S$, $j \neq i$ on the utilities of i and 0^i. By having β_{ji} and β_{ji^0} terms in the utility structure, we can model and estimate the effect of offering other items beside item i and their effect on purchase (choice) decisions about i. In other words, if there exists another item j that can be considered as a direct substitute for item i and customers prefer item j to i, the presence of item j should either negatively affect the utility of item i or positively affect the utility of 0^i. Note that β_{ji} can be positive as well. For instance, if there exists an item j which can be regarded as the decoy item for target item i, β_{ji} can be positive. The random parts ϵ_i and ϵ_{i^0} are assumed to be iid with Gumbel distribution.

To model the multi-purchase behavior, we assume that customers decide about the purchase of each item $i \in S$ separately. Given the defined decision and utility structure, we can obtain the following logistic-type probability structure for the choice of item i:

$$\begin{cases} P_i(S) = \dfrac{\exp(\gamma_i + \sum_{j \in S, j \neq i} \beta_{ji})}{\exp(\gamma_{i^0} + \sum_{j \in S, j \neq i} \beta_{ji^0}) + \exp(\gamma_i + \sum_{j \in S, j \neq i} \beta_{ji})} & \forall i \in S, \\ P_{i^0}(S) = \dfrac{\exp(\gamma_{i^0} + \sum_{j \in S, j \neq i} \beta_{ji^0})}{\exp(\gamma_{i^0} + \sum_{j \in S, j \neq i} \beta_{ji^0}) + \exp(\gamma_i + \sum_{j \in S, j \neq i} \beta_{ji})} & \forall i \in S. \end{cases} \tag{4.2}$$

We can rewrite these probability structures by multiplying the numerator and denominator by $\exp(\gamma_{i^0} + \sum_{j \in S, j \neq i} \beta_{ji^0})$. In order to have unique parameter estimates and probabilities, we will use the following normalized versions and substituting $\mu_i = \gamma_i - \gamma_{i^0}$, $\alpha_{ji} = \beta_{ji} - \beta_{ji^0}$, $\forall i, j \in S, i \neq j$. When learning the parameter values, it suffices to estimate the normalized intrinsic utility of each item i and normalized

effect of other items $j \in S$:

$$\begin{cases} P_i(S) = \dfrac{\exp\left(\mu_i + \sum_{j \in S, j \neq i} \alpha_{ji}\right)}{1 + \exp\left(\mu_i + \sum_{j \in S, j \neq i} \alpha_{ji}\right)} & \forall i \in S, \\ P_{i^0}(S) = \dfrac{1}{1 + \exp\left(\mu_i + \sum_{j \in S, j \neq i} \alpha_{ji}\right)} & \forall i \in S. \end{cases} \tag{4.3}$$

Note that unlike MNL-type models in our modeling framework, we can still obtain choice probability structure (4.3) if ϵ_is, $i \in S$ are not iid. We only need every pair of distributions ϵ_i and ϵ_{i^0} to be iid w.r.t each other.

With this modeling framework, we can model multi-purchase while accounting for substitution and context effects induced by competing options. Now, we proceed to explaining how CL model can help us in detecting different types of interactions among the products that may arise in multi-purchase/click settings.

4.1.2 Special Cases for capturing Substitution and Similarity Effects

When the products are substitutable w.r.t. one another, we may expect offering them beside each other has a negative effect on the purchase probability or the CTR of items. While most of the choice models are capable of modeling the negative effect of substitute products, they fail to address the cases where the user buys/clicks on more than one item. We can use CL to model both the substitution effect of products and multi purchase/click. By restricting α_{ji} parameters, $i, j \in \mathcal{N}$, to be non-positive one can capture the negative effect of substitute products.

In addition, as already mentioned, the degree of similarity between the substitute products has a direct relation with the magnitude of this negative effect. We illustrate this point by giving an example. As can be seen in Figure 4.1-(a), assume that 3 products $\{1, 2, 3\}$ with the baseline utilities of $\mu_1 = 0.5$, $\mu_2 = 0.3$, $\mu_3 = 0.1$ are offered in a product recommendation module. Now, let us replace product 2 with a product which is very similar to product 1. In an extreme case, let us assume we offer product 1 twice (see Figure 4.1-(b)) and denote the second link of product 1, by $1'$. In this case, due to similarity effect we may expect a drop in the utility score and CTRs of

Figure 4.1: Example for illustrating the similarity effect. (a) Items recommended in product recommendation module, (b) Second item is replaced with a replica of first item in order to illustrate the similarity effect.

products 1 and $1'$ and even a change in the relative rankings among products. Thus, when optimizing on metrics like CTR, we should be aware of similarity effect.

In a given data set, if most of the items are relatively similar to each other, we may expect restricting the parameter space of the interaction terms to $\alpha_{ji} \approx \alpha_j$ and $\alpha_{ji} \approx \alpha_i$, $i, j \in \mathcal{N}$, be good approximations of the full model. Since the items are relatively similar, they may have a similar effect on other items ($\alpha_{ji} = \alpha_j, \forall i, j \in \mathcal{N}$); or being influenced almost the same from other similar items which are offered in the assortment ($\alpha_{ji} = \alpha_i, \forall i, j \in \mathcal{N}$). Note that the first and second indexes determine the interaction term α_{ji} in special cases $\alpha_{ji} = \alpha_j$ and $\alpha_{ji} = \alpha_i$ and because of this we call them S1 and S2 respectively. By the way we restrict the parameter space in S1 and S2, if the similarity effect is the dominant type of interaction among the items, we expect these models to have higher prediction capabilities (almost as good as the full model if the data set is large enough).

4.1.3 Special Case for Capturing Complementarity and Synergistic Effects

The recommended items can be complementary w.r.t. one another as well. Complementary items are not generally from the same category of products and there exist

modules in e-commerce that offer complementary items beside each other. For instance, each item page of walmart.com may include a module titled "Customers also bought these products". This module recommends products that are co-purchased with the anchor item of the item page, and themselves can be complementary for each other. See Figure 4.2 for an example. In this figure items which are purchased beside the anchor item (office chair) are recommended. Among those office desks and desk lamps are complementary.

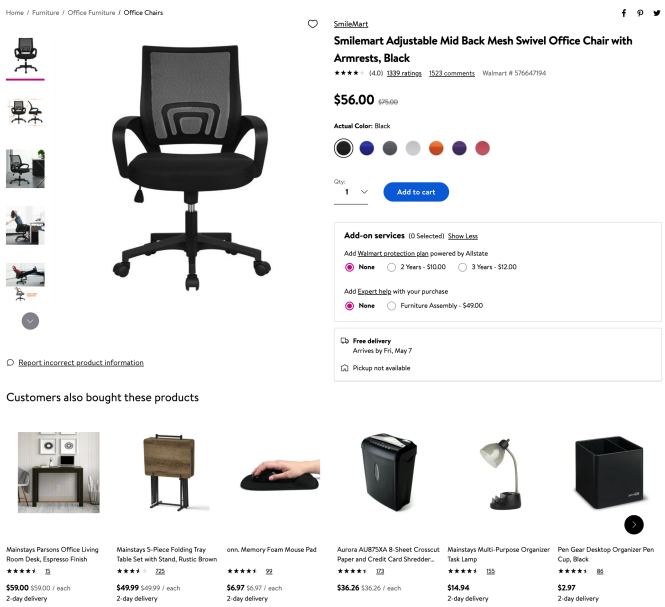

Figure 4.2: Some of the items offered in the Customers also bought these products can be considered complementary for each other.

Offering complementary items beside each other may have a positive effect on their utility as customers may end up considering more ways to consume those products. Using CL model, if items i and j are complementary we may expect $\alpha_{ji} \geq 0$. Positive effects on the utility of a given item can be caused by reasons other than complementarity (see Introduction). We denote the CL model with restricted parameter space $\alpha_{ji} \geq 0$, $\forall i, j \in \mathcal{N}$, the "CL-syn" model. There may be other examples that CL-syn can be used for modeling. For instance, there are several recommendation modules

in the grocery homepage of walmart.com[1] (See Figure 4.3 for an screen shot of the homepage).

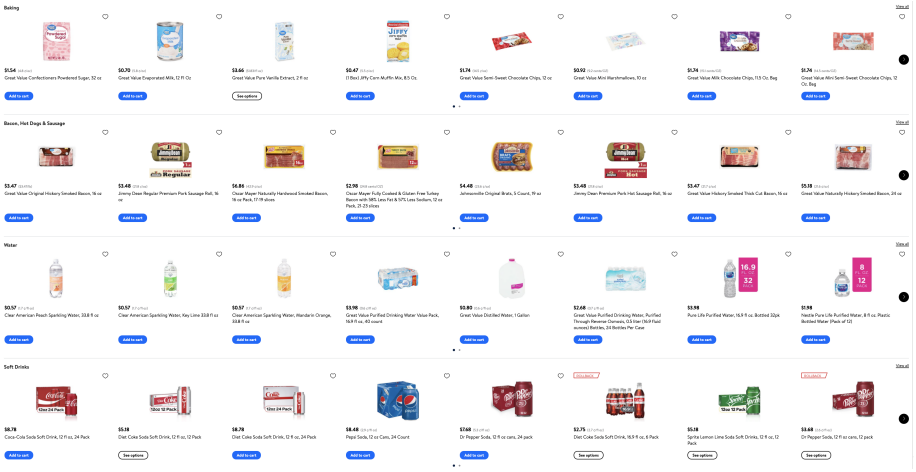

Figure 4.3: The recommendation modules (each row) that can be recommended in https://www.walmart.com/grocery/. The interactions are either positive (due to some degree of complementarity) or zero (as some module pairs are independent).

The content of each module is determined independently using complex ML models. Each of these modules specifically focuses on a product category and because of this, if we consider the module-level CTR of each of them, it is either independent of other modules, or it is complementary w.r.t. others. Thus, in this setting it is appropriate to assume $\alpha_{ji} \geq 0$, $\forall i, j \in \mathcal{N}$, when modeling CTR interactions among recommendation modules.

For any other special case of CL, when we add "-syn" we restrict that special case further to positive interaction terms.

4.1.4 Special Case for Capturing Decoy and Compromise Effects, and the case of new Trend Setting Items

A decoy item is designed to boost the attractiveness of a target item as explained in the introduction. If item j is a decoy for target item i, then we may expect $\alpha_{ji} > 0$

[1]https://www.walmart.com/grocery/

and $\alpha_{jk} < 0$ for $k \neq i, j$ under the CL model. In this way, we may be able to model the positive effect of item i on the purchase rate or CTR of item j. If the decoy effect is the dominant type of interaction in a given data set, we may model the effects from decoy item j on the utility of other items by merely setting the j^{th} row of the interaction matrix $A = [\alpha_{ji}]_{i,j\in\mathcal{N}}$ to be non-zero and the rest of the rows to be zero. We call this special case of the CL model d_j.

This special case can also be used for modeling purpose in a market in which a dominant and popular product is newly introduced. This newly introduced product can change the purchase/click rate of other products significantly. One example is the introduction of a new gaming console "Play Station 5" in November 12, 2020. After introduction of this product, market shares of the gaming consoles and the relevant accessories changed (Taylor 2021).

Similarly, the effect of adding a luxury item j to an existing option i (as a compromise between extremely expensive or cheap items) can be captured using the CL model and setting $\alpha_{ji} > 0$.

4.2 Empirical Validation

In this section, we present the tests we have performed on real data sets from diverse categories of products in order to prove the efficiency of our considered model by comparing it to some of the proposed models for multi-purchase/click modeling in the literature.

We compare the CL model and its special cases with three benchmark models: the point-wise logistic regression, the ListMLE model, and Multivariate Logit Model. Table 4.1 summarizes needed information for each model for a better reference.

We performed two sets of tests for evaluating goodness of fit as well as the prediction power of the models.

First, we compute the log-likelihood, AIC, and BIC scores of the models on the whole data sets. When computing AIC and BIC scores, the log-likelihood of the models are penalized for having larger number of parameters to make a fairer comparison

Table 4.1: Special Cases of the CL Model

The Model	Parameter Restriction	Short Description or Interpretation
PW-LogReg	$\alpha_{ji} = 0$	Simple logistic regression for the choice probability of each item, logistic regression independent of assortment set
ListMLE	NA	Multi-purchase MNL model, (see Xia et al. 2008, and Feldman et al. 2020)
MVL	NA	Multivariate Logit Model
CL	None	Full Contextual Logit Model
CL-Syn	$\alpha_{ji} \geq 0$	CL model with complementary/synergistic interactions
S1	$\alpha_{ji} = \alpha_j$	First index, j, determines the interaction term, May work well when the similarity effect is the dominant type of contextual interaction
S1-Syn	$\alpha_{ji} = \alpha_j \geq 0$	S1 with Complementary/Synergistic interactions
S2	$\alpha_{ji} = \alpha_i$	Second index, i, determines the interaction term, Utility being a linear function of cardinality of assortment, May work well when the similarity effect is dominant
d_i	$\alpha_{jl} = 0, \forall j \neq i$	item i is the decoy/trend setting item,

between simpler and more complex models.

We also test the predictive power of the models by (i) dividing the data sets into training and testing subsets, (ii) finding the maximum likelihood estimators of the models on the train sets, and (iii) calculating Cross Entropy (CE), Mean Reciprocal Rank (MRR), and Normalized Discounted Cumulative Gain (NDCG) scores of the models in the test data sets.

4.2.1 The Data

In our tests, we use the click and impression data of a substitute product recommendation module in the item pages of an online retailer. This module can be seen after clicking on an item and scrolling down the page. It tries to recommend substitute alternative products for the anchor item of the page. See figure 4.4 for an example of this module. For each browsing session, the data includes the items shown to the customer and the ones which are clicked with the time stamp of each click.

We considered a comprehensive data for 70 item pages with the highest average number of clicks per session in this module from different product categories. We performed the tests for the data related to each of these 70 pages separately as they include different items from different categories of products including but not limited to furniture (home and office), cell phones & mobile accessories, toys, computers & laptops, TV & video, patio & garden products, household essentials, decor, sexual wellness products, and kitchen and dining among others. We specifically tried to test the performance of the models on data related to products from diverse categories for robustness of results. The average number of clicks per session varies between about 0.25 to 0.55 in our data sets with 10 to 50 percent of the sessions with clicks having more than one click. In addition, the data sets include 14,902-116,111 sessions and 8-37 products.

4.2.2 Performed Tests

First, the MLEs of the models are computed when fitting the models on the whole data sets. We use Barzilai-Borwein (BB) method (see Barzilai and Borwein 1988, Fletcher 2005) for calculating the MLEs of our benchmark models as well as the CL model, and all the special cases other than the synergistic model.[2] We terminate the BB gradient algorithm when the L_2-norm of the gradient vector is less than 10^{-4} in all the models. For the MLEs of the synergistic special cases of the CL model (as

[2]We used BB since it converges faster than the gradient descent and stochastic gradient descent algorithms in most of our data sets. A comprehensive study can be performed on the performance of different optimization algorithms on log-likelihood maximization of the multi-choice models.

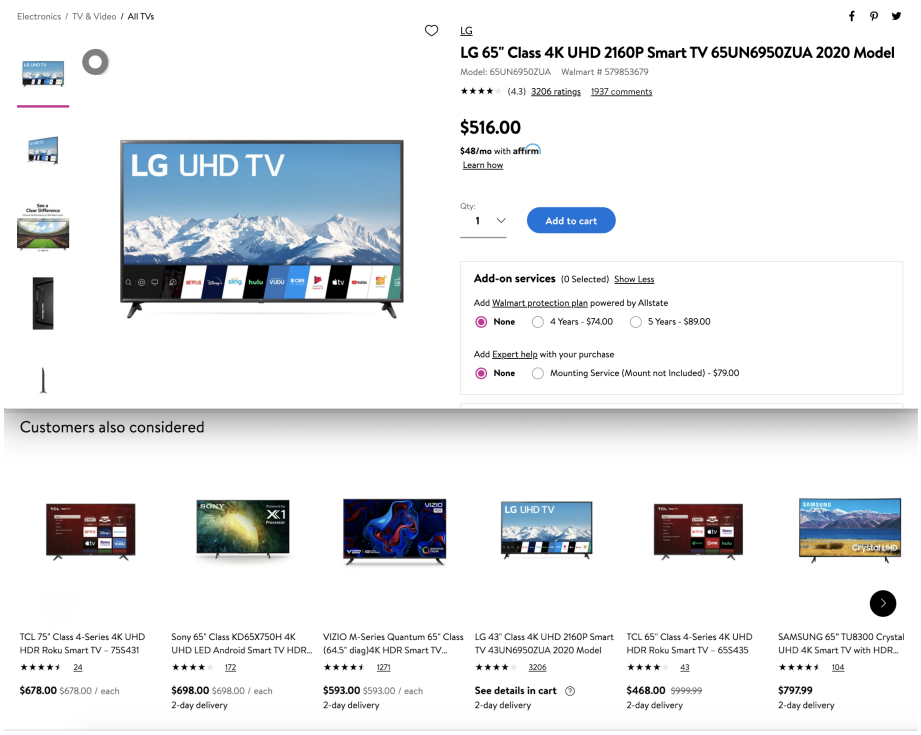

Figure 4.4: An example for the recommended items in the Customer- also-considered module of the item page. The recommended products recommend alternative substitutes for the anchor item of the page.

their parameter space is restricted to positive interaction terms), we use Branch-And-Reduce Optimization Navigator (BARON) solver version 20.10.16 (Sahinidis 2017; Tawarmalani and Sahinidis 2005) under Pyomo package (Hart et al. 2011, 2017) in Python with default terminating conditions on a PC with an 2.4 GHz 8-Core Intel Core i9 processor and 64 GB 2667 MHz DDR4 memory operating on macOS. BARON is a global optimization solver.

The log-likelihood, AIC, and BIC Score: After calculating the MLEs of each model, we calculate the log-likelihood scores of the models. This score can be a indication for how likely the data is under the model of consideration and can reflect how well the model can describe the data.

However, the log-likelihood scores cannot merely reflect the descriptive power of

the models when the number of the parameters are not the same in models that are compared. AIC and BIC scores penalize the log-likelihood score for having larger number of parameters as follows:

$$\begin{cases} \text{AIC} = -2\hat{L} + 2d \\ \text{BIC} = -2\hat{L} + d\ln(T) \end{cases}$$

where $\hat{L}$ is the log-likelihood of the given model, d is the number of parameters, and T is the number of browsing sessions in the data (Burnham and Anderson 2002). The middle column of Table 4.2 reports the average percent improvement for different models over our 70 click/impression data sets w.r.t. the benchmark PW-LogReg model. Note that these scores are log-based scores and even a relatively small change in the scores means a significance difference in likelihood of the models.

To compare the predictive power of the models, we divide the data sets into random train-test splits with equal size. Then, we obtain the MLEs of the models in train splits; and with the obtained parameters, we calculate the following scores (see Chen et al. 2009, Radlinski et al. 2008, Zhang and Sabuncu 2018 for more on these scores):

1. **CE Score**: This is actually the log-likelihood of the test split, when the parameters are the MLEs of model in training split. This score reflects how likely the model is in a given data set and a higher log-likelihood score means that the model can predict the click behavior in the test set better. If a set B_t of items are clicked when set S_t is shown to the customer in session t, the following is the log-likelihood of the session in logit-based models (including PW-LogReg, MVL, CL and its special cases):

$$\hat{L}_t = \ln\big(\prod_{i\in B_t} P_i \times \prod_{i\in S_t\setminus B_t} (1-P_i)\big)$$

   where P_i is the probability of clicking item i. Note that P_i is determined according to a logistic regression in logit-type models, and each model uses a different structure for calculating this probability. To calculate the log-likelihood of the

listMLE model, we consider one of the rankings which are consistent with B_t and S_t with clicked items being ranked higher than non-clicked ones and the clicked items being ranked according to the time stamp of the clicks. Then, we sum the probability of these rankings to obtain the likelihood of the model for session t. CE score is calculated by summing the log-likelihood of all the sessions in the test set.

2. **MRR score**: This score is widely used in Information Retrieval literature. To calculate this score, we consider all the items S_t, then, without considering the click set B_t, we sort the items based on their utility scores in that session. Note that as different models assign different utility scores to each item, they may lead to a different sorted order of the items. Then, we consider the reciprocal of the rank of the first item which is clicked in the sorted list. Mean Reciprocal Rank score is the average of these reciprocal ranks over all sessions. A larger MRR indicates a better suggested ranking (as well as utility score) by the model, since if in a ranking suggested by a model all the items which are clicked have a better rank, this will lead to a larger reciprocal rank and, consequently, a larger MRR.

3. **NDCG score**: Similar to MRR score, NDCG is also widely used in Information Retrieval literature. To calculate this score, first we should sort the items based on their utilities. Let's denote the rank of item i in session (if it is impressed) t by r_i^t and the indicator function for item i being clicked by $\mathbb{1}\{i \in B_t\}$. The Discounted Cumulative Gain in session t, DCG_t, reflects whether a model predicts better rank and higher utility for clicked items or not, and is calculated as follows:

$$\text{DCG}_t = \sum_{i \in S_t} \frac{2^{\mathbb{1}\{i \in B_t\}} - 1}{\log_2(r_i^t + 1)}$$

NDCG score normalizes the above score by dividing it to ideal DCG for every t given B_t and S_t. Ideal DCG is equal to the DCG score of a model that ranks all the items in B_t before other items. The final NDCG score is the average of scores in sessions of each testing splits.

The average percent improvements in CE, MRR, and NDCG scores of the models w.r.t. the PW-LogReg in all of the 70 data sets are presented in the right columns of table 4.2. We report the MRR and NDCG scores of the models inside parenthesis in addition to their percent improvement w.r.t. PW-LogReg. Note that MRR and NDCG are normalized scores and do not scale up or down with the number of browsing sessions in data sets.

When calculating the MRR and NDCG score, the utility rankings of items should be calculated without considering the clicked set. This is why these scores cannot be computed for the MVL model, as we should be aware of the clicks in the testing sets before calculating the utilities of items (and the suggested ranking) for MVL.

Table 4.2: Average percent improvement in log-likelihood ($\hat{L}$), AIC, BIC, Cross-Entropy(CE), MRR, and NDCG scores of different models w.r.t the Point-wise Logistic Regression (PW-logReg) Model. As MRR and NDCG are normalized scores we also report the average value of the scores for the models inside parenthesis.

The Model	$\hat{L}$	AIC	BIC	CE	MRR	NDCG
PW-LogReg					0.00 (0.6401)	0.00 (0.7252)
ListMLE	-0.00	-0.00	-0.00	-0.02	0.55 (0.6434)	0.38 (0.7278)
MVL	1.47	0.32	-4.62	0.53	NA	NA
CL	**4.66**	**3.52**	-1.44	**3.47**	**6.92** (0.6820)	**4.63** (0.7576)
CL-syn	0.38	-0.77	-5.71	0.05	0.69 (0.6443)	0.45 (0.7284)
S1	**2.02**	**1.96**	**1.70**	**2.13**	**2.4** (0.6544)	**1.64** (0.7365)
S1-syn	0.04	-0.02	-0.27	0.06	0.01 (0.6401)	-0.00 (0.7251)
S2	**1.67**	**1.61**	**1.35**	**1.83**	**2.57** (0.6554)	**1.78** (0.7374)
$d_{\text{high-impr}}$	0.10	0.04	-0.21	0.04	0.15 (0.6410)	0.10 (0.7259)
$d_{\text{high-clk}}$	0.17	0.11	-0.14	0.09	0.25 (0.6416)	0.17 (0.7264)

4.2.3 Empirical Validation Results

According to our results, CL model obtains the best improvement in 5 (out of 6) scores. Special cases S1 and S2 perform consistently better than all of the benchmark models in all of the scores. These special cases obtain the best BIC scores as

they seem to present a better balance between increased number of parameters and improved likelihood when we optimize on BIC with severe complexity penalization. However, in all of the three prediction scores CL model outperforms the the other models, which can be interpreted as the high prediction performance of this model in forecasting the click behavior of the users.

Note that the results presented in Table 4.2 are related to the data fit of the models. One other thing that we may care about is the computational complexity of AOP and CTROP under any given model. While the AOP is shown to be NP-hard under MVL (Tulabandhula et al. 2020), listMLE with number of purchases be given exogenously (Feldman et al. 2020), and CL model (Section 3), it is polynomially solvable under PW-LogReg, all the synergistic special cases of the CL model, and special cases S2 and d_i ($i \in \mathcal{N}$). This points out the advantage of the special case S2. While this case improves all the descriptive and predictive scores when compared to benchmark model, the AOP and CTROP under S2 is polynomially solvable. Thus, S2 has a double advantage of better data fit and tractable AOP.

In addition to the good performance of S2, S1 also improves all of the scores. As discussed, we expect these two models to work well when the similarity effect is the dominant type of contextual effect in driving the selection behavior. We speculate, in our click data sets, due to higher effect of similarity effect (compared to other types of contextual interactions) $\alpha_{ji} = \alpha_j$ and $\alpha_{ji} = \alpha_i$ are good enough approximations of the full CL model. In fact, current implemented recommendation algorithm for "People also considered" is a complex ML algorithm which computes a score for each product. This computed score merely depends on the feature values of items itself and often, ends up recommending some items with relatively similar feature values. This may be the real reason for the good performance of the S1 and S2 models. Furthermore, this accentuates the necessity of diversifying the recommended assortments to the customers.

Note that not all of the recommended products are necessarily similar in our data. In addition, S1 and S2 models are not as good as the full model in predicting the click behavior due to presence of other types of interactions between the items, since our data includes items from very diverse product categories and markets. The items'

diversity in our data set may induce a very diverse types of contextual interactions and thus, lead to high performance of the full CL model.

One other relevant note is the weak performance of the decoy models. In fact, having a decoy effect among pairs of products require items which are asymmetrically dominated w.r.t other products of the offer set. This requires a very specific feature relation among the products, which does not happen often. We think this is why the decoy models do not improve the prediction scores significantly.

Another speculation that we may conclude from the results is the importance of the platform which customer uses. Different platforms (mobile app, desktop, mobile/tablet web) may have different screen sizes and the number of products which are impressed to the customer (cardinality of the assortment) varies among the platforms. As we will show in the next section, the special case S2 is equivalent to the case where the context is driven by the cardinality of the assortment presented to the customer. The good performance of S2 w.r.t. the benchmark models can be an indication for the importance of the cardinality of the assortment (which is determined by the platform that the customer uses) in predicting the CTR of the items.

According to Table 4.2, the synergistic models have weak performances. When fitting these models, most of the interaction terms become zero, and thus, these models approximately become the same as PW-LogReg. We may expect this to happen as the items recommended in the "Customers also considered" section are substitutes of each other; and offering them beside each other may only reduce the CTR of the recommended list. In other words, in most of the cases, we may not expect an increase in the utility scores of products while offering beside each other when the products are substitute of each other and not complementary. One of the cases where we may observe synergy between the substitute product is when synergy exists due to halo effect, which requires a very specific product feature structure (see section 1.2). As our 70 data sets come from diverse product categories and backgrounds, we conjecture this weak performance of synergistic models can be extrapolated to other data sets including substitute products as well.

Finally, we would like to compare the full CL and MVL model. These two models have the same number of parameters (both n^2 parameters). Note that MVL should

perform well when the act of selecting an item affects the utility scores of the other items, while CL model should perform well when merely showing an item beside the item of interest changes the utility (not necessarily selecting/purchasing/clicking). We speculate that generally on impression/click data sets (data sets including the items which are shown and clicked in each browsing session) MVL model to perform weaker than CL model. However, in the item impression/transaction data sets (data sets including the items which are shown to the customer and the items which are purchased) MVL may produce closer results to the CL model (or maybe better). Clicking on one of the recommended items (see Figure 4.4) may not necessarily change the perceived utility of other items and affect their CTR, while buying an item (spending money) may have a stronger effect due to reasons like customers' budget constraints. In other words, the act of "clicking" may not induce a burden as large as purchase on the selection behavior. Thus, we may expect MVL perform weaker in impression/click data sets when compared to its performance on impression/purchase data sets.

All in all, our extensive data analysis on impression/click data illustrates that context and substitution effects may be an important factor in shaping the click behavior, and CL model and some of its special cases may provide a superior performance on predicting selection outcomes.

4.3 Assortment Optimization and Click Through Rate Optimization Problems

In the AOP, the retailer is interested in finding a subset of products, $S \subseteq \mathcal{N} = \{1, \ldots, n\}$, which yields the highest expected revenue. S can be represented by a binary vector $x = (x_1, \ldots, x_n)$, where $x_i = 1$ if $i \in S$, and $x_i = 0$ o.w. If we denote the revenue of product i (or equivalently the expected reward from clicking on link i) by r_i, we can define the AOP as follows:

$$R^*(\mu, A, r) = \max_{x \in \{0,1\}^n} R(\mu, A, r, x), \tag{4.4}$$

where $R(\mu, A, r, x) = \sum_{i \in S} x_i r_i P_i(x)$ is the expected revenue from offering subset x with a revenue vector $r = (r_1, \ldots, r_n)$, when the underlying selection probability is according to a CL model with baseline utility vector $\mu = (\mu_1, \ldots, \mu_n)$ and contextual matrix $A = [\alpha_{ji}]_{i,j \in \mathcal{N}}$. WLOG, assume r_i in non-increasing in index $i \in \mathcal{N}$, and when $r_i = r_{i+1}$, we have $\mu_i \geq \mu_{i+1}$, $\forall i \in \{1, \ldots, n-1\}$.

In some special problem settings, one may associate equal rewards to each of the different products or links that the retailer/advertiser offers. For instance, when the objective is to optimize on the CTR of the offered products in an e-commerce module, each click is worth equally. Click Through Rate Optimization Problem (CTROP) is the problem of finding a subset of products that maximizes the CTR, which is an instance of AOP with equal reward for each product.

$$M^*(\mu, A) = \max_{x \in \{0,1\}^n} \sum_{i=1}^{n} x_i P_i(x), \tag{4.5}$$

The following theorem shows that both CTROP and AOP are NP-hard when the underlying choice model follows the general CL model. The proofs of all the theoretical results can be found in Appendix B.

Theorem 4.1. *Problems* (4.4) *and* (4.5) *are NP-hard.*

The revenue ordered assortments include items with k highest revenues, i.e. $\{1, \ldots, k\}$, $k = 1, \ldots, n$. These assortments have proven to either provide the optimal solution or at least provide some approximation guarantees under DCM framework (see Talluri and Van Ryzin 2004, Davis et al. 2014 for instance). However, as stated in the following lemma, these assortments can perform arbitrarily bad.

Lemma 4.1. *Under the CL model, we can construct instances where the ratio between the revenues generated by the best revenue-ordered assortment and the optimal assortment is arbitrarily small.*

In contrast to Theorem 4.1, in the presence of specific types of contextual interactions, AOP may become tractable. Note that as CTROP is an instance of the AOP, tractability of AOP implies the tractability of CTROP. For brevity we only state the results for AOP in these cases.

As stated in Section 3, there are instances in which products (or links or recommendation modules) have merely positive contextual interactions with each other, or they have different degrees of complementarity (see Figure 4.3). The following proposition shows that $\mathcal{N}$ is an optimal assortment of the AOP under this special case.

Proposition 4.1. *When $\alpha_{ji} \geq 0$, $\forall i, j \in \mathcal{N}$, $\mathcal{N}$ is an optimal assortment for problem* (4.4).

This result is proven by showing that the objective of (4.4) is monotone; i.e. $R(\mu, A, r, x_1) \geq R(\mu, A, r, x_2)$ for $x_1 \geq x_2$. Note that when the products have synergy, they increase each others CTR, hence making the objective of AOP monotone. Special case S2 ($\alpha_{ji} = \alpha_i$, $\forall i, j \in \mathcal{N}$) can be associated with the case where the similarity effect is the dominant type of contextual interaction among the products as explained is Section 3.

In this case, the utility of product $i \in S$ is a functions of cardinality of the assortments offered, as shown below:

$$u_i = \mu_i + \sum_{j \in S, j \neq i} \alpha_i = \mu_i + (|S|-1)\alpha_i \qquad \forall i \in S,$$

i.e. the size of the assortment creates the context. The following result presents the tractability of AOP under special case S2:

Proposition 4.2. *The AOP is polynomially solvable under special case S2 of the CL model.*

In the proof of Proposition 4.2, the solution space can be divided to n subspaces and the optimal solution in each subspace can be found in polynomial times.

Note that according to the empirical results presented in Section 4.2, S2 performs better than all of the benchmark models in prediction scores, while having a tractable AOP. This further accentuates the applicability of this special class in some real world instances.

Special case d_i corresponds to the situation where item i is the decoy item or it is a trend-setting item, and its effect is the dominant type of contextual interaction in the data. The following result confirms the tractability of AOP under this case.

Proposition 4.3. *AOP is polynomially solvable under special case d_i; or when there are $O(1)$ many trend-setting items with dominant contextual interactions on the other items.*

To prove tractability of AOP, we can divide the solution space to $O(1)$ subspaces, for each of which one can find the optimal solution in polynomial times.

4.3.1 Heuristics

In this section, we present several heuristics for solving the NP-hard AOP under the general CL model and study their performance. Note that we propose and test these heuristic for the general CL model, since it can be used to capture diverse types of context effects. Also, note that we already have proven the tractability of obtaining of the optimal solution of AOP for some special cases of the CL model and we do not need to propose heuristics for those. More specifically, we study the performance of the solutions suggested by Revenue Ordered (RO) heuristic which reports the best solution among the revenue ordered assortments. Two other heuristics that we study are Greedy Add (GA) and Greedy Remove (GR) heuristics. GA starts from the empty set and in each step greedily adds the assortment which gives the highest increase in the AOP's objective. In contrast to GA, GR starts from the full assortment, $\mathcal{N}$, and greedily removes the items in each step. We also study the local search algorithms ($m$opt heuristics with $m = 1, 2, 3$). In each step, these heuristics search for the best neighbor point to the solution at hand and proceed to the one which attains the highest increase in the objective of the AOP. Since the trajectory of search depend on the starting point, we test the performance of the local search heuristics with different starting points as well.

Heuristics with $O(n)$ number of steps: RO heuristic finds the best revenue-ordered assortment. Note that according to Lemma 4.1, this heuristic can perform

arbitrarily bad. And according to our results, it produces the worst solutions when compared to other heuristics. Note that RO leads to optimal solution under the MNL model (Talluri and Van Ryzin 2004).

GA heuristic starts from the empty set and adds the products which give the highest increase in the objective of the AOP one by one. GR is executed in the same theme as GA but starts from $\mathcal{N}$ and removes the products greedily. Both GA\GR are terminated in a given step if adding\removing available products do not lead to an increase in the objectives. By construction, RO, GA, and GR check at most n points.

Local search heuristics: We denote these heuristics by $m$opt, $m = 1, 2, 3$. At each step, these heuristics look for the best neighbor solution. When performing an $m$opt local search step on current solution $x = (x_1, \ldots, x_n)$ the best solution is selected from set of neighbours $x^+ = \{y = (y_1, \ldots, y_n) : \sum_{i=1}^{n} |x_i - y_i| \leq m\}$. This leads to checking $O(n^m)$ neighbours at each step.

As the starting point affects the series of solutions generated, we test $m$opt heuristics with different starting points as well. We specifically consider five starting points: (i) The empty set, (ii) the full set, $\mathcal{N}$, (iii) the output of RO heuristic, (iv) The output of GA heuristic, and (v) the output of the GR heuristic. We denote the $m$opt heuristics with these starting points by (i) $m$opt, (ii) $m$opt-$\mathcal{N}$, (iii) $m$opt-RO, (iv) $m$opt-GA, and (v) $m$opt-GR respectively.

In our implementations of $m$opt local search heuristics, we first perform a 1opt step and only perform a higher order local search if the lower order search does not improve the objective value of the AOP in that step. We terminate the $m$opt local search if in a given step, even a $m$opt step does not improve the objective value. This implementation makes the search faster.

Since the starting point affects the quality of the final solution, we also consider the version of $m$opt heuristic which starts from multiple points and reports the best terminating point. We denote this heuristic by $m$opt-best.

4.3.2 Numerical Study

In our numerical study, AOP instances with different number of products were randomly generated. More precisely, 20 instances for each problem size (n = 30, 40, 50, 60, 70, 80, 90 ,100) were generated by selecting the problem's parameter values according to a uniform distribution with $\mu_i \sim U(-2, 0)$, $\alpha_{ji} \sim U(-0.25, +0.25)$, and $r_i \sim U(0, 1)$.

We restrict the baseline utility parameter values (μ_is) to non-positive values to be consistent with our empirical observations. Note that normally in practice, clicking on a given link (or selecting a purchase option) is less than the chance of not clicking (not purchasing). Setting $\mu_i \leq 0$, $\forall i \in \mathcal{N}$, we can get $P_i(\{i\}) \leq P_{i^0}(\{i\})$.

To compare the quality of generated solutions by heuristics, we calculate the optimality gap as follows:

$$\frac{\text{Optimal Objective Value} - \text{Objective Value of heuristic solution}}{\text{Optimal Objective Value}} \times 100$$

We use BARON solver version 20.10.16 (Sahinidis 2017; Tawarmalani and Sahinidis 2005) under Pyomo package (Hart et al. 2011, 2017) in Python on a PC with an 2.4 GHz 8-Core Intel Core i9 processor and 64 GB 2667 MHz DDR4 memory operating on macOS to find the optimal solution for each instance. For faster termination of BARON, we first execute all heuristics, and use the best solution with highest AOP objective as the starting point for BARON.

The percentage optimality gap results of different heuristics are reported in Table 4.3. To have a clearer presentation, we do not report the results for all the $m$opt heuristics staring from different points.

As it can be seen form the table, RO performs the worst among the heuristics. This is consistent with Lemma 4.1, which states the best RO solution can have arbitrarily large optimality gap.

Also, according to the table, the median percentage optimality gap for all the local search algorithms is zero regardless of the problem size. This means that even by using 1opt heursitic, we can solve the problem to optimality in at least half of the

instances.

2opt-Best and 3opt-Best heuristics solve all the 160 instances with less than or equal to 80 to optimality, and 1opt-Best fails to attain the optimal solution in just 1 of these 160 instances, which shows their efficiency for these instance sizes.

Mean CPU time of heuristics are reported in Table 4.4. An interesting observation is the lesser running time of local search heuristics when starting from $\mathcal{N}$, or the output of GR heuristic when compared to starting point of empty set. The optimal solutions for the randomly generated instances considered in our numerical study include assortments with high cardinality of products, and maybe because of this when we set the starting point nearer to the optimal solution (starting points of $\mathcal{N}$ and GR), the local search algorithm reaches to the optimal solution faster.

By contrasting mean running time and the optimality gap of different heuristics, 1opt-GR performs relatively well. Its mean running time is 0.61 seconds and its mean optimality gap is 0.05 percent of the optimal solutions for the instances with size $n = 100$.

The other point we can notice by paying attention to table 4.4 is the higher running time of 2opt and 3opt algorithms when compared to 1opt local searches. For instance, 2opt-GR and 3opt-GR takes 3.2 and 74.13 times more than 1opt-GR to be terminated respectively. The reason for this is the more computationally expensive steps of the higher order local heuristics. Thus, we may conclude 1opt algorithms are more efficient for our considered instances as they provide near optimal solutions relatively faster.

# P	30	40	50	60	70	80	90	100
RO	3.52	4.38	5.48	7.18	8.04	8.63	9.26	9.39
	3.49	4.13	5.58	8.21	7.26	8.96	8.97	8.99
GA	0.02	0.08	0.08	0.30	0.34	0.21	0.39	0.31
	0.00	0.00	0.00	0.00	0.22	0.03	0.16	0.17
GR	0.01	0.00	0.02	0.03	0.00	0.05	0.09	0.07
	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02
MII	0.00	0.05	0.05	0.12	0.07	0.05	0.10	0.06
	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00
MII-GR	0.01	0.00	0.00	0.03	0.00	0.03	0.07	0.05
	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2opt	0.00	0.03	0.03	0.04	0.03	0.04	0.08	0.02
	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2opt-GR	0.00	0.00	0.00	0.03	0.00	0.00	0.03	0.03
	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3opt	0.00	0.00	0.00	0.04	0.00	0.03	0.03	0.01
	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3opt-GR	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.02
	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
MII-Best	0.00	0.00	0.00	0.03	0.00	0.00	0.03	0.02
	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2opt-Best	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.01
	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3opt-Best	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.01
	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Table 4.3: Mean\Median optimality gap percentage for different heuristic solving the AOP with underlying CL model with instant sizes 30-100.

# P	30	40	50	60	70	80	90	100
RO	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
GA	0.01	0.01	0.02	0.03	0.05	0.07	0.09	1.17
GR	0.00	0.01	0.01	0.02	0.02	0.03	0.04	0.58
MII	0.01	0.02	0.04	0.05	0.08	0.11	0.15	1.95
MII-GR	0.00	0.01	0.01	0.02	0.02	0.03	0.04	0.61
2opt	0.02	0.03	0.06	0.09	0.13	0.17	0.24	3.26
2opt-GR	0.01	0.02	0.03	0.05	0.07	0.10	0.14	1.95
3opt	0.07	0.19	0.39	0.67	1.25	2.03	3.24	45.67
3opt-GR	0.06	0.16	0.33	0.66	1.03	1.71	2.65	45.22
MII-Best	0.03	0.05	0.09	0.13	0.20	0.26	0.36	4.79
2opt-Best	0.06	0.11	0.19	0.30	0.44	0.61	0.84	11.44
3opt-Best	0.34	0.83	1.78	3.27	5.57	9.16	14.24	234.60

Table 4.4: Mean CPU time spent under each heuristics when solving the AOP with an underlying CL model for each instance sizes.

Chapter 5

Top-K Retrieval Problem with Featurized Contextual Model

In this chapter we propose a model that can capture the contextual interactions among the documents in IR settings. We elaborate on the modeling framework and estimation procedure, and study the Top-K retrieval problem under this model.

5.1 Model and Estimation

First, we introduce some notation. Consider that there are T queries in a given training data and assume that subset $S^{(t)} = \{d_1^{(t)}, \ldots, d_{K^{(t)}}^{(t)}\}$ of objects with cardinality $K^{(t)}$ are presented in query t. Denote the i^{th} object in this query by $d_i^{(t)}$ and its feature vector by $x_i^{(t)}$. We can assume that all the queries have fixed cardinality, $K^{(t)} = K, \forall t = 1, \ldots, T$, as in the case of Top-$K$ retrieval problem but for the sake of generality in choice modeling we assume we can have varying cardinality.

When $S^{(t)}$ is presented, the user can either select from $S^{(t)}$ or does not choose any object. Assuming the decision of "choosing nothing" as one of the options recommended, and denoting this option with index 0, the customer chooses from $S^{(t)} \cup \{0\}$ when is recommended set $S^{(t)}$.

Let $P_i(S^{(t)})$ be the probability of choosing object i from an offered subset $S^{((t)}$.

The Contextual MNL model is a Random Utility (RU) based model. In a RU-based framework, the users assign different random utility values to different objects and end up choosing the object with the highest utilities. In most of the models RU-based, the perceived utility of $d_i^{(t)}$, $U_i^{(t)}$, is composed of two parts: (i) a deterministic part, $f_i^{(t)}$, that approximates the expected utility for a user (or user group) given its features and (ii) a random part with mean zero, $\epsilon_i^{(t)}$, to accommodate the variation in choice selection:

$$U_i^{(t)} = f_i^{(t)} + \epsilon_i^{(t)}. \tag{5.1}$$

For instance, in the MNL model, $f_i^{(t)}$ is dependent on the features of $d_i^{(t)}$ and $\epsilon_i^{(t)}$ has i.i.d. Gumbel distribution. The MNL does not consider the contextual effects that products have on each other.

To approximate the context effects, the contextual MNL model, extends MNL, by assuming f_i is dependent on subset $S^{(t)}$: $f_i^{(t)}(S^{(t)}) = \mu_i^{(t)} + \sum_{j \in S: j \neq i} \alpha_{ji}^{(t)}$, where $\mu_i^{(t)}$

is the baseline utility for $d_i^{(t)}$ and $\alpha_{ji}^{(t)}$ is the contextual effect of adding d_j^t besides d_i^t.

In our featurized version of the model, $\mu_i^{(t)}$ can be a function of $x_i^{(t)}$; and α_{ji} can depend on both $x_i^{(t)}$ and $x_j^{(t)}$ to reflect the context effects. More specifically, we can define $\mu_i^{(t)} = \psi_1(x_i^{(t)})$ and $\alpha_{ji}^{(t)} = \psi_2(x_i^{(t)}, x_j^{(t)})$, where $\psi_1(.)$ and $\psi_2(.)$ are neural nets. Defining the model this way enables us to apply the model in large scale IR settings. After calculating the baseline and interaction terms of scores for each object, one can derive the following choice probability for $d_i^{(t)}$ as follows:

$$P_i(S^{(t)}) = \frac{\exp\left(f_i^{(t)}(S^{(t)})\right)}{1 + \sum_{l \in S} \exp\left(f_l^{(t)}(S^{(t)})\right)}. \tag{5.2}$$

In the above probability structure, the score of option of "Choosing nothing" is normalized to be 0, which gives the term 1 (exp(0)), in the denominator.

When fixing $S^{(t)}$, the relative ranking among the items of S can be determined by sorting the documents based on values $f_i^{(t)}(S^{(t)})$, or equivalently $P_i(S^{(t)})$. In addition, we use $P_i(S^{(t)})$ later to explicitly optimize on indices like CTR and Expected Reward.

For the sake of completeness, we need to mention that in order to capture more contextual complexity among the documents, one can consider higher order contextual interaction terms when calculating the utility. For instance, $d_j^{(t)}$ and $d_k^{(t)}$ together can have a third-order effect of $\alpha_{jk,i}^{(t)}$ on $f_i^{(t)}(S^{(t)})$ in addition to second-order effects $\alpha_{ji}^{(t)}$ and $\alpha_{ki}^{(t)}$. We call the contextual model capturing the contextual patterns up-to m^{th} order CMNL-Net(m) in this paper. We can define $\psi_m(x_{i_1}^{(t)}, \ldots, x_{i_m}^{(t)})$ to capture the m^{th}-order interaction terms. We denote CMNL-Net(2) by CMNL-Net in this paper.

5.1.1 How to fit the model on a choice data?

We now elaborate the estimation procedure of choice probabilities using choice data. Assume that a given choice data includes T queries.

With the defined notation and choice probability structure, if an item $i^{(t)}$ is chosen (clicked) from set $S^{(t)}$, then $\log P_i(S^{(t)})$ is the log-likelihood of query t. If more

Table 5.1: **Algorithm 1** SGD Algorithm for CMNL-Net.

Algorithm 1 SGD Algorithm
input: training data $\{(x^{(1)}, B^{(1)}, S^{(1)}), \ldots, (x^{(m)}, B^{(m)}, S^{(m)})\}$
Parameter: learning rate η, tolerance rate ϵ, SGD batch size m
Repeat
For $i = 1$ **to** floor(T/m) do
Randomly select a batch m queries, M,
input $(x^{(i)}, B^{(i)}, S^{(i)})$ for queries i in the batch,
compute the gradient of loglikelihood $\nabla\mathcal{LL}()$ of the batch with current ω values,
Update $\omega = \omega - \eta \times \nabla\mathcal{LL}(\{B^{(t)}, S^{(t)}\}_{t\in M})$
end for
Calculate $\mathcal{LL}(\{B^{(t)}, S^{(t)}\}_{t=1}^{T})$,
Until change of $\mathcal{LL}(\{B^{(t)}, S^{(t)}\}_{t=1}^{T})$ is below ϵ,
Output: Neural Net model ω.

than one item (subset $B^{(t)} \subseteq S^{(t)}$)is chosen in query t, we can either consider $\log \sum_{i\in B^{(t)}} P_i(S^{(t)})$ to be the log-likelihood of query t or treat each choice independently and create separate data points for each unique chosen option. In the later case, $\sum_{i\in B^{(t)}} \log P_i(S^{(t)})$ will be the log-likelihood associated with query t. We assume the first case as it leads to less number of log terms in the overall log-likelihood optimization.[1] Also, since it is common to have sparse click data in domains like e-commerce, the data may barely include more than one click in a given query and if this is the case two forms end up being almost the same. Then we use either descent methods like SGD (as Presented in Table 5.1) or solvers for global optimization like BARON (Sahinidis 2017, Tawarmalani and Sahinidis 2005) to optimize the likelihood function for T queries:

$$\mathcal{LL}(\{B^{(t)}, S^{(t)}\}_{t=1}^{T}) = \sum_{t=1}^{T} \log \sum_{i\in B^{(t)}} P_i(S^{(t)}). \tag{5.3}$$

[1]We assume the same for the log-likelihood of the MNL model in our experiments.

5.1.2 Case Studies

In this subsection, we present some toy examples to elaborate on how context created by the set of objects can change the scores as well as the relative relative rankings of objects and consequently affect the Top-K retrieval problem. We show how the CMNL-Net model can capture these.

Effect of Offering Similar items: According to the similarity effect, adding an object to a recommendation set hurts the CTR and score of similar objects to the newly added object more than the dissimilar ones. We will elaborate on this by giving an example from an e-commerce setting.

Under the CMNL-Net model, we can use the distance between the feature vector of two documents as similarity measure between two documents. For instance, we can use the inverse of L_2 (or L_p in general) distance as measure for the similarity of between the pairs of documents. Note that if the distance between the two feature vectors of the documents are is small, then we may conclude that those two documents are similar to each other.

Influence Caused by Attraction Effect: Attraction effect is referred to changes in perceived utility of object caused by adding a dominated object. Here, we mention the famous example of economist.com presented in Sahi (2009) to explore how this effect can interfere with the Top-K retrieval problem.

According to an experiment, the following alternatives were given as the subscription options for economist magazine: (i) Online one-year Subscription with the price of US $59.00, (ii) Print one-year subscription with the price of US $125.00, (iii) Print & web one-year subscription with the price of US $125.00

When these choices were given in an experiment, 16% of the participants, chose the first cheaper option, while 84% chose the third option and nobody chose the second option. Although no one chose the second option, when that alternative was eliminated from the subscription options, 68% chose option 1, while 32% chose option 3 which indicates the positive $\alpha_{23} >> 0$. Given the percentages observed in the experiments, one can capture these changes using the CMNL model (and a deep neural net for estimating the interaction terms in CMNL-Net) when we normalize

the utility of option 1 to be zero:

$$\begin{cases} \mu_1 = 0, \quad \mu_2 = -\infty, \quad \mu_3 \approx -0.75 \\ \alpha_{23} \approx 1.66, \quad \alpha_{12} = \alpha_{13} = \alpha_{21} = \alpha_{31} = \alpha_{32} = 0. \end{cases}$$

The Compromise Among the Choices: When the compromise effect is the dominant type of context effect, users tend to avoid the objects with very high or very low feature values. Here, we mention an example from Wernerfelt (1995) to elaborate more on how the relative ranking can be affected by compromise.

Suppose, a person wants to buy wine and this person belongs to the group of "average Americans" when it comes to wines. When facing two options of a $14 wine and a $20 wine, the person may not be sure which wine to select. But, if this person learns that there exists another $26 option, "$20 bottle looks more middle of the road" and hence this person ends up buying that. Using percentages of purchase in each scenario, we can calculate the baseline and contextual parameters of the CMNL-Net model.

5.1.3 Special Cases of the Model

The CMNL-Net-(m) model has $O(k^m)$ interaction terms, which makes it hard to calculate the scores for larger values of m. Even for $m = 2$, this may lead to overfitting problem when k and n are large and we have limited transaction history to fit the model. Following the approaches of Volkovs and Zemel (2009) and Yousefi Maragheh et al. (2020a), here we introduce some special cases for the CMNL-Net model which try to approximate the full model with lower number of interaction terms. By using these special cases (submodels) one can overcome the overfitting problem in smaller data sets while approximating the contextual patterns among the products.

- **Sub-model K-Net**: In this sub-model, we assume that the interaction term estimator $\alpha_{ji}^{(t)}$ only depends on the features of object $d_i^{(t)}$, $x_i^{(t)}$: $\alpha_{ji}^{(t)} = \alpha_i^{(t)} = \psi_2(x_i^{(t)})$. If this is the case, $f_i^{(t)}(S^{(t)}) = \mu_i^{(t)} + \sum_{j \in S, j \neq i} \alpha_i^{(t)} = \mu_i + (k-1)\alpha_i^{(t)}$. In other words, the utility of each item will be a function of K in Top-K

retrieval problem. As we discuss in the section for experiments, this sub-model may work well when the objects in agiven training data set are similar to each other.

- **Sub-model θ -Net**: When we further restrict the parameter space and make all the contextual parameters equal to a constant value: $\alpha_{ji} = \theta$, $\forall i, j \in \mathcal{N}$.

All the above sub-models restrict the parameter space and try to estimate the contextual patterns among the products by less number of interaction terms. K-Net sub-model has N interaction terms and θ-Net has only 1 interaction term to estimate.

5.2 Top-K Retrieval

After estimation of contextual effects among objects, a natural question will be what subset S of objects to retrieve from a superset $\mathcal{N} = \{1, 2, \ldots, n\}$ of relevant objects in an IR setting when there is a k-cardinality limit for offering. To answer this question we define the following three optimization objectives. Depending on the setting each of them might be of interest. Also, for notational convenience, we eliminate the superscript t for the query in our formulations and focus on the problems for one given query.

5.2.1 Surplus Maximization Problem (SMP)

Customer surplus is defined as the utility (or score) that a user can perceive from the recommended set, $\mathbb{E}[\sum_{i=1}^{k} U_i]$, and is associated with user satisfaction and we may be interested in SMP (see Gallego and Wang 2019). For CMNL-Net-(m), this will be $\sum_{i \in S} f_i(S)$ and Top-K retrieval problem while optimizing on user surplus will be equivalent to the following problem:

$$\max_{S \subseteq \mathcal{N}: |S| = k} \textit{Surplus}(S) = \max_{S \subseteq \mathcal{N}: |S| = k} \sum_{i \in S} f_i(S), \tag{5.4}$$

where $f_i(S) = \mu_i + \sum_{j \in S, j \neq i} \alpha_{ji} + \cdots + \sum_{j_1,\ldots,j_m \neq i \in S} \alpha_{j_1 \ldots, j_{m-1}, i}$ under CMNL-Net-(m). We have the following complexity result for SMP.

Proposition 5.1. *For $m \geq 2$, problem* (5.4) *is NP-hard under the CMNL-Net-(m) utility structure.*

All the proofs are omitted because of the space constraints and presented in the Appendix C. When there is no context effect, i.e. $f_i(S) = \mu_i$, score of each document is not dependent on S. In this case, solving problem (5.4) is trivial: sort the documents based on f_i and select the top-k. However, adding even second-order contextual interaction to the utility makes it intractable.

5.2.2 CTR and Expected Reward Maximization

Optimizing on CTR is linked with customer satisfaction and market share Richardson et al. (2007). Assume that we merely care about the CTR from the offered set S and we want to maximize the chance of clicks. This can be the case in online advertising platforms where the platform gets a reward for each click. Top-K retrieval problem when optimizing on CTR can be formalized as follows:

$$\max_{S \subseteq \mathcal{N}: |S| = k} M(S) = \max_{S \subseteq \mathcal{N}: |S| = k} \sum_{i \in S} P_i(S), \tag{5.5}$$

which is equivalent to minimizing the chance of choosing nothing when a user is offered set S. The following Proposition shows that this problem is NP-hard as well.

Proposition 5.2. *For $m \geq 2$, problem* (5.5) *is NP-hard under the CMNL-Net-(m) probability structure.*

Now, suppose r_i, $i \in N$, is the associated reward with document i. For instance, r_i can be the obtained revenue from product i in a product recommendation system. Note that it may be in the interest of the recommender system to prioritize a more profitable document over more appealing but less profitable documents in or-

der to maximize the expected reward. The following problem, formalizes the Top-K retrieval problem with this objective:

$$\max_{S\subseteq\mathcal{N}:|S|=k} R(S) = \max_{S\subseteq\mathcal{N}:|S|=k} \sum_{i\in S} r_i P_i(S), \tag{5.6}$$

In the above formulation, $r_i P_i(S)$ is the expected reward from recommending product i in set S. Also, note that problem (5.5) is a special instance of problem (5.6) with $r_i = 1, \forall i \in \mathcal{N}$. Thus, we have the following corollary for the NP-hardness of problem (5.6):

Corollary 5.1. *For $m \geq 2$, problem (5.6) is NP-hard under the CMNL-Net-(m) probability structure.*

Since all of the three mentioned combinatorial problems are not polynomially solvable, we present a swapping algorithm to provide high quality solutions for these problems.

We need to mention that problem (5.4) can be reduced to the well-explored problem of binary polynomial optimization problem and thus, much more sophisticated algorithms can be used to circumvent the NP-hardness of that problem (see Appendix C).

5.2.3 Swapping Algorithms

We propose the following Binary Swapping Algorithms with a local search scheme to provide high quality solutions for each of the three mentioned objectives. According to our numerical study on our synthetic data, using our presented swapping algorithm can significantly boost the relevant metrics when compared to the benchmark models. We first start with the 1-swap algorithm.

1-Swap Algorithm: In this heuristic algorithm, we start with a randomly chosen subset S of cardinality k or any reasonable warm start for the algorithm. Denote the solution at iteration l by S_l. Then, in each step of the heuristic, we generate multiple neighbour solutions by swapping the place of one of the objects of the subset, $i_1 \in S_l$,

by one of the documents which is not in the subset, $i_2 \notin S_l$. More formally, if we denote the set of generated subsets from S_l by Ω_l^1, then: $\{S_l \cup \{i_2\}/\{i_1\}\} \in \Omega_l^1$, $\forall i_1 \in S_1, i_2 \notin S_2$.

Note that Ω_l^1 contains $k \times (N-k)$ as $|S_l| = k$, $\forall l$. After generating the candidate sets, the algorithm calculates the objective of interest, $\mathcal{F}(S)$ (can be Surplus, CTR, or Expected Reward) and chooses the subset which gives the highest increase as the solution for the next iteration: $S_{l+1} = \arg\max_{S \in \Omega_t^1} \mathcal{F}(S)$.

The heuristic algorithm stops if all of the generated solutions obtain a lower function value than S_l, and outputs the S_l as the output of the algorithm.

m-Swap Algorithm: This heuristic algorithm is an extension of 1-swap algorithm. In each step of this heuristic, instead of swapping one of the products of the assortment with a product outside of the assortment, we swap m products $\{i_1, i_2, \ldots, i_m\} \subseteq S_l$ with $\{i_{m+1}, i_{m+2}, \ldots, i_{m+m}\} \subseteq \mathcal{N}/S_l$. Everything else is similar to 1-Swap.

5.3 Experiments

In our experiments, we first test our model's prediction performance by comparing it to some of the benchmark listwise ranking/choice models on real data set. We used Mean Reciprocal Rank (MRR), Normalized Distributed Cumulative Gain (NDCG), adn Cross Entropy (CE) as metrics to compare the prediction power of the models in ranking and object CTR estimation (See Radlinski et al. 2008, Chen et al. 2009, and Zhang and Sabuncu 2018 for more on these metrics). We compared our model and some of its special cases with MNL, ListMLE, and Top-K list MLE models. To estimate the baseline parameters, we assume a one-layer neural net taking the features of objects as input. To estimate the interaction terms in the contextual models, we assume a one-layer neural taking the vector of feature differences as iput. More specifically we assume $\alpha_{ji}^{(t)} = C * \beta^T * (x_j - x_i)$, with a hyper parameter C and coefficient vector of β. For estimating the interaction terms $\alpha_{ji}^{(t)}$ in sub-model K-Net we assume a one-layer neural net which uses feature vector $x_i^{(t)}$ as input. As

Table 5.2: MRR, NDCG, CE scores for different models and % improvement when compared to the benchmark models. The first two columns show the scores for Top-1, Top-3 ListMLE. Last three rows show the percent improvement of contextual models w.r.t. best benchmark model.

	Top-1	Top-3	ListMLE	MNL	CMNL-Net	K-Net	θ
MRR	0.6253	0.6108	0.5959	0.6276	**0.6555**	0.6377	0.6026
NDCG	0.7135	0.7022	0.6909	0.7152	**0.7368**	0.7232	0.696
MRR–vs Best					**4.65%**	1.69%	-3.97%
NDCG–vs Best					**3.10%**	1.16%	-2.68%
CE–vs Best					**1.41%**	1.09%	-6.48%

the results confirm considering contextual effects among the objects can provide in better ranking of the products in IR settings.

Then, using a synthetic data set, we compare the quality of suggested solutions in optimizing metrics like Surplus, CTR, and Expected Revenue.

5.3.1 Real Data Experiments

Proprietary click/impression data: For empirical validation, we first consider the proprietary impression and click data from a substitute product recommendation module under the item page. This module tries to recommend substitutes for the anchor item of the page.

The data set includes click/impression information for 1,773,265 sessions and 104 subcategories of products. Normally, for each subcategory, the recommended items are selected among 10 to 100 relevant products (objects) and include tens of features. Each session of this data set includes a subset of products presented to the customers.

We are interested in seeing whether taking the context effects into account really boosts our understanding about the relative rankings among the products and whether there is evidence for existence of context effects in click patterns in Product Recommendation Modules.

To show this, we randomly divide the data for each subcategory of products into

training and testing sets with equal size. We obtain the MLE parameters for each of the models by fitting them on the training sets.

With the obtained parameters for each model, we calculate the score of items in the testing sets. For each impression set, S, in the testing set, we sort the items based on the calculated score under each model. With the obtained ranking, we measure MRR@6, NDCG@6 and the CE Scores for each model. We choose MRR@6 and NDCG@6 since the users are presented with the average of 6 items in this data set. The scores for the contextual models and percent improvement of each model with respect to the benchmark MNL model is reported in Table 5.2.

As can be seen from the table, the CMNL-Net model improves the MRR and NDCG scores by 4.65% and 3.10% respectively. It increases the CE by 1.41%. Note that the CE is a log-based score and even a relatively small change in the CE score mean a significance difference in likelihood of the models. CMNL-Net model performs the best in this metric as well.

The other observation is the bad performance of the sub-model θ-Net . Sub-model θ-Net has only one extra parameter than the benchmark MNL model and this leads to worse scores in our data set. However, according to our results, sub-model K-Net work better than all the benchmark models in all three metrics. Note that K-Net the number of interaction terms are reduced by an order of N when compared to the CMNL-Net model. In fact, K-Net is a simplification of the full model which tries to approximate the full contextual patterns with lower interaction terms. We may expect that this sub-model work well when the products are similar to each other. In other words, when two products with indices k_1 and k_2 are similar, we may expect that they are influenced by other products similarly: $\alpha^{(t)}_{jk_1} \approx \alpha^{(t)}_{jk_2}$, $\forall j \in \mathcal{N}$. If most of the products (or objects in general) in a data set are similar to each other feature-wise, sub-model K-Net with $\alpha^{(t)}_{ji} = \alpha^{(t)}_{i}$, $\forall i, j \in \mathcal{N}$ can be a good approximation of the full model. This may be the reason for the good performance of this sub-model in this data set as the recommended products under each subcategory of our data set are relatively similar to each other.

With the given results in this subsection, one may conclude that considering the changes in relative ranking as a result of context effects may provide better ranking

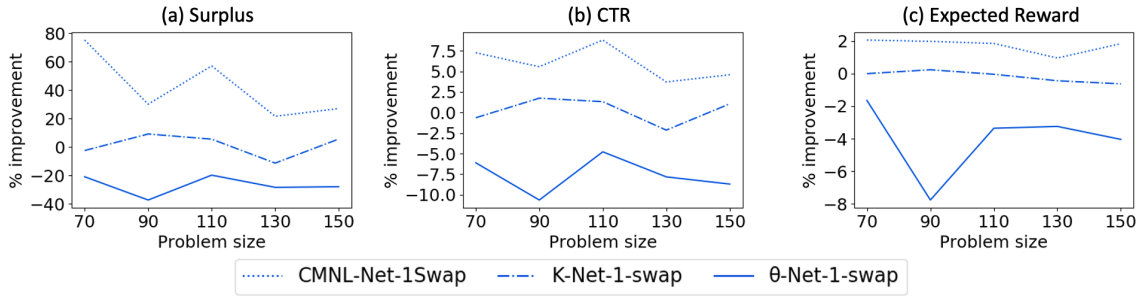

Figure 5.1: Percent improvement in Surplus, CTR, and Expected Reward of CMNL-Net model and its sub-models w.r.t. to the best of the benchmark models

of products in some product recommendation modules.

5.3.2 Experiments on Synthetic Data

In this section, we conduct an experiment using a synthetic data set to elaborate on the efficiency of our approach for optimizing Surplus, CTR, and Expected Reward. First, we randomly generate instances of baseline and pairwise interactions in utility terms with different instance sizes ($n = 70, 90, 110, 130, 150$). For each instant, we generate n objects with 2-dimensional feature vector, $x_i = (x_{i_1}, x_{i_2})$ by uniform sampling. We generate the scores of objects by setting $\psi_1(x_i) = \beta_1 x_{i_1} + \beta_2 x_{i_2}$ and $\psi_2(x_i, x_j) = \beta_3(x_{j_1} - x_{i_1}) + \beta_4(x_{j_2} - x_{i_2})$ and randomly sampling $\beta_1, \beta_2 \sim \mathit{Uniform}(0, 1)$ and $\beta_3, \beta_4 \sim \mathit{Uniform}(0, 1/3)$. Similar to the coupon recommendation module of Alibaba (See Feldman et al. 2018), we assume a cardinality constraint of $k = 6$ for each of these instances. For each problem size, we generate $50,000$ training sets of impressions and clicks.

We fit CMNL-Net and its variations, MNL, ListMLE, and Top-K List-MLE (with k=1,3) in each training set. After fitting the models, we find the suggested top-k objects by each algorithm and calculate the true expected Surplus/CTR/Expected Revenue for the Top-K objects suggested by each algorithm. Note that we select the documents with the highest k scores. Figure 5.1 shows the percent improvement in Surplus, CTR, and Expected Reward of CMNL-Net model and its sub-models

w.r.t. to the best of the benchmark models. As it can be confirmed from the figure, when having an underlying contextual interaction between the objects, our approach can significantly boost the relevant metrics. As we generated the feature vectors of objects randomly, the objects are not similar to each other and thus we may not expect K-Net and θ-net to work well for the data and the figure confirms this. For detailed presentation of results please refer to Appendix C.

Chapter 6

Conclusion and Future Work

In this thesis, we discuss how the product recommendation modules can be further optimized in different settings while considering the context interactions between the products. We specifically, study recommendation optimization problem (i) when the choice data is sparse and under a single-choice outcome assumption, (ii) when the underlying choice data is dense and with a multi-choice outcome assumption, and (iii) when the input data is featurized with an IR setting.

In Chapter 3, we propose an extension of the MNL model, the "Contextual MNL" model, to approximate context effects. Specifically, in this model we assume that the perceived utility of an individual item in an assortment depends on what items are offered beside it, and adding/removing other items from/to the assortment changes this perceived utility. We also show that by using this model we can potentially approximate different types of context effects. We test the descriptive and predictive power of this choice model on a real data set. Our results indicate that for this data set, our proposed choice model significantly enhances the descriptive and predictive scores when compared to several widely used choice models. This may suggest that the context effects play a major role in how customers decide when choosing from a set of available products in some business settings, and considering models that incorporate these effects enables us to better predict the choice outcomes.

In addition to the empirical validation of the data, we show that the assortment optimization problem and the click through rate optimization problem are NP-hard when the underlying choice model is the CMNL. We also present some of the special cases of the model which makes the AOP and CTROP polynomially solvable. To be more specific, we show the polynomially solvability of AOP and CTROP when (i) decoy-target patterns exist among the offered products, (ii) the utility of products is a function of the cardinality of the assortment, or the items are similar to each other or (iii) context effects caused by any item has similar synergistic effect on other items and the items have similar revenues. Interestingly, these special cases of the CMNL model have high predictive and descriptive scores according to our empirical validation and hence possess both tractability of the AOP as well as high data modeling power. We present sufficient conditions for the monotonicity and submodularity of the objective of the mentioned combinatorial optimization problems

that are easily verifiable and can provide us some approximation guarantees. Finally, we propose some efficient heuristics for solving the NP-hard assortment optimization problem under the general CMNL model. We test these heuristics on randomly generated instances and show that they output near-optimal solutions in a reasonable amount of time.

In Chapter 4, we extend the idea of Chapter 3 to a multi-choice setting and we propose a utility based listwise logistic regression model, which is applicable in estimating the context effects in dense impression/click (or impression/purchase) data sets with a multi-choice outcome assumption. We perform an extensive empirical study on on 70 impression/click data chosen from diverse categories of products and show the effciency of the model and some special case of it in metrics like AIC, BIC, cross entropy, MRR, and NDCG. We prove the NP-hardness of AOP under the general CL model, and show that when some specific types of contextual interactions are dominant in the data, the AOP is tractable. We also devise efficient heuristics and numerically test them.

In Chapter 5, we study the Top-K retrieval problem in presence of context effects. In the problem setting presented, we assume that the score of an object in an IR setting depends on the features of objects presented besides it in addition to its own features. We propose an estimator that can approximate this dependency in objects' scores through mapping their features to contextual interaction terms by using an underlying neural net structure. In addition, we study the complexity of problem of retrieving Top-K objects in presence of such interactions and under three different combinatorial objectives: (i) User surplus Maximization (ii) CTR maximization, and (iii) Expected reward maximization. We show that the problem is NP-hard under all three objectives. In order to deal with the NP-hardness, we propose swapping algorithms. We empirically validate our used estimation model and swapping algorithm by comparing it to benchmark models on real and synthetic data sets. Our results show that incorporating for context effects can boost ranking related metrics like MRR, NDCG and Cross Entropy. This can be an indication for significance of considering the context effects in IR settings for some data sets. Also, our analysis suggests that using our approach in Top-K retrieval can significantly enhance the

quality of recommendations when compared with benchmark approaches.

One thing worth mentioning here is the importance of the similarity effect in affecting the choice behavioral patterns in our large-scale data sets in e-commerce. In almost all of our empirical results, similarity effect is the dominant type of context effect, since the special cases used for modeling this effect obtain high prediction scores. This may emphasise the importance of the diversification while offering relevant products in product recommendation systems. As confirmed by our results, the recommendation algorithms which use merely each item's feature values to calculate a relevance/utility score may end up offering products with similar feature values and this may affect the indices like CTR and customer satisfaction negatively. Thus, designing and implementing recommender systems which are aware of context effects like similarity effect may help in diversifying the assortments more and improve indices like CTR.

Our research can be continued in several directions. We considered the contextual version of the MNL and Logistic Regression model and it will be interesting to design and test the performance of contextual versions of other widely used choice models like Markov Chain Choice Model, Nested Logit, and Mixed MNL.

It will also be interesting to study the effect of methods like regularization on the prediction performance of the contextual models and contextual interaction parameters. We conjecture regularization can help in both increasing the prediction power of the models as well as increasing the convergence speed of iterative optimization methods like BB or SGD used for finding MLEs as result of increase in the convexity of the relevant negative log-likelihood functions. In addition, we conjecture that regularization can help in identifiability of the models and studying the relevant questions seems interesting.

One other thing that we find interesting to explore is extending our proposed approach in this thesis to multi-unit multi-choice setting. Note that in our multi-choice setting, each customer can choose multiple products; however, we assume a binary choice outcome for each product. This binary choice-outcome is true when modeling clicks by a user in recommendation module as each user either clicks on each link or she does not click. It is also true when the customers buy at most

one unit from each item. For instance, customers tend to buy at most one unit of each product in general merchandise categories like home appliances, electronics, or fashion industry (this is true according to our observations of real transaction data sets, and also it is confirmed in the literature, (see Zhu et al. 2014)) even if they buy multiple products.

However, there are cases where the choice outcome of each product is not binary and customers may end up buying more than one unit from each item. This can be true in transaction data sets of categories like household essentials for example. To model multi-unit multi-purchase settings, one possible direction is to assume a Poisson distribution on the number of products purchased in each time unit from each item. To incorporate context effects, we may assume that the Poisson rate parameter to be dependent on the set offered to the customer.

All in all, we hope different models proposed in this thesis and the relevant empirical studies as well as the recommendation optimization results will stimulate in future studies to incorporate complex interactions of products when modeling demand and choice behavior of customers and push the industry to consider revenue/inventory management systems that are aware of these effects. The results of this research may be found useful in modeling demand alteration patterns in the time of supply chain/demand disruptions (like pandemics) as well, where stock-outs happen more frequently. Detecting and predicting these patterns are of higher importance in the times of supply chain/demand disruptions and more accurately estimating them may lead to better management of these disruptions.

Appendix A

Proofs of Chapter 3

Theorem 3.1. When the underlying probability structure is CMNL, the CTROP (3.5) is NP-hard.

Proof. Before stating the proof of this theorem, we state the following result from Atamtürk and Gómez (2017).

For any nonnegative vectors $\alpha, V \in \mathbb{R}^N$, and any strictly concave function $g : \mathbb{R} \to \mathbb{R}$, problem

$$\max_{X \in \{0,1\}^N} -\alpha^T X + g(V^T X)$$

is NP-hard.

As the this result establishes the NP hardness for any strictly concave function $g(.)$, it holds true for $g(u) = \log(u)$.

Now, we are ready to prove the NP-hardness result for our problem of interest. We will show the NP-hardness of CTROP under the CMNL model by a reduction from

$$\max_{X \in \{0,1\}^N} -\alpha^T X + \log(V^T X).$$

As $v_i \geq 0$, $\forall i \in \mathcal{N}$, we can obtain $\mu_i \in \mathbb{R}$, $\forall i \in \mathcal{N}$ such that:

$$v_i = e^{\mu_i + \sum_{j \in \mathcal{N}} \alpha_j}.$$

Therefore, we have the equivalence of the following problems:

$$\begin{aligned}
&\max_{X\in\{0,1\}^N} -\alpha^T X + \log(V^T X) \\
\iff &\max_{X\in\{0,1\}^N} \prod_{j=1}^N e^{-\alpha_j x_j} \sum_{i=1}^N v_i x_i \iff \max_{X\in\{0,1\}^N} \prod_{j=1}^N e^{-\alpha_j x_j} \sum_{i=1}^N x_i e^{(\mu_i+\sum_{j=1}^N \alpha_j)} \\
\iff &\max_{X\in\{0,1\}^N} \sum_{i=1}^N x_i e^{(\mu_i+\sum_{j=1}^N \alpha_j)-\sum_{j=1}^N \alpha_j x_j} \iff \max_{X\in\{0,1\}^N} \sum_{i=1}^N x_i e^{\mu_i+\sum_{j=1}^N (1-x_j)\alpha_j},
\end{aligned}$$

which in turn are equivalent to:

$$\max_{X\in\{0,1\}^N} \frac{\sum_{i=1}^N x_i e^{\mu_i+\sum_{j=1}^n (1-x_j)\alpha_j}}{1+\sum_{i=1}^N x_i e^{\mu_i+\sum_{j=1}^N (1-x_j)\alpha_j}}.$$

This is an instance of the CTROP under the CMNL model with $\alpha_{ji} = \alpha_j$, $\forall i, j \in \mathcal{N}$. Thus, this proves the NP-hardness of the CTROP under the CMNL model. □

Theorem 3.2. Under the CMNL model with utility and probability structure defined in (3.1) and (3.2) respectively, if condition **(C1)** is satisfied, then Θ is identifiable.

Proof. Given the structure of the $\mathbb{Q}$-matrix in **(C1)**, let us rewrite the the log-likelihood function, by splitting the sum in periods $m = 1, \ldots, M_1$ where everything is offered (first part of the $\mathbb{Q}$-matrix) and periods $m = M_1 + 1, \ldots, M$ where one item is missing (second part of the $\mathbb{Q}$-matrix):

$$\begin{aligned}
\ell(^-, \mathbb{A}) &= \ell(^-, \mathbb{A})_1^{M_1} + \ell(^-, \mathbb{A})_{M_1+1}^{M} \\
&= \sum_{m=1}^{M_1} \left\{ -\kappa^{(m)} \log(1 + \sum_{k=1}^N \exp(\mu_k)) + \sum_{j=1}^N z_j^{(m)} \mu_j \right\} \\
&+ \sum_{m=M_1+1}^{M} \left\{ -\kappa^{(m)} \log(1 + \sum_{k=1;k\neq i_m}^N \exp(\mu_k + \alpha_{i_m k})) \right. \\
&\qquad \left. + \sum_{j=1;j\neq i_m}^N z_j^{(m)} (\mu_j + \alpha_{i_m j}) \right\}.
\end{aligned} \tag{A.1}$$

Where $\ell(^{-}, \mathbb{A})_1^{M_1}$ is the log-likelihood function for the first M_1 periods and $\ell(^{-}, \mathbb{A})_{M_1+1}^{M}$ is the log-likelihood for the rest of the periods. Also, i_m is the item which is missing in period m (note that none of the items are missing in periods $m = 1, \ldots, M_1$).

$\ell(^{-}, \mathbb{A})_{M_1+1}^{M}$ can be further decomposed according to the item which is missing in the offered set. Denote the set of periods which item i is missing by $M^{(i)} \subset \{M_1 + 1, \ldots, M\}$. Then, $\ell(^{-}, \mathbb{A})_{M_1+1}^{M}$ can be rewritten as:

$$\begin{aligned}\ell(^{-}, \mathbb{A})_{M_1+1}^{M} = \sum_{i=1}^{N} \sum_{m \in M^{(i)}} \Bigg\{ -\kappa^{(m)} \log(1 + \sum_{k=1; k \neq i_m}^{N} \exp(\mu_k + \alpha_{i_m k})) \\ + \sum_{j=1; j \neq i_m}^{N} z_j^{(m)} (\mu_j + \alpha_{i_m j}) \Bigg\}\end{aligned} \tag{A.2}$$

By having the Equations (A.1) and (A.2), one can write the first order conditions for finding the maximum likelihood estimators as follows:

$$\begin{aligned}\frac{\partial \ell(^{-}, \mathbb{A})}{\partial \alpha_{ij}} &= \sum_{m \in M^{(i)}} \Bigg\{ -\kappa^{(m)} \exp(\mu_j + \alpha_{ij}) + z_j^{(m)} \Big(1 + \sum_{k=1; k \neq i}^{N} \exp(\mu_k + \alpha_{ik})\Big) \Bigg\} \\ &= 0 \qquad i, j = 1, \ldots, N \qquad i \neq j\end{aligned} \tag{A.3}$$

$$\begin{aligned}\frac{\partial \ell(^{-}, \mathbb{A})}{\partial \mu_j} = \sum_{m=1}^{M_1} \Bigg\{ -\kappa^{(m)} \exp(\mu_j) + z_j^{(m)} \Big(1 + \sum_{k=1}^{N} \exp(\mu_k)\Big) \Bigg\} \\ + \sum_{i=1; i \neq j}^{N} \sum_{m \in M^{(i)}} \Bigg\{ -\kappa^{(m)} \exp(\mu_j + \alpha_{i_m j}) \\ + z_j^{(m)} \Big(1 + \sum_{k=1; k \neq i_m}^{N} \exp(\mu_k + \alpha_{i_m k})\Big) \Bigg\} = 0 \\ j = 1, \ldots, N\end{aligned} \tag{A.4}$$

By doing some straightforward calculations one can see that system of equations

(A.3) and (A.4) can be solved uniquely and the following estimates of the parameters can be obtained:

$$\hat{\mu}_j = \ln\left(\frac{\sum_{m=1}^{M_1} z_j^{(m)}}{\sum_{m=1}^{M_1} z_0^{(m)}}\right), \qquad j = 1, \ldots, N \tag{A.5}$$

$$\hat{\alpha}_{ij} = \ln\left(\frac{\sum_{m\in M^{(i)}} z_j^{(m)}}{\sum_{m\in M^{(i)}} z_0^{(m)}}\right) - \hat{\mu}_j, \qquad i, j = 1, \ldots, N \tag{A.6}$$

Thus, in this case, identifiability is guaranteed.

□

Theorem 3.3. Under the CMNL model with utility and probability structure defined in (3.1) and (3.2) respectively, when the matrix $\mathbb{A}$ is upper triangular and the $\mathbb{Q}$ matrix satisfies Condition **(C2)**, then the model is identifiable.

Proof. Given the specific structure of $\mathbb{Q}$-matrix in **(C2)**, and similar to the idea of the proof of Theorem (3.2), we rewrite the log-likelihood of the data by splitting the terms of log-likelihood function into N different sums. The first sum includes periods $m = 1, \ldots, M_1$ when everything if offered; the second sum includes periods $m = M_1 + 1, \ldots, M_2$ when everything but item 1 is offered; the third sum includes periods $m = M_2+1, \ldots, M_3$ when everything but items 1 and 2 are offered. Similarly, the i-th sum includes periods $m = M_{i-1} + 1, \ldots, M_i$ where everything but items $\{1, 2, \ldots, i-1\}$ are offered, for $i = 2, \ldots, N$. If we write the log-likelihood function in this form and the first order conditions, with a similar process as the proof of Theorem (3.2), we can get the following unique maximum likelihood estimates for the data:

$$\begin{aligned}
\hat{\mu}_j &= \ln\left(\frac{\sum_{m=1}^{M_1} z_j^{(m)}}{\sum_{m=1}^{M_1} z_0^{(m)}}\right), && \forall j = 1, \ldots, N \\
\hat{\alpha}_{ij} &= \ln\left(\frac{\sum_{M_i+1}^{M_{i+1}} z_j^{(m)}}{\sum_{M_i+1}^{M_{i+1}} z_0^{(m)}}\right) - \ln\left(\frac{\sum_{M_{i-1}+1}^{M_i} z_j^{(m)}}{\sum_{M_{i-1}+1}^{M_i} z_0^{(m)}}\right), && \forall i, j = 1, \ldots, N, i < j
\end{aligned} \tag{A.7}$$

Where $M_0 = 1$. Thus, the result of theorem follows.

□

Proposition 3.1. Under the CMNL model, when the $\mathbb{Q}$ matrix satisfies Condition **(C2)**, then the model is locally identifiable.

Proof. The steps for the proof here are similar to these of Theorem 3.3.

□

Proposition 3.2. The AOP (and consequently the CTROP) under "P_i", $i \in \mathcal{N}$, is polynomially solvable.

Proof. According to the result in Section 3.2. of Davis et al. (2013), we know that any problem of the form:

$$\begin{aligned} \max_{X \in \{0,1\}^N} \ & \frac{f(X)}{g(X)} \\ \text{s.t. } & BX = b, \end{aligned} \tag{A.8}$$

with B being a Totally Unimodular (TU) matrix; and $f(X)$ and $g(X)$ being linear in the binary vector X is polynomially solvable. Indeed, with this linear fraction structure in the objective function and the presence of TU constraints, the LP relaxation is tight, i.e., it gives the same optimal objective value of (A.8). The AOP for the MNL model is a special case of problem (A.8) and hence polynomially solvable even in presence of TU constraints like the cardinality constraints.

Now, WLOG, assume $i = 1$ (i is the index of P_i). The AOP in this case is equivalent to the following problem:

$$\max_{X \in \{0,1\}^N} \frac{\sum_{i \in \mathcal{N}} x_i r_i e^{\mu_i + x_1 \alpha_{1i}}}{1 + \sum_{i \in \mathcal{N}} x_i e^{\mu_i + x_1 \alpha_{1i}}}.$$

When $x_1 = 0$, then, the problem reduces to the AOP with $N - 1$ products under the MNL model, and hence, it is polynomially solvable as stated.

When $x_1 = 1$, then, the problem reduces to the following problem:

$$\max_{X\in\{0,1\}^N} \frac{r_1 e^{\mu_1} + \sum_{i\in\mathcal{N}, i\neq 1} x_i r_i e^{\mu_i+\alpha_{1i}}}{1 + e^{\mu_1} + \sum_{i\in\mathcal{N}, i\neq 1} x_i e^{\mu_i+\alpha_{1i}}},$$

In the above problem both the numerator and the denominator are linear in the availability vector X, and hence, again polynomially solvable. This concludes the desired result. Also, the pollynomially solvability holds even in presence of TU constraints like the cardinality constraint. □

Proposition 3.3. Under S1-Syn, if $r(S, \mu, A) \leq r_i$, $\forall S \in 2^{\mathcal{N}}$ and $i \notin S$, then $\mathcal{N}$ is an optimal assortment.

Proof. First, note that the objective function of the AOP when having $\alpha_{ji} = \alpha_j$, $\forall i, j \in \mathcal{N}$ can be rewritten as follows:

$$\begin{aligned}\frac{\sum_{j\in S} r_j e^{\mu_j+\sum_{l\notin S}\alpha_l}}{1+\sum_{j\in S} e^{\mu_j+\sum_{l\notin S}\alpha_l}} &= \frac{\sum_{j\in S} r_j e^{(\mu_j+\sum_{l\in\mathcal{N}}\alpha_l)-\sum_{l\in S}\alpha_l}}{1+\sum_{j\in S} e^{(\mu_j+\sum_{l\in\mathcal{N}}\alpha_l)-\sum_{l\in S}\alpha_l}} \\ &= \frac{\sum_{j\in S} r_j v_j}{e^{\sum_{j\in S}\alpha_j} + \sum_{j\in S} v_j},\end{aligned} \tag{A.9}$$

where $v_j = e^{\mu_j+\sum_{l\in\mathcal{N}}\alpha_l}$.

Now, Denote the optimal set maximizing $r(., \mu, A)$ by S^* and the optimal objective value by Z^*: $Z^* = r(S^*, \mu, A)$.

We want to show $S^* = \mathcal{N}$. Assume $S^* \neq \mathcal{N}$ and denote the index of the item with smallest revenue which is missing from S^* by index l, i.e., $l = \min\{r_i : i \notin S^*\}$. Under the assumption of the proposition,

$$Z^* \leq r_l. \tag{A.10}$$

Using Equation (A.9);

$$Z^* = \frac{\sum_{j\in S^*} r_j v_j}{e^{\sum_{j\in S^*}\alpha_j} + \sum_{j\in S^*} v_j}.$$

By rewriting the above equation we get

$$Z^* = \frac{\sum_{j\in S^*}(r_j - Z^*)v_j}{e^{\sum_{j\in S^*}\alpha_j}}. \tag{A.11}$$

Since $\alpha_l \leq 0$, the inequality (A.10) implies that:

$$Z^* = \frac{\sum_{j\in S^*}(r_j - Z^*)v_j}{e^{\sum_{j\in S^*}\alpha_j}} \leq \frac{\sum_{j\in S^*\cup\{l\}}(r_j - Z^*)v_j}{e^{\sum_{j\in S^*\cup\{l\}}\alpha_j}} \tag{A.12}$$

Thus, using (A.11) and (A.12):

$$Z^* \leq \frac{\sum_{j\in S^*\cup\{l\}}(r_j - Z^*)v_j}{e^{\sum_{j\in S^*\cup\{l\}}\alpha_j}},$$

which implies that $S^* \cup \{l\}$ is also an optimal set. By repeating this argument we can conclude $S^* = \mathcal{N}$ is an optimal solution.

□

Lemma 3.1. Assume $r_{\max} = \max_{i\in\mathcal{N}} r_i$ and $r_{\min} = \min_{i\in\mathcal{N}} r_i$. Under the S1-Syn, if the revenues of the products are close to each other such that

$$\frac{r_{\max} - r_{\min}}{r_{\min}} \leq \min_{S\subseteq\mathcal{N}} \frac{P_0(S)}{1 - P_0(S)},$$

then $r(S, \mu, A) \leq r_i$, $\forall S \in 2^{\mathcal{N}}$ and $i \notin S$.

Proof. Under the assumption of the lemma,

$$\frac{r_{\max} - r_{\min}}{r_{\min}} \leq \frac{P_0(S)}{1 - P_0(S)}, \qquad \forall S \subseteq \mathcal{N}.$$

Rewriting the right side of the above inequality, we get:

$$\frac{r_{\max} - r_{\min}}{r_{\min}} \leq \frac{1}{\sum_{i \in S} e^{\mu_i + \sum_{j \notin S} \alpha_j}}, \qquad \forall S \subseteq \mathcal{N},$$

which by rearranging the terms is equivalent to:

$$\frac{\sum_{i \in S} r_{\max} e^{\mu_i + \sum_{j \notin S} \alpha_j}}{1 + \sum_{i \in S} e^{\mu_i + \sum_{j \notin S} \alpha_j}} \leq r_{\min}, \qquad \forall S \subseteq \mathcal{N}. \tag{A.13}$$

As $r_{\max} \geq r_i$, $\forall i \in \mathcal{N}$, we have

$$\frac{\sum_{i \in S} r_i e^{\mu_i + \sum_{j \notin S} \alpha_j}}{1 + \sum_{i \in S} e^{\mu_i + \sum_{j \notin S} \alpha_j}} \leq \frac{\sum_{i \in S} r_{\max} e^{\mu_i + \sum_{j \notin S} \alpha_j}}{1 + \sum_{i \in S} e^{\mu_i + \sum_{j \notin S} \alpha_j}} \qquad \forall S \subseteq \mathcal{N}, \tag{A.14}$$

The left hand side of (A.14) is $r(S, \mu, A)$. Thus using (A.13) and (A.14), we have

$$r(S, \mu, A) \leq r_{\min} \qquad \forall S,$$

which establishes the result of the lemma. □

Lemma 3.2. Under S1-Syn of the CMNL model, if $r(\mathcal{N}, \mu, A) \leq \min\{r_i\}_{i \in \mathcal{N}}$, then $r(S, \mu, A) \leq r_i$, $\forall S \in 2^{\mathcal{N}}$ and $i \notin S$.

Proof. Using Equation (A.9) from the proof of Proposition 2, if $r(\mathcal{N}, \mu, A) \leq r_{\min}$, we have:

$$\frac{\sum_{j \in \mathcal{N}} r_j v_j}{e^{\sum_{j \in \mathcal{N}} \alpha_j} + \sum_{j \in \mathcal{N}} v_j} \leq r_{\min}.$$

Rewriting the above inequality we get:

$$\sum_{j \in \mathcal{N}} (r_j - r_{\min}) v_j \leq r_{\min} \times e^{\sum_{j \in \mathcal{N}} \alpha_j}.$$

As $\alpha_j \leq 0$, $\forall j \in \mathcal{N}$, and $r_j \geq r_{\min}$, we can get the following:

$$\sum_{j \in S} (r_j - r_{\min}) v_j \leq r_{\min} \times e^{\sum_{j \in S} \alpha_j} \qquad \forall S \subseteq \mathcal{N}.$$

Rewriting the above term, we get:

$$\frac{\sum_{j \in S} r_j v_j}{e^{\sum_{j \in S} \alpha_j} + \sum_{j \in S} v_j} \leq r_{\min},$$

or $r(S, \mu, A) \leq r_{\min}$, which completes the proof.

□

Proposition 3.4. The AOP (and consequently the CTROP) is polynomially solvable under S2 of the CMNL model.

Proof. Similar to the proof of the previous proposition, we prove polynomially solvability by showing the equivalence of the AOP under this special case of CMNL model to problem (A.8). The AOP under this case is equivalent to

$$\max_{X \in \{0,1\}^N} \frac{\sum_{i \in \mathcal{N}} r_i x_i e^{\mu_i + \sum_{j \in \mathcal{N}} \alpha_i (1 - x_j)}}{1 + \sum_{i \in \mathcal{N}} x_i e^{\mu_i + \sum_{j \in \mathcal{N}} a_i (1 - x_j)}},$$

which is equivalent to

$$\max_{X \in \{0,1\}^N} \frac{\sum_{i \in \mathcal{N}} r_i x_i e^{\mu_i + (N - \sum_{j \in \mathcal{N}} x_j) \alpha_i}}{1 + \sum_{i \in \mathcal{N}} x_i e^{\mu_i + (N - \sum_{j \in \mathcal{N}} x_j) \alpha_i}}.$$

By splitting the feasible space region of the above problem into N subclasses of $\sum_{j \in \mathcal{N}} x_j = c$, $c = 1, \ldots, N$, we get at most N of the following sub-problems:

$$\max_{X \in \{0,1\}^N} \frac{\sum_{i \in \mathcal{N}} r_i x_i e^{\mu_i + (N-c) \alpha_i}}{1 + \sum_{i \in \mathcal{N}} x_i e^{\mu_i + (N-c) \alpha_i}}$$
$$\text{s.t.} \sum_{i \in \mathcal{N}} x_i = c.$$

Each of these sub-problems are polynomially solvable as they are of the form of problem (A.8). This concludes the desired result of the proposition. Note that the results hold in presence of the cardinality constraint as well. □

Lemma 3.3. The obejctive of CTROP under the CMNL-Syn model, with $\alpha_{ji} \leq 0$ $\forall i, j \in \mathcal{N}$ is monotone.

Proof. Let $M(S, \mu, A)$ be the CTR of the assortment S with parameters $A = [\alpha_{ji}]_{i,j=1,\ldots,N}$ and $\mu = [\mu_j]_{j=1,\ldots,N}$:

$$\begin{aligned} M(S, \mu, A) &= \frac{\sum_{l \in S} e^{\mu_l + \sum_{j \notin S} \alpha_{jl}}}{1 + \sum_{l \in S} e^{\mu_l + \sum_{j \notin S} \alpha_{jl}}} \\ &= 1 - \frac{1}{1 + \sum_{l \in S} e^{\mu_l + \sum_{j \notin S} \alpha_{jl}}}. \end{aligned}$$

Note that $\forall S$ and $\forall l \in S$, $e^{\mu_l + \sum_{j \notin S} \alpha_{jl}} \leq e^{\mu_l + \sum_{j \notin S \cup \{i\}} \alpha_{jl}}$, since $\alpha_{jl} \leq 0$, $\forall j, l \in \mathcal{N}$. Hence,

$$\sum_{l \in S} e^{\mu_l + \sum_{j \notin S} \alpha_{jl}} \leq \sum_{l \in S} e^{\mu_l + \sum_{j \notin S \cup \{i\}} \alpha_{jl}},$$

which implies that, $\sum_{l \in S} e^{\mu_l + \sum_{j \notin S} \alpha_{jl}} \leq \sum_{l \in S \cup \{i\}} e^{\mu_l + \sum_{j \notin S \cup \{i\}} \alpha_{jl}}$ and

$$M(S, \mu, A) \leq M(S \cup \{i\}, \mu, A).$$

Thus, the monotonocity is established. □

Lemma 3.4. When $\alpha_{ji} = \alpha_j \leq 0$ in CMNL model, if $r(S, \mu, A) \leq r_i$, $\forall S \cup \{i\} \in \mathcal{F}$ and $\forall i \notin S$, then the objective of AOP is monotone.

Proof. As shown previously, the objective function of the AOP for S1 can be rewritten as follows:

$$r(S, \mu, A) = \frac{\sum_{l \in S} r_l v_l}{e^{\sum_{l \in S} \alpha_l} + \sum_{l \in S} v_l},$$

where $v_l = e^{\mu_l + \sum_{j=1}^{N} \alpha_j}$. In other words, the above equation shows that when $\alpha_{ji} = \alpha_j$, we can assume that adding an item has no effect on the preference of other items and merely affects the preference of the no-purchase alternative.

By the assumption given, we have $\forall S \in \mathcal{F}$ and $i \notin S$:

$$\begin{aligned} & r(S, \mu, A) \leq r_i \\ \Longrightarrow & \frac{\sum_{l \in S} r_l v_l}{e^{\sum_{l \in S} \alpha_l} + \sum_{l \in S} v_l} \leq r_i. \end{aligned}$$

By replacing $R_S = \sum_{l \in S} r_l v_l$, $V_S = \sum_{l \in S} v_l$, and $v_0^S = e^{\sum_{l \in S} \alpha_l}$, we get:

$$\begin{aligned} & \frac{R_S}{v_0^S + V_S} \leq r_i \\ \Longrightarrow & \frac{R_S v_i}{v_0^S + V_S} \leq r_i v_i \\ \Longrightarrow & (1 + \frac{v_i}{v_0^S + V_S}) R_S \leq R_S + r_i v_i \\ \Longrightarrow & \frac{v_0^S + V_S + v_i}{v_0^S + V_S} R_S \leq R_S + r_i v_i. \end{aligned}$$

Since $\alpha_i \leq 0$, $\forall i \in \mathcal{N}$, the above inequality implies

$$\frac{R_S}{v_0^S + V_S} \leq \frac{R_S + r_i v_i}{v_0^S e^{\alpha_i} + V_S + v_i},$$

or equivalently,

$$r(S, \mu, A) \leq r(S \cup \{i\}, \mu, A),$$

which implies monotonocity. □

Theorem 3.4. Denote the the attractiveness of product i under special case S1-Syn, and when set S is offered by $v_i^{(S)}$, i.e. $v_i^{(S)} = e^{\mu_i + \sum_{k \notin S} \alpha_k}$. If $\sum_{i \in S} v_i^{(S)} \geq 1$, $\forall S \in \mathcal{F}$, then the objective of CTROP is submodular, and consequently, adding the items in a greedy manner approximates the optimal solution of CTROP under a cardinality constraint with a $(1 - 1/e)$ approximation factor.

Proof. First, denote $V_S = (\sum_{k\in S} e^{\mu_k})(e^{\sum_{k\in\mathcal{N}}\alpha_k})$, and $V_0^S = e^{\sum_{k\in S}\alpha_k}$. We have:

$$\begin{aligned} &(\sum_{k\in S} e^{\mu_k})(e^{\sum_{k\in\mathcal{N}}\alpha_k}) \geq e^{\sum_{k\in S}\alpha_k} \\ \Longleftrightarrow &(\sum_{k\in S} e^{\mu_k})(e^{\sum_{k\notin S}\alpha_k}) \geq 1 \\ \Longleftrightarrow &\sum_{k\in S} v_i^{(S)} \geq 1, \end{aligned} \tag{A.15}$$

which confirms $V_S \leq V_0^S$ if and only if $\sum_{i\in S} v_i^{(S)} \geq 1$. Now, we show that if $\sum_{i\in S} v_i^{(S)} \geq 1$, $\forall S$, then the objective of the CTROP is submodular.

Note that, a set function $f(.)$ is submodular if the following property holds:

$$f(S\cup\{i,j\}) - f(S\cup\{j\}) \leq f(S\cup\{i\}) - f(S),$$

$\forall S \subseteq \mathcal{N}$, $i,j \notin S$, and $S\cup\{i,j\} \in \mathcal{F}$.

Thus, $M(S,\mu,A)$ is submodular if

$$M(S\cup\{i,j\},\mu,A) - M(S\cup\{j\},\mu,A) \leq M(S\cup\{i\},\mu,A) - M(S,\mu,A).$$

This is equivalent to:

$$\frac{V_S + v_i + v_j}{V_0^S A_i A_j + V_S + v_i + v_j} - \frac{V_S + v_j}{V_0^S A_j + V_S + v_j} \leq \frac{V_S + v_i}{V_0^S A_i + V_S + v_i} - \frac{V_S}{V_0^S + V_S},$$

or:

$$\frac{V_0^S A_j}{V_0^S A_j + V_S + v_j} - \frac{V_0^S A_i A_j}{V_0^S A_i A_j + V_S + v_i + v_j} \leq \frac{V_0^S}{V_0^S + V_S} - \frac{V_0^S A_i}{V_0^S A_i + V_S + v_i}.$$

Dividing both sides by V_0^S:

$$\frac{A_j}{V_0^S A_j + V_S + v_j} - \frac{A_i A_j}{V_0^S A_i A_j + V_S + v_i + v_j} \leq \frac{1}{V_0^S + V_S} - \frac{A_i}{V_0^S A_i + V_S + v_i}.$$

By multiplying both sides in the denominators and after a tedious algebraic calculations and rearrangement of terms we get:

$$
\begin{aligned}
&(1-A_i)(1-A_j)(V_SV_SV_S + V_SV_Sv_i + V_SV_Sv_j + V_Sv_iv_j) + V_0^SV_0^SA_iA_j^2(1-A_i)\\
&+ (V_S - V_0^S)(v_iv_j(1-A_i)) + v_iv_i(V_S(1-A_j) + V_0^SA_j)\\
&+ v_iv_j(v_i + v_j + V_0^SA_iA_j)\\
&+ (v_i(1-A_j) + v_j(1-A_i))(V_SV_S - V_0^SV_0^SA_iA_j)\\
&+ V_S(2v_iv_j - V_0^SV_0^SA_iA_j(1-A_i)) \geq 0,
\end{aligned}
\tag{A.16}
$$

which holds if all the terms are nonnegative.

The terms in the first and second lines of inequality (A.16) are nonnegative as $0 \leq A_i \leq 1$, $\forall i \in \mathcal{N}$. The third line is nonnegative.

Also, if $\sum_{i\in S} v_i^{(S)} \geq 1$, $\forall S \in \mathcal{F}$, then by (A.15) we have $V_S \geq V_0^S$ and hence $V_SV_S - V_0^SV_0^SA_iA_j \geq 0$, which concludes the nonnegativity of the forth line.

In addition: $\alpha_j \leq 0$, $\forall j \in \mathcal{N}$ and hence $V_0^S = e^{\sum_{k\in S}\alpha_k} \leq 1$. Also, by replacing $S = \{i\}$ and $S = \{j\}$ in $\sum_{i\in S} v_i^S$ we get:

$$
\begin{cases}
e^{\mu_i}e^{\sum_{k\in\mathcal{N};k\neq i}\alpha_k} \geq 1\\
e^{\mu_j}e^{\sum_{k\in\mathcal{N};k\neq j}\alpha_k} \geq 1
\end{cases},
$$

which by the defined notation is equivalent to:

$$
\begin{cases}
v_i \geq A_i\\
v_j \geq A_j.
\end{cases}
$$

Finally this implies $v_iv_j \geq A_iA_j$, and consequently results in the nonegativity of the last line.

Thus, the submodularity is established. Since when $\alpha_j \leq 0$, the objective of CTROP is monotone, we conclude the result of the theorem from Proposition 4.3 of Nemhauser et al. (1978). □

Theorem 3.5 Denote $\max_{i\in\mathcal{N}} \mu_i = \mu_{\max}$, $\min_{i\in\mathcal{N}} \mu_i = \mu_{\min}$, and $\max_{i\in\mathcal{N}} \alpha_i = \alpha_{\max}$ when the underlying model is S1. If

$$\log(1 + \frac{2e^{\mu_{\min}}}{ne^{\mu_{\max}}}) \geq \alpha_{\max},$$

then the function $M^{'}()$ is submodular, and there exists a randomized linear time 1/3-approximation for the CTROP problem.

Proof. We have:

$$\begin{aligned} &\log(1 + \frac{2e^{\mu_{\min}}}{ne^{\mu_{\max}}}) \geq \alpha_{\max} \\ \Longleftrightarrow &1 + \frac{2e^{\mu_{\min}}}{ne^{\mu_{\max}}} \geq e^{\alpha_{\max}} \\ \Longleftrightarrow &\frac{2e^{\mu_{\min}}}{e^{\alpha_{\max}} - 1} \geq ne^{\mu_{\max}}. \end{aligned}$$

From the last line of the above expression we can get:

$$\frac{e^{\mu_j}}{e^{\alpha_j} - 1} + \frac{e^{\mu_i}}{e^{\alpha_i} - 1} \geq \sum_{k\in S} e^{\mu_k} \qquad \forall S \in \mathcal{F}; i, j \in \mathcal{N},$$

Denote: $b_i = \frac{1}{e^{\alpha_i}}$, $c_i = e^{\mu_i}$, and $C_S = \sum_{k\in S} e^{\mu_k}$. The above inequality implies

$$\frac{c_j b_i}{1 - b_i} + \frac{c_i b_j}{1 - b_j} \geq C_S \qquad \forall S \in \mathcal{F}; i, j \in \mathcal{N}.$$

By multiplying both sides in $(1 - b_i)(1 - b_j)$ we have:

$$C_S b_i b_j + c_i b_i b_j + c_j b_i b_j - C_S b_i - c_j b_i \leq C_S b_j + c_i b_j - C_S \qquad \forall S \in \mathcal{F}; i, j \in \mathcal{N}.$$

Rearranging the terms and multiplying by $B_S = e^{\sum_{k \notin S} \alpha_k}$:

$$(C_S + c_i + c_j)(B_S b_i b_j) - (C_S + c_j)(B_S b_i) \leq (C_S + c_i)(B_S b_j) - C_S B_S \qquad \forall S \in \mathcal{F}; i, j \in \mathcal{N}.$$

By replacing the defined notation we obtain the submodularity of $M^{'}(.)$:

$$M^{'}(S \cup \{i, j\}) - M^{'}(S \cup \{j\}) \leq M^{'}(S \cup \{i\}) - M^{'}(S) \qquad \forall S \in \mathcal{F}; i, j \in \mathcal{N},$$

and as $M^{'}()$ is nonnegative, there exist a randomized linear time (1/2)-approximation for it, according to Theorem 1.2. of Buchbinder et al. (2015). Thus, there exists a $\frac{1/2}{1+1/2} = (1/3)$-approximation for the CTROP objective. □

Appendix B

Proofs of Chapter 4

Theorem 4.1. Problems (4.4) and (4.5) are NP-hard.

Proof. Atamtürk and Gómez (2017) show that the problem of finding the binary vector $x = (x_1, \ldots, x_n)$ that maximizes the objective of the following problem is NP-hard, when assuming vectors $\lambda = (\lambda_1, \ldots, \lambda_n)^T, \alpha = (\alpha_1, \ldots, \alpha_n)^T$ are non-negative and g is a strictly concave function:

$$\max_{x \in \{0,1\}^n} -\lambda^T x + g(\alpha^T x). \tag{B.1}$$

Since (B.1) is NP-hard, the following version of this problem with reduced feasibility space of $\sum_{i=1}^n x_i = k$ is also NP-hard:

$$\max_{x \in \{0,1\}^n : \sum_{i=1}^n x_i = k} -\lambda^T x + g(\alpha^T x), \tag{B.2}$$

since to solve Problem (B.1) we can solve $n+1$ problem of type of (B.2):

$$\max_{x \in \{0,1\}^n} -\lambda^T x + g(\alpha^T x) = \max_{k=0,\ldots,n} \Big\{ \max_{x \in \{0,1\}^n : \sum_{i=1}^n x_i = k} -\lambda^T x + g(\alpha^T x) \Big\}.$$

Define $g(x) = -C/(1 + \exp(x))$ (with $C > 0$) which is strictly concave for $x > 0$. Since (B.2) is NP-hard for any strictly concave function g, the following problem is

NP-hard (with $\lambda_i = C\zeta_i$, and $0 < \zeta_i < 1$):

$$\max_{x\in\{0,1\}^n:\sum_{i=2}^n x_i=k-1,x_1=1} \frac{-C}{1+\exp(\sum_{i=2}^n \alpha_i x_i)} - \sum_{i=2}^n C\zeta_i x_i. \tag{B.3}$$

We can rewrite Problem (B.3) as follows:

$$\begin{aligned}
&\max_{x\in\{0,1\}^n:\sum_{i=2}^n x_i=k-1,x_1=1} \frac{-C}{1+\exp(\sum_{i=2}^n \alpha_i x_i)} - \sum_{i=2}^n C\zeta_i x_i \\
=& -C\times k + \max_{x\in\{0,1\}^n:\sum_{i=2}^n x_i=k-1,x_1=1} C\Big(1 - \frac{1}{1+\exp(\sum_{i=2}^n \alpha_i x_i)}\Big) + \sum_{i=2}^n C(1-\zeta_i)x_i.
\end{aligned}$$

Thus, solving (B.3) is equivalent to solving the following:

$$\max_{x\in\{0,1\}^n:\sum_{i=2}^n x_i=k-1,x_1=1} \frac{C\exp(\sum_{i=2}^n \alpha_i x_i)}{1+\exp(\sum_{i=2}^n \alpha_i x_i)} + \sum_{i=2}^n C(1-\zeta_i)x_i \tag{B.4}$$

However, the above problem is an special instance of the CTROP with:

$$\begin{cases}
r_i = C & \forall i = 1,\dots,n \\
\alpha_{ji} = 0 & \forall i = 2,\dots,n \\
\alpha_{j1} = \alpha_j & \forall i = 2,\dots,n \\
\mu_i = \log(1/\lambda_i - 1) \text{ with } 0 < \lambda_i < 1 & i = 2,\dots,n \\
\mu_1 = 0
\end{cases}$$

Thus, Problem (4.4) is NP-hard.

□

Lemma 4.1 Under the CL model, we can construct instances where the ratio between the revenues generated by the best revenue-ordered assortment and the optimal assortment is arbitrarily small.

Proof. Let us denote the optimal objective of the AOP by z^*. Here, we present

an instance of the AOP that revenue ordered assortments generate objectives with values less than βz^*, where $\beta \to 0$.

Consider instance with $\mathcal{N} = \{1, 2\}$, with $r_1 = 2$, $r_2 = 1$, and $\mu_2 = \alpha_{21} = 0$. The revenue generated by all possible feasible solutions are as follows:

$$\begin{cases} R((0,0), \mu, A) = 0, \\ R((1,0), \mu, A) = \dfrac{2\exp(\mu_1)}{1+\exp(\mu_1)}, \\ R((0,1), \mu, A) = 0.5, \\ R((1,1), \mu, A) = \dfrac{2\exp(\mu_1)}{1+\exp(\mu_1)} + \dfrac{\exp(\alpha_{12})}{1+\exp(\alpha_{12})}. \end{cases}$$

As we can see the generated revenues by revenue-ordered assortments $(1, 1)$ and $(1, 0)$ can converge to zero as $\mu_1, \alpha_{12} \to -\infty$. □

Propositions 4.1. When $\alpha_{ji} \geq 0$, $\forall i, j \in \mathcal{N}$, $\mathcal{N}$ is an optimal assortment for problem (4.4).

Proof. We prove the result by showing that Problem 4.4 is monotone. Assume subsets S_1 and S_2 such that $S_1 \subseteq S_2$. We have

$$u_i(S_1) = \mu_i + \sum_{j \in S_1, j \neq i} \alpha_{ji} \leq \mu_i + \sum_{j \in S_2, j \neq i} \alpha_{ji} = u_i(S_2),$$

where the inequality follows from non-negativity of the α_{ji} parameters. From this, we can conclude $P_i(S_1) \leq P_i(S_2)$, $\forall i \in S_1$, since $P_i(S) = \exp(u_i(S))/(1+\exp(u_i(S)))$ is increasing in $u_i(S)$. Thus, we can conclude:

$$R(\mu, A, r, S_1) = \sum_{i \in S_1} r_i P_i(S_1) \leq \sum_{i \in S_2} r_i P_i(S_2) = R(\mu, A, r, S_2),$$

which proves the monotonosity. □

Proposition 4.2. The AOP is polynomially solvable under special case S2 of the CL model.

Proof. The utility of each item i under special case S_2, can be written as follows:

$$u_i(S) = \mu_i + \sum_{j \in S, j \neq i} \alpha_i = \mu_i + (|S|-1)\alpha_i.$$

Thus, for all the sets with the same cardinality, the utility of each item is the same, and if we divide the solution space into subspaces of subsets with the same cardinality ($|S| = 1, \ldots, n$), the selection probability of each item, $P_i(S)$ will be the same for all S in that subspace.

Because of this, the best solution in the subspace of assortments with cardinality $|S| = k$, can be obtained by selecting k items with highest values of $r_i P_i(S)$ which can be done in polynomial time. As there are n subspace of assortments, each including the assortments with the same cardinality, finding the best $S \subseteq \mathcal{N}$ requires polynomial time. □

Proposition 4.3. AOP is polynomially solvable under special case d_i; or when there are $O(1)$ many trend-setting items with dominant contextual interactions on the other items.

Proof. First, suppose there is only one trend-setting item, and WLOG assume that item is the one with index n. In this case, the utilities of other items, $i = 1, \ldots, n-1$, depends on whether we are offering item n or not. Thus, when dividing the solution space into two subspaces of (i) assortments including item n, and (ii) assortments not including item n, the utilities (and choice probabilities) of each item in each sub space will be the same.

In this case, the optimal assortment of subspace (i) is $\mathcal{N}$ and the optimal assortment of (ii) is $\mathcal{N} - n$. Similarly, if there are $O(1)$ trend-setting items, we can divide solution space into $2^{O(1)}$ subspaces and find the optimal assortment in each subspace in polynomial time. □

Appendix C

Notes and Proofs of Chapter 5

C.1 Proofs

Proposition 5.1. For $m \geq 2$, problem (5.4) is NP-hard under the CMNL-Net-(m) utility structure.

Proof. Note that we can denote any subset S of objects with an equivalent binary vector $Y = (y_1, \ldots, y_n)$, where $y_i = 1$ if $i \in S$, and $y_i = 0$ o.w. First, we show SMP is NP-hard for $m = 2$ and when there is no cardinality constraint on the number of objects by reduction from the partition problem.

In the partition problem, there is a set of positive integers $\{a_1, a_2, \ldots, a_n\}$ and we want to find a subset of this set which sum of its elements is $(\sum_{i \in \mathcal{N}} a_i)/2$. In other words, we want to find $Y^* = (y_1^*, \ldots, y_n^*)$ such that:

$$\sum_{i \in \mathcal{N}} a_i y_i^* = \frac{1}{2}(\sum_{i \in \mathcal{N}} a_i).$$

Partition problem is NP-hard Garey and Johnson (1990). Now, consider the following optimization problem:

$$\max_{Y \in \{0,1\}^N} -(\sum_{i \in \mathcal{N}} a_i y_i - \frac{1}{2} \sum_{i \in \mathcal{N}} a_i)^2 + \frac{1}{4}(\sum_{i \in \mathcal{N}} a_i)^2. \tag{C.1}$$

Problem (C.1) has an optimal objective value of $\frac{1}{4}(\sum_{i\in\mathcal{N}} a_i)^2$ if and only if $\sum_{i\in\mathcal{N}} a_i y_i = (\sum_{i\in\mathcal{N}} a_i)/2$. By rearranging the terms in problem (C.1) we get the following equivalent problem:

$$\max_{X\in\{0,1\}^N} \sum_{i\in\mathcal{N}} (a_i(\sum_{j\in\mathcal{N}} a_j) - a_i^2)y_i + \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}, j\neq i} -a_i a_j y_i y_j.$$

The above problem is an instance of the SMP with $\mu_i = a_i(\sum_{j\in\mathcal{N}} a_j) - a_i^2$ and $\alpha_{ji} = -a_j a_i$; $\forall i, j \in \mathcal{N}$. If we can solve this problem in polynomial time, we can answer whether there exist a desired subset in the partition problem in polynomial time. Thus, the NP-hardness of SMP for $m = 2$ and with no cardinality constraint follows.

Now, note that we can divide the feasible solution space for SMP problem with no cardinality constraint ($Y = (y_1, \ldots, y_n) \in \{0,1\}^n$) to n partition, each including subsets with equal cardinality($Y = (y_1, \ldots, y_n) \in \{0,1\}^n$, $\sum_{i\in\mathcal{N}} y_i = k$, $\forall k = 1, \ldots, n$.

Thus, if SMP is polynomially solvable under each partition of feasible solution space, then, we can conclude polynomially solvablity of SMP with no cardinality constraint. Thus, SMP with constraint on the cardinality subsets is NP-hard as well. The NP-hardness for $m > 2$ follows by having a similar argument on an instance of this problem with $\mu_i = a_i(\sum_{j\in\mathcal{N}} a_j) - a_i^2$ and $\alpha_{ji} = -a_j a_i$; $\forall i, j \in \mathcal{N}$, and $\alpha_{j_1 \ldots j_{l-1}, i} = 0$, $\forall j_1, \ldots, j_{l-1}, i \in \mathcal{N}$ and $\forall l > 2$. □

Proposition 5.2. For $m \geq 2$, problem (5.5) is NP-hard under the CMNL-Net-(m) probability structure.

Proof. Yousefi Maragheh et al. (2020a) shows the NP-hardness of the CTR maximization problem with $m = 2$ and when there is no cardinality constraint(Theorem 1).

Similar to the argument presented in Proposition 1, we can divide the feasible solution space for CTR maximization problem with no cardinality constraint ($Y = (y_1, \ldots, y_n) \in \{0,1\}^n$) to n partition, each including subsets with equal cardinality($Y = (y_1, \ldots, y_n) \in \{0,1\}^n$, $\sum_{i\in\mathcal{N}} y_i = k$, $\forall k = 1, \ldots, n$; and the NP-hardness of the

unconstrained problem results in the NP-hardness of the Top-K retrieval problem. The NP-hardness for $m > 2$ follows since the model with $m = 2$ is an instance of the model with $m > 2$ with $\alpha_{j_1 \ldots j_{l-1},i} = 0$, $\forall j_1, \ldots, j_{l-1}, i \in \mathcal{N}$ and $\forall l > 2$.

□

C.2 Note on the equivalence of SMP to maximizing the binary polynomial problem

Let us denote a given subset S with binary vector $Y = (y_1, \ldots, y_n)$ as discussed in the proof of Proposition 1. We can rewrite problem the SMP as follows:

$$\begin{aligned} \max_{y \in \{0,1\}^n} \sum_{i \in N} y_i f_i(S) = \max_{y \in \{0,1\}^n} \sum_{i \in N} y_i (\mu_i + \sum_{j \in N, j \neq i} y_j \alpha_{ji} \\ + \cdots + \sum_{j_1, \ldots, j_{m-1} \neq i \in S} (\alpha_{j_1 \ldots, j_{m-1}, i} \prod_{t=1}^{m-1} y_{j_t})), \end{aligned} \tag{C.2}$$

which is a Binary Polynomial Optimization problem. For $m = 2$ this reduces to the binary quadratic problem. See Bertsimas and Shioda (2009) and Kochenberger et al. (2014) for more on how to solve the binary quadratic problem.

C.3 More on Numerical Results

The following three tables show the (i) average customer surplus, (ii) average CTR, and (iii) average expected reward obtained by different models in our numerical study.

Table C.1: Numerical results on average surplus for each recommendation algorithms and problem size

n	70	90	110	130	150
CMNL-Net-1Swap	7.9707	5.2500	6.2446	5.5440	5.5208
K-Net-1-swap	4.437	4.4025	4.1962	4.0432	4.5882
θ-Net-1-swap	3.5968	2.5300	3.1896	3.2661	3.1312
MNL	4.5505	4.0351	3.8728	4.5607	4.3128
ListMLE	4.546	4.0351	3.8695	4.5607	4.3128
Top-1	4.5505	4.0351	3.8728	4.5607	4.3128
Top-3	4.5362	4.0351	3.9779	4.5607	4.3471
CMNL-Net-Naive	3.4879	2.0055	2.9559	2.7219	2.3713

Table C.2: Numerical results on average CTR for each recommendation algorithms and problem size

n	70	90	110	130	150
CMNL-Net-1Swap	0.8716	0.8393	0.8603	0.8468	0.8461
K-Net-1-swap	0.8074	0.8089	0.8011	0.799	0.8175
θ-Net-1-swap	0.7629	0.7105	0.7529	0.7526	0.7386
MNL	0.8126	0.7951	0.7877	0.8165	0.8082
ListMLE	0.8125	0.7951	0.7881	0.8165	0.8082
Top-1	0.8126	0.7951	0.7877	0.8165	0.8082
Top-3	0.8122	0.7951	0.7907	0.8165	0.809
CMNL-Net-Naive	0.7532	0.637	0.737	0.7219	0.6907

Table C.3: Numerical results on average expected reward for each recommendation algorithms and problem size

n	70	90	110	130	150
CMNL-Net-1Swap	0.7953	0.7652	0.7952	0.7874	0.7786
K-Net1-swap	0.7792	0.7521	0.7804	0.7765	0.7597
θ-Net-1swap	0.7664	0.6920	0.7545	0.7546	0.7336
MNL	0.7792	0.7503	0.7807	0.7799	0.7645
ListMLE	0.4089	0.3766	0.3396	0.3944	0.4276
Top-1	0.3959	0.3766	0.3435	0.3944	0.4276
Top-3	0.3990	0.3766	0.3803	0.3944	0.4245
CMNL-Net-Naive	0.3710	0.3247	0.3460	0.4019	0.3363

Bibliography

Adomavicius, Gediminas, Alexander Tuzhilin. 2015. Context-aware recommender systems. Francesco Ricci, Lior Rokach, Bracha Shapira, eds., *Recommender Systems Handbook*. Springer, 191–226. doi:10.1007/978-1-4899-7637-6_6. URL `https://doi.org/10.1007/978-1-4899-7637-6_6`.

Aggarwal, Charu C, et al. 2018. Neural networks and deep learning. *Springer* **10** 978–3.

Atamtürk, Alper, Andrés Gómez. 2017. Maximizing a class of utility functions over the vertices of a polytope. *Operations Research* **65**(2) 433–445.

Aurier, Philippe, Victor Mejia. 2014. Multivariate logit and probit models for simultaneous purchases: Presentation, uses, appeal and limitations. *Recherche et Applications en Marketing (English Edition)* **29**(2) 75–94.

Barzilai, Jonathan, Jonathan M Borwein. 1988. Two-point step size gradient methods. *IMA journal of numerical analysis* **8**(1) 141–148.

Berlyne, Daniel E. 1960. *Conflict, arousal, and curiosity.*. McGraw-Hill Book Company.

Bertsimas, Dimitris, Romy Shioda. 2009. Algorithm for cardinality-constrained quadratic optimization. *Computational Optimization and Applications* **43**(1) 1–22.

Bettman, James R, Mary Frances Luce, John W Payne. 1998. Constructive consumer choice processes. *Journal of consumer research* **25**(3) 187–217.

Blanchet, Jose, Guillermo Gallego, Vineet Goyal. 2016. A markov chain approximation to choice modeling. *Operations Research* **64**(4) 886–905.

Boatwright, Peter, Joseph C Nunes. 2001. Reducing assortment: An attribute-based approach. *Journal of marketing* **65**(3) 50–63.

Borle, Sharad, Peter Boatwright, Joseph B Kadane, Joseph C Nunes, Shmueli Galit. 2005. The effect of product assortment changes on customer retention. *Marketing science* **24**(4) 616–622.

Boyd, J Hayden, Robert E Mellman. 1980. The effect of fuel economy standards on the us automotive market: an hedonic demand analysis. *Transportation Research Part A: General* **14**(5-6) 367–378.

Buchbinder, Niv, Moran Feldman, Joseph Seffi, Roy Schwartz. 2015. A tight linear time

(1/2)-approximation for unconstrained submodular maximization. *SIAM Journal on Computing* **44**(5) 1384–1402.

Burnham, Kenneth P, David R Anderson. 2002. A practical information-theoretic approach. *Model selection and multimodel inference, 2nd ed. Springer, New York* **2**.

Buzzell, Robert D, Bradley T Gale, Ralph GM Sultan. 1975. Market share-a key to profitability. *Harvard business review* **53**(1) 97–106.

Cao, Zhe, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, Hang Li. 2007. Learning to rank: from pairwise approach to listwise approach. Zoubin Ghahramani, ed., *Machine Learning, Proceedings of the Twenty-Fourth International Conference (ICML 2007), Corvallis, Oregon, USA, June 20-24, 2007*, *ACM International Conference Proceeding Series*, vol. 227. ACM, 129–136. doi:10.1145/1273496.1273513. URL `https://doi.org/10.1145/1273496.1273513`.

Cardell, N Scott, Frederick C Dunbar. 1980. Measuring the societal impacts of automobile downsizing. *Transportation Research Part A: General* **14**(5-6) 423–434.

Chen, Wei, Tie-Yan Liu, Yanyan Lan, Zhi-Ming Ma, Hang Li. 2009. Ranking measures and loss functions in learning to rank. *Advances in Neural Information Processing Systems* **22** 315–323.

Choplin, Jessica M, John E Hummel. 2005. Comparison-induced decoy effects. *Memory & cognition* **33**(2) 332–343.

Davis, James, Guillermo Gallego, Huseyin Topaloglu. 2013. Assortment planning under the multinomial logit model with totally unimodular constraint structures. *Available at https://people.orie.cornell.edu/jmd388/publications/MNLConstr.pdf* .

Davis, James M, Guillermo Gallego, Huseyin Topaloglu. 2014. Assortment optimization under variants of the nested logit model. *Operations Research* **62**(2) 250–273.

Farias, Vivek F, Srikanth Jagabathula, Devavrat Shah. 2013. A nonparametric approach to modeling choice with limited data. *Management science* **59**(2) 305–322.

Feldman, Jacob, Laura Wagner, Huseyin Topaloglu. 2020. Assortment planning under the multi-purchase mnl model. URL `https://www.abstractsonline.com/pp8/?utm_campaign=2020%20Annual&utm_medium=email&_hsenc=p2ANqtz-_mfhGR2XUZ4FwAlPNbLMoLtHY95QFQE-FDBoJyy2y1231X-QZ4FX-kdNhpjYVYaZ3mCElDLfwS8_NIkodBoZ2nNq7tiA&_hsmi=96067111&utm_content=96067111&utm_source=hs_email&hsCtaTracking=54468d0b-b5d2-4de7-bbd4-c5c1ac354a56%7Ce961609f-c080-4a9a-b6b9-7c935b218577#!/9022/presentation/2609`. Informs Annual Meeting.

Feldman, Jacob, Dennis Zhang, Xiaofei Liu, Nannan Zhang. 2018. Taking assortment optimization from theory to practice: Evidence from large field experiments on alibaba. *Available at SSRN* .

Feng, Guiyun, Xiaobo Li, Zizhuo Wang. 2018. On substitutability and complementarity in discrete choice models. *Operations Research Letters* **46**(1) 141–146.

Fletcher, Roger. 2005. On the barzilai-borwein method. *Optimization and control with applications*. Springer, 235–256.

Gallego, Guillermo, Richard Ratliff, Sergey Shebalov. 2014. A general attraction model and sales-based linear program for network revenue management under customer choice. *Operations Research* **63**(1) 212–232.

Gallego, Guillermo, Huseyin Topaloglu, et al. 2019. *Revenue management and pricing analytics*, vol. 209. Springer.

Gallego, Guillermo, Ruxian Wang. 2019. Threshold utility model with applications to retailing and discrete choice models. *Available at SSRN 3420155* .

Garey, Michael R., David S. Johnson. 1990. *Computers and Intractability; A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., USA.

Han, Shaoning, Andrés Gómez, Oleg A Prokopyev. 2019. Assortment optimization and submodularity. *Available at http://pitt.edu/ agomez/publications/Assortment.pdf* .

Hart, William E., Carl D. Laird, Jean-Paul Watson, David L. Woodruff, Gabriel A. Hackebeil, Bethany L. Nicholson, John D. Siirola. 2017. *Pyomo–optimization modeling in python*, vol. 67. 2nd ed. Springer Science & Business Media.

Hart, William E, Jean-Paul Watson, David L Woodruff. 2011. Pyomo: modeling and solving mathematical programs in python. *Mathematical Programming Computation* **3**(3) 219–260.

Huber, Joel, John W Payne, Christopher Puto. 1982. Adding asymmetrically dominated alternatives: Violations of regularity and the similarity hypothesis. *Journal of consumer research* **9**(1) 90–98.

Immorlica, Nicole, Brendan Lucier, Jieming Mao, Vasilis Syrgkanis, Christos Tzamos. 2021. Combinatorial assortment optimization. *ACM Transactions on Economics and Computation (TEAC)* **9**(1) 1–34.

Kahn, Barbara E. 1995. Consumer variety-seeking among goods and services-an integrative review. *Journal of Retailing and Consumer Services* **3**(2) 139–148.

Kahn, Barbara E, Donald R Lehmann. 1991. Modeling choice among assortments. *Journal of Retailing* **67**(3) 274–299.

Kalyanam, Kirthi, Sharad Borle, Peter Boatwright. 2007. Deconstructing each item's category contribution. *Marketing Science* **26**(3) 327–341.

Kochenberger, Gary, Jin-Kao Hao, Fred Glover, Mark Lewis, Zhipeng Lü, Haibo Wang, Yang Wang. 2014. The unconstrained binary quadratic programming problem: a survey. *Journal of Combinatorial Optimization* **28**(1) 58–81.

Liu, Tie-Yan. 2011. *Learning to rank for information retrieval*. Springer Science & Business Media.

Lo, Venus, Huseyin Topaloglu. 2019. Assortment optimization under the multinomial logit model with product synergies. *Operations Research Letters* **47**(6) 546–552.

Luce, R Duncan. 1959. *Individual choice behavior, a theoretical analysis*. John Wiley and Sons.

Lyu, Chengyi, Stefanus Jasin, Sajjad Najafi, Huanan Zhang. 2021. Assortment optimization with multi-item basket purchase under multivariate mnl model. *Available at SSRN 3818886* .

Manchanda, Puneet, Asim Ansari, Sunil Gupta. 1999. The "shopping basket": A model for multicategory purchase incidence decisions. *Marketing science* **18**(2) 95–114.

Mao, Huiqiang, Yanzhi Li, Chenliang Li, Di Chen, Xiaoqing Wang, Yuming Deng. 2020. Pars: Peers-aware recommender system. *Proceedings of The Web Conference 2020*. 2606–2612.

McFadden, D. 1974. Conditional logit analysis of qualitative choice behavior. *Frontiers in Econometrics* 105–142.

McFadden, Daniel, Kenneth Train. 2000. Mixed mnl models for discrete response. *Journal of applied Econometrics* **15**(5) 447–470.

Nemhauser, George L, Laurence A Wolsey, Marshall L Fisher. 1978. An analysis of approximations for maximizing submodular set functions—i. *Mathematical programming* **14**(1) 265–294.

Orhun, A Yeşim. 2009. Optimal product line design when consumers exhibit choice set-dependent preferences. *Marketing Science* **28**(5) 868–886.

Radlinski, Filip, Madhu Kurup, Thorsten Joachims. 2008. How does clickthrough data reflect retrieval quality? *Proceedings of the 17th ACM conference on Information and knowledge management*. 43–52.

Read, Timothy RC, Noel AC Cressie. 2012. *Goodness-of-fit statistics for discrete multivariate data*. Springer Science & Business Media.

Richardson, Matthew, Ewa Dominowska, Robert Ragno. 2007. Predicting clicks: estimating the click-through rate for new ads. *Proceedings of the 16th international conference on World Wide Web*. 521–530.

Rooderkerk, Robert P, Harald J Van Heerde, Tammo HA Bijmolt. 2011. Incorporating context effects into a choice model. *Journal of Marketing Research* **48**(4) 767–780.

Rusmevichientong, Paat, David Shmoys, Chaoxu Tong, Huseyin Topaloglu. 2014. Assortment optimization under the multinomial logit model with random choice parameters. *Production and Operations Management* **23**(11) 2023–2039.

Sahi, Shalini Kalra. 2009. Predictably irrational: the hidden forces that shape our decisions. *Vision* **13**(3) 88.

Sahinidis, N. V. 2017. *BARON 17.8.9: Global Optimization of Mixed-Integer Nonlinear Programs,* User's Manual.

Seetharaman, PB, Siddhartha Chib, Andrew Ainslie, Peter Boatwright, Tat Chan, Sachin

Gupta, Nitin Mehta, Vithala Rao, Andrei Strijnev. 2005. Models of multi-category choice behavior. *Marketing Letters* **16**(3) 239–254.

Simonson, Itamar. 1989. Choice based on reasons: The case of attraction and compromise effects. *Journal of consumer research* **16**(2) 158–174.

Simonson, Itamar, Amos Tversky. 1992. Choice in context: Tradeoff contrast and extremeness aversion. *Journal of marketing research* **29**(3) 281–295.

Talluri, Kalyan, Garrett Van Ryzin. 2004. Revenue management under a general discrete choice model of consumer behavior. *Management Science* **50**(1) 15–33.

Tang, Huajin, Kay Chen Tan, Zhang Yi. 2007. *Neural networks: computational models and applications*, vol. 53. Springer Science & Business Media.

Tawarmalani, M., N. V. Sahinidis. 2005. A polyhedral branch-and-cut approach to global optimization. *Mathematical Programming* **103** 225–249.

Taylor, Manning. 2021. Playstation 5 vs xbox series x: Data reveals the most popular console. URL `https://gamingsmart.com/playstation-5-vs-xbox-series-x/`. Accessed: 2021-05-02.

Thorndike, Edward L. 1920. A constant error in psychological ratings. *Journal of applied psychology* **4**(1) 25–29.

Train, Kenneth E. 2009. *Discrete choice methods with simulation*. Cambridge university press.

Trueblood, Jennifer S, Scott D Brown, Andrew Heathcote, Jerome R Busemeyer. 2013. Not just for consumers: Context effects are fundamental to decision making. *Psychological science* **24**(6) 901–908.

Tulabandhula, Theja, Deeksha Sinha, Prasoon Patidar. 2020. Multi-purchase behavior: Modeling and optimization. *Available at SSRN https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3626788* .

Tversky, Amos. 1972. Elimination by aspects: A theory of choice. *Psychological review* **79**(4) 281.

Tversky, Amos, Daniel Kahneman. 1991. Loss aversion in riskless choice: A reference-dependent model. *The quarterly journal of economics* **106**(4) 1039–1061.

Tversky, Amos, Itamar Simonson. 1993. Context-dependent preferences. *Management science* **39**(10) 1179–1189.

Volkovs, Maksims, Richard S. Zemel. 2009. Boltzrank: learning to maximize expected ranking gain. Andrea Pohoreckyj Danyluk, Léon Bottou, Michael L. Littman, eds., *Proceedings of the 26th Annual International Conference on Machine Learning, ICML 2009, Montreal, Quebec, Canada, June 14-18, 2009*, *ACM International Conference Proceeding Series*, vol. 382. ACM, 1089–1096. doi:10.1145/1553374.1553513. URL `https://doi.org/10.1145/1553374.1553513`.

Wang, Ruxian. 2018. When prospect theory meets consumer choice models: Assortment and pricing management with reference prices. *Manufacturing & Service Operations Management* **20**(3) 583–600.

Wernerfelt, Birger. 1995. A rational reconstruction of the compromise effect: Using market data to infer utilities. *Journal of Consumer Research* **21**(4) 627–633.

Williams, Huw CWL. 1977. On the formation of travel demand models and economic evaluation measures of user benefit. *Environment and planning A* **9**(3) 285–344.

Xia, Fen, Tie-Yan Liu, Hang Li. 2009a. Statistical consistency of top-k ranking. *Advances in Neural Information Processing Systems*. Citeseer, 2098–2106.

Xia, Fen, Tie-Yan Liu, Hang Li. 2009b. Statistical consistency of top-k ranking. Yoshua Bengio, Dale Schuurmans, John D. Lafferty, Christopher K. I. Williams, Aron Culotta, eds., *Advances in Neural Information Processing Systems 22: 23rd Annual Conference on Neural Information Processing Systems 2009. Proceedings of a meeting held 7-10 December 2009, Vancouver, British Columbia, Canada*. Curran Associates, Inc., 2098–2106. URL `http://papers.nips.cc/paper/3879-statistical-consistency-of-top-k-ranking`.

Xia, Fen, Tie-Yan Liu, Jue Wang, Wensheng Zhang, Hang Li. 2008. Listwise approach to learning to rank: theory and algorithm. William W. Cohen, Andrew McCallum, Sam T. Roweis, eds., *Machine Learning, Proceedings of the Twenty-Fifth International Conference (ICML 2008), Helsinki, Finland, June 5-9, 2008*, *ACM International Conference Proceeding Series*, vol. 307. ACM, 1192–1199. doi: 10.1145/1390156.1390306. URL `https://doi.org/10.1145/1390156.1390306`.

Xiang, Biao, Daxin Jiang, Jian Pei, Xiaohui Sun, Enhong Chen, Hang Li. 2010. Context-aware ranking in web search. Fabio Crestani, Stéphane Marchand-Maillet, Hsin-Hsi Chen, Efthimis N. Efthimiadis, Jacques Savoy, eds., *Proceeding of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2010, Geneva, Switzerland, July 19-23, 2010*. ACM, 451–458. doi: 10.1145/1835449.1835525. URL `https://doi.org/10.1145/1835449.1835525`.

Yousefi Maragheh, Reza, Xin Chen, James Davis, Jason Cho, Sushant Kumar. 2020a. Choice modeling and assortment optimization in the presence of context effects. *Available at SSRN* .

Yousefi Maragheh, Reza, Xin Chen, Luyi Ma, Chuanwei Ruan, Jason Cho, Sushant Kumar, Kannan Achan. 2020b. Set dependent ranking model: Evidence for contextual click patterns in walmart.com data. URL `https://www.abstractsonline.com/pp8/?utm_campaign=2020%20Annual&utm_medium=email&_hsenc=p2ANqtz-_mfhGR2XUZ4FwAlPNbLMoLtHY95QFQE-FDBoJyy2y1231X-QZ4FX-kdNhpjYVYaZ3mCElDLfwS8_NIkodBoZ2nNq7tiA&_hsmi=96067111&utm_content=96067111&utm_source=hs_email&hsCtaTracking=54468d0b-b5d2-4de7-bbd4-c5c1ac354a56%`

7Ce961609f-c080-4a9a-b6b9-7c935b218577#!/9022/presentation/10255. Informs Annual Meeting.

Zhang, Zhilu, Mert Sabuncu. 2018. Generalized cross entropy loss for training deep neural networks with noisy labels. *Advances in neural information processing systems*. 8778–8788.

Zhu, Tao, Patrick Harrington, Junjun Li, Lei Tang. 2014. Bundle recommendation in ecommerce. *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*. 657–666.