EXPLORATION VS. EXPLOITATION IN THE INFORMATION FILTERING PROBLEM AND ITS APPLICATION IN ARXIV.ORG

A Dissertation

Presented to the Faculty of the Graduate School

of Cornell University

in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

by

Xiaoting Zhao

August 2015

EXPLORATION VS. EXPLOITATION IN THE INFORMATION FILTERING PROBLEM AND ITS APPLICATION IN ARXIV.ORG

Xiaoting Zhao, Ph.D.

Cornell University 2015

We consider information filtering, in which we face a stream of items too voluminous to process by hand (e.g., scientific articles, blog posts, emails), and must rely on a computer system to automatically filter out irrelevant items. Such systems face the exploration vs. exploitation tradeoff, in which it may be beneficial to present an item despite a low probability of relevance, just to learn about future items with similar content. We first present a simple Bayesian sequential decision-making model of this problem, where there is a unit forwarding cost and an user provides immediate feedback on every item forwarded. In the simple model, we can maximize expected total relevance minus forwarding cost using dynamic programming and a decomposition that exploits special problem structure, and show structural results for the optimal policy. We then extend the model in two realistic ways, allowing the user to provide periodic reviews on a bunch of accumulated items, and considering a constrained information filtering system where the user's cost of time is unknown. With that, we develop a policy that ranks items among categories with inspired costs. The proposing methods are especially useful when facing the cold start problem, i.e., when filtering items for new users without a long history of past interactions. We then present an application of the information filtering method to the arxiv.org repository of scientific articles, and show its implementation status in my.arxiv.org, a beta testing version of the website with recommender systems.

BIOGRAPHICAL SKETCH

Xiaoting Zhao was born in China, and moved to U.S.A during high school. She obtained her bachelor degree in Physics and Economics from Smith College in 2009, and subsequently a post-baccalaureate graduate certificate in mathematics in 2010. At Cornell, her research area was sequential decision making problems under uncertainty, with a direct application in building a recommender system (or information filtering systems) for arXiv.org. Her advisor was Peter Frazier.

The thesis is dedicated to my grandmother, Guiying Zheng.

ACKNOWLEDGEMENTS

Five years back on a Sunday, when I was taking a coach shortline bus from New York City to Ithaca for the first time, Juan and Chao picked me up at B lot. They were surprised that I had only one small luggage with me. A day before, I just finished my summer internship in Munich, took a flight to JFK that arrived at midnight and then took a morning bus to Ithaca in order to attend the orientation on Monday. Before coming to Cornell, I told my sister that I would like to learn swimming, scientific computing, golf and dancing at Cornell. Except that I still do not play golf, I have lived my life here to the fullest.

I would like to thank my doctoral advisor, Peter I. Frazier, who has been unconditionally supportive and encouraging to me throughout my years here. He is not only a knowledgeable scholar in the research area that he is dedicated to, but also a kind and mindful guide who has never shown any impatience to the many innocent questions that I have asked again and again. I cannot imagine a better doctoral advisor than Peter. Without his guidance, encouragement and unlimited support, I would not be here today. I feel deeply indebted to him.

I would like to thank Thorsten Joachims and Huseyin Topaloglu for serving on my dissertation committee and providing invaluable feedback on my research. The many discussions with them were enlightening to me and offered me alternative perspectives on my research problems. I would also like to thank Mike Todd for having served on my committee for more than two years and the enlightening mathematical programming courses he taught. I feel grateful to Robert Bland and James Renegar for helping me and encouraging me when I was struggling with my mathematical programming courses. Their kindness and encouragement was indispensable to me and I feel deeply indebted to them. I would like to thank Paul Ginsparg for sharing his experience

in running arXiv.org for the past 20 more years and Vladimir Menkov for implementing our algorithms to my.arXiv.org. I would also like to thank Paul Kantor, David Blei, Laurent Charlin, and Alexander A. Alemi for constructive feedback and interactive collaboration. I feel fortunate to have worked with young and brilliant researchers like J. Massey Cashier, Darlin Alberto, and Saketh Are, from whom I learnt many tricks for writing good code and doing better research. I would also like to thank NSF IIS-1142251 and NSF IIS-1247696 that supported my graduate studies at Cornell.

I feel grateful to have met many good friends that will last for many years here at Cornell. I would like to thank Sin-Shuen Cheung, Shanshan Zhang, Tia Sondjaja, Juan Li, Chao Ding, Yi Shen, Zhengyi Zhou, Kenneth Chong for the many inspiring discussion over academics and the many valuable memories that we share.

Finally, I am deeply thankful to my family, especially my mom and my sister, for listening to me whenever I needed someone to talk to and for being unconditionally supportive and encouraging when I was weak and stressed.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

In this thesis we consider the information filtering problem, in which a human (the *user*) is tasked with processing a stream of time-sensitive items (e.g., emails, text documents, intelligence information, bug reports, scientific articles, music, books, news). Some of these items are relevant (or interesting) and should be examined in detail, while the rest are irrelevant (or uninteresting) and can be ignored. When the stream is too voluminous to be processed by hand, a computer system can be tasked with automatically pre-processing, or *filtering*, these items, forwarding some on to the user, and discarding others. In creating such an information filtering system, we wish to forward as many relevant items as possible, without forwarding too many irrelevant items.

Information filtering systems typically rely on large amounts of historical data to fit statistical models for predicting item relevance (e.g., Agarwal et al. (2011a), Agarwal et al. (2011b), Shivaswamy and Joachims (2012)). When historical relevance data from the user is abundant, we can train a statistical classifier, and forward only those items predicted to be relevant. However, when historical data from the user is limited, e.g., because the user is new, or because we are dealing with new kinds of items, or because items' characteristics or users' interests are rapidly evolving, we face the *cold start problem*, in which we do not have enough training data to build a reliable classifier. This so-called "cold start" problem is prevalent in many information filtering systems (Schein et al., 2002; Rubens et al., 2011).

When historical data is limited, an information filtering system can also learn about item relevance from user-provided feedback (implicitly, through clicks, or

explicitly, through ratings) about previously forwarded items. Moreover, when faced with a sub-stream of items originally predicted to be irrelevant based on limited historical data, an information filtering system might forward a small number of these to the user for feedback, learning with greater certainty their true relevance. Such *exploration* of user preferences is useful because, if this sub-stream is revealed to be relevant, future items from that sub-stream can be forwarded.

However, too much exploration will lead to too many irrelevant forwarded items. Thus, an information filtering system should also put some weight on *exploitation*, i.e., forwarding only those items predicted to be relevant with a high degree of certainty. This tradeoff between exploration and exploitation, which appears in other problem domains including reinforcement learning (Sutton and Barto, 1998; Jaksch et al., 2010), approximate dynamic programming (Powell et al., 2004; Powell, 2007), revenue management (Araman and Caldenty, 2009; Besbes and Zeevi, 2009; den Boer and Zwart, 2013), and inventory control (Lariviere and Porteus, 1999; Ding et al., 2002), is also important for understanding the information filtering problem in regimes with little historical data.

We study this tradeoff between exploration vs. exploitation in a Bayesian setting of the information filtering problem, using a Markov decision process (MDP). We assume that there are k categories of items in the system, in which the sequence of arriving items are independent and identically distributed, with some fixed probabilities. Each category has a hidden and fixed probability θ_x that represents the category's relevance to a user, and irrelevant items are penalized by a user-specified cost per item shown. In the first model, we consider the case where immediate feedback are collected from every forwarded item.

There we show how the MDP defining the Bayes-optimal algorithm for making forwarding decisions may be solved efficiently using a decomposition, and provide structural results of the problem, in which the optimal policy is a threshold policy in terms of the total number of items shown. In addition, we provide an efficient method for computing the optimal information filtering strategy.

Built on the previous model, we then extend it to address two of its realistic limitations. First, we allow items to queue in the system until the next user visit and relax the previous assumption of "immediate feedback" to allow "periodic review". Secondly, we provide a method for ranking over items when a cost-per-item of the user is unknown, and show how an upper bound on the Bayes-optimal procedure and a heuristic index policy can be obtained for the setting when the user will examine only a limited number of forwarded items.

Our focus on a Bayesian setting and Bayes-optimal procedures (rather than procedures that are just optimal up to a constant), is in contrast with the portion of the literature on multi-armed bandits that examines regret in a worst-case setting, and provides algorithms that have optimal dependence on time or other problem parameters, but ignore constants. This focus allows us to apply our method profitably in small-sample regimes, where the best worst-case guarantee would be much worse than the best average-case guarantee, and where constants are just as important as the dependence on time. A downside of our focus on the Bayesian setting is that it requires us to choose a prior distribution to use when measuring performance. However, in many applied settings, including arXiv.org, we argue that a reasonable prior distribution can be estimated from historical data.

1.1 Application to arXiv.org

The work in this thesis is motivated by an information filtering system we are building for the online repository of scientific articles, arXiv.org (arXiv.org, 2014). By December 2014, arXiv.org had accumulated over 1 million full-text articles, was receiving an additional 7000+ new articles each month, and was distributing about 1 million downloads weekly to 400,000 unique users (Ginsparg, 2011; Van Noorden, 2014). In popular categories like astro-ph (Astrophysics) and hep-th (high-energy physics), roughly 80 new articles are submitted each week (arXiv.org, 2014). This massive stream of articles creates a challenge for researchers who wish to keep abreast of those new articles relevant to their research. Thus, we were motivated to build a recommender system for users who visit arXiv.org on a regular basis, either daily or weekly. In Chapter 5, we show the current status of the information filtering system implemented in my.arxiv.org, a beta testing version of the website embedded with recommender systems and other tools that enhance the user's ability to find useful scientific articles.

1.2 Literature Review

Exploration vs. exploitation has been studied extensively in the context of the multi-armed bandit problem in both Bayesian treatments (Gittins and Jones, 1974; Whittle, 1980; Gittins et al., 2011), and non-Bayesian treatments (Auer et al., 1995, 2002). In the multi-armed bandit problem, at each step a player has to choose an alternative from a selection, and each alternative generates a random reward according to some distribution which is unknown aprior to the

player. The player only observes an observation from the alternative he/she chooses to play. The goal of the player is to maximize his/her total rewards through the sequence of decision on choosing alternatives to play. Gittins and Jones (1974) derive the Bayes-optimal policy for the discounted infinite horizon problem. The optimal policy is an index policy, in which the Gittin's index, first called as "dynamic allocation index", reflects the total expected reward collected when a process starts from the current state. Also see (Gittins et al., 2011; Berry and Fristedt, 1985) for a full range of multi-armed bandits problems. Index policies are often not optimal in cases when alternatives are correlated or states of unchosen alternatives are also evolving, but researchers still attempted to develop index policies for these altered problems and found these policies to be useful (Whittle, 1988; Nino-Mora, 2001; Sonin, 2008).

Exploration vs. Exploitation has also been studied in the ranking and selection literature (R&S), see (Bechhofer, 1954; Bechhofer et al., 1968) for the overview. In R&S, there is a collection of alternatives, each with an unknown value. Through a sequence of measurements with noises, one would like to determine the best one among all alternatives. Different from the multi-armed bandit problems, R&S are offline problems, where measurements are collected to help make the final decision, while the multi-armed bandits problems consider overall rewards collected. R&S have also been considered from both non-Bayesian (Paulson, 1964; Rinott, 1978; Hartmann, 1991; Kim and Nelson, 2001) and Bayesian perspectives (Swisher et al., 2003).

This tradeoff between exploration and exploitation also appears in other problem domains, including reinforcement learning (Kaelbling et al., 1998; Sutton and Barto, 1998; Jaksch et al., 2010), approximate dynamic programming

(Powell et al., 2004; Powell, 2007), revenue management (Araman and Caldenty, 2009; Besbes and Zeevi, 2009; den Boer and Zwart, 2013), optimization algorithms (Xie and Frazier, 2013a; Frazier et al., 2008), inventory control (Lariviere and Porteus, 1999; Ding et al., 2002), and design of experiments (Box et al., 1978). In information retrieval problems, exploration vs. exploitation has also been studied in (Zhang et al., 2003; Agarwal et al., 2009; Yue et al., 2009; Hofmann et al., 2013).

There are many works in the active learning and information retrieval communities researching information filtering systems. For an overview, see Rubens et al. (2011), Manning et al. (2008), and Adomavicius and Tuzhilin (2005). The earliest stage of research focuses on cost/credit of delivering relevant/irrelevant documents by various classifiers, including support vector machines (Joachims, 1998), inference networks (Callan, 1996), and maximizing historical data likelihood (Lafferty and Zhai, 2001; Zhang and Callan, 2001). In almost all of this work, the future benefit of reducing uncertainty through exploration is ignored, and a pure exploitation policy is used.

One paper closely related to our approach is Zhang et al. (2003), which studies the exploration vs. exploitation tradeoff in information retrieval using a Bayesian decision-theoretic model. Using Bayesian logistic regression to measure model quality, that previous work quantifies the one-step value of information associated with observing feedback on an item, and computes this approximately using Monte Carlo.

Among other related work, Agarwal et al. (2009), Radlinski et al. (2008), Yue et al. (2009) and Hofmann et al. (2013) study multi-armed bandit methods in recommender systems, and Xu and Akella (2008) considers the problem of choos-

ing results to an initial user query, so as to best improve later search results. While both lines of research study the exploration vs. exploitation tradeoff in information retrieval, neither directly considers the information filtering problem. Shani et al. (2005) introduces the concept of Markov decision processes when modeling recommender systems while Letham et al. (2013) applies a sequential event prediction technique to recommender systems. Both works focus on the fact that recommendation is a sequential decision problem, where the revenue earned in the current period may depend on more than just the most recent action.

Our model can be seen as a Bayesian contextual bandit problem with two arms (forward and discard), and a particular structure (discussed in Section 2.1) for the relationship between context and reward. While finding an optimal policy for general Bayesian contextual bandits is challenging (Langford and Zhang, 2007; May et al., 2012; Agrawal and Goyal, 2012), the special structure that we assume allows us to compute an optimal policy tractably.

1.3 Thesis Organization

Below we summarize the organization for the rest of the thesis.

Chapter 2

In Chapter 2 we propose and analyze a mathematical model of the information filtering problem in a simple setting where users provide *immediate* feedback on forwarded items and there is a unit cost, $c \in [0, 1]$, associated with forwarding each item. Each arriving items is labelled with (exactly) one of k categories while assuming categories are observable. Examples of such categorization schemes

are described in Chapter 4. In this model, we assume that there is a hidden unobservable probability vector, $(\theta_1, ..., \theta_k) \in [0, 1]^k$, reflecting category's relevance to the user.

Under the assumption that θ_x are independent across x, we formulate the problem as a stochastic control problem using Bayesian statistics and stochastic dynamic programming to balance rewards and costs. We then provide an efficient solution by decomposing the original problem into multiple sub-problems that can be solved efficiently. Next, we show structural results: the optimal policy always forwards at least those items forwarded by a pure exploitation policy, and is a threshold policy whose threshold is non-decreasing in the total number of observed items. We also relate these structural results to known properties of two-armed bandit problems. Lastly, we present experimental results using both idealized Monte Carlo simulations and trace-driven simulations with historical data from arXiv.org. This chapter is based on Zhao and Frazier (2014a).

Chapter 3

Chapter 3 builds on the previous chapter and generalizes the information filtering problems in two ways. Instead of providing immediate feedback on forwarded items, we allow items to queue in the system until the next user visit. It is only upon visiting the system that the user provides feedback. This "periodic review" assumption is more realistic in many information filtering systems. Secondly, this chapter considers the case when the unit cost of the user's time is unknown but there is a budget constraint on the total number of items that the user can view. This constrained stochastic dynamic program scales up exponentially, and could not be decomposed as previous problems. We then consider a Lagrangian relaxation of the problem, and provides an upper bound for

the original constrained problem. Inspired by the Lagrangian relaxation of the problem, we then derive a ranking algorithm, called Markov-Decision-Making Information Filtering (MDP-IF), that presents items to the user in this budgeted problem. Lastly, we present complementary experimental results comparing the MDP-IF policy. This chapter is based on Zhao and Frazier (2014b, 2015).

Chapter 4

Motivated by an application to the arXiv (arXiv.org, 2014), in Chapter 4 we consider the problem of finding hard clusters of scientific articles in the presence of user-interaction data and document content. We develop content-augmented stochastic blockmodels (CASB), which is a generative model of user-item interaction as well as item content, such that each document is associated with a single scalar latent variable, indicating its cluster membership. Lastly, we present experimental results of CASB and several state-of-the-art benchmark methods on datasets arising from scientists interacting with scientific articles, and show that the CASB model provides highly accurate clusters with respect to metrics representative of the underlying community structure. The chapter is based on Cashore et al. (2015).

Chapter 5

Chapter 5 evaluates assumptions made in information filtering algorithms we developed in Chpater 2 and Chapter 3. In addition, it describes the current implementation of the clustering scheme and the MDP-IF ranking policy in my.arxiv.org, a beta testing version of personalized recommender system for arXiv.org.

CHAPTER 2

INFORMATION FILTERING: IMMEDIATE REVIEWS WITH FIXED UNIT COST

In this chapter, we consider a simple information filtering problem, in which a user provides immediate reviews on items shown and there is a fixed unit cost for forwarding every item in order to capture the opportunity cost for the user's time on viewing the item. We then propose a mathematical model for this simple information filtering problem, and show structural properties of the problem. In Chapter 3 we generalize the model to consider period reviews and unknown forwarding costs, reflecting characteristics in many real information filtering systems. A variant of such problems presented in Chapter 3 becomes intractable, there we provide a Lagrangian relaxation of the problem.

Through the simple assumptions of immediate reviews and fixed costs, analysis in this chapter provides clear insight into the exploration vs. exploitation tradeoff in information filtering, and more generally into the structure of the optimal information filtering strategy. In comparison with a myopic "pure exploitation" strategy, we show that the optimal filtering strategy forwards every item that the myopic strategy forwards, and potentially forwards additional items that the myopic strategy would not forward. Moreover, this willingness to forward additional items, i.e., to explore, is largest when the number of items on which we have relevance feedback is small, and decreases as this number of items with feedback grows larger. In the limit as the number of items with feedback grows to infinity, this willingness to forward additional items vanishes, and the decisions of the optimal strategy match those of the myopic strategy.

We additionally provide an efficient method for computing the optimal information filtering strategy. While the optimal strategy is the solution to a partially observable Markov Decision Processes, and thus can be computed, at least conceptually, using dynamic programming (Frazier, 2011), the curse of dimensionality prevents directly computing this solution in practice. To circumvent this issue, we show that the problem can be decomposed into a collection of much smaller dynamic programs, which can be solved efficiently. Indeed, each smaller dynamic program is a two-armed bandit problem, with one unknown Bernoulli arm and one known arm, and can be solved directly, or using methods from the extensive literature on two-armed bandits (Bellman, 1956; Gittins and Jones, 1974; Gittins, 1979; Whittle, 1980; Berry and Fristedt, 1985; Katehakis and Veinott, 1987; Gittins et al., 2011).

Below we summarize the organization for the rest of this chapter. In Section 2.1, we formulate the information filtering problem in a setting, a known unit forwarding cost and immediate review on items shown, as a stochastic control problem. In Section 2.2, we provide an efficient solution by decomposing the original problem into multiple sub-problems that can be solved efficiently. We then show structural results: the optimal policy always forwards at least those items forwarded by a pure exploitation policy, and is a threshold policy whose threshold is non-decreasing in the total number of observed items. We also relate these structural results to known properties of two-armed bandit problems. Lastly, we present experimental results in Section 2.4, using both idealized Monte Carlo simulations and trace-driven simulations with historical data from arXiv.org.

2.1 Mathematical Model

Each item arriving from our information stream is labelled with (exactly) one of k categories, and we let $X_n \in \{1, \ldots, k\}$ be the category of the nth item in our steam. This category is observable by our information filtering algorithm. In the application to arXiv.org that we present in Section 2.4, we describe a setting where the category is provided by a human (the author) who submits the item to the stream. This category could also be obtained automatically by a machine learning algorithm from item contents, see such an example in Chapter 4 and Chapter 5. We model the sequence of random variables $(X_n : n = 1, 2, \ldots)$ as being independent and identically distributed, and we let $p_x = P(X_n = x) > 0$ for $x = 1, \ldots, k$, with $\sum_{x=1}^{k} p_x = 1$.

In our model and analysis, we focus on a single user, and then in implementation we apply the resulting algorithm separately for each user. Fixing this user, we model each category as having associated with it some latent unobservable value $\theta_x \in [0, 1]$, which is the probability that the user under consideration would find an item from this category to be relevant, if it were forwarded to her/him. We let $\theta = [\theta_1, \ldots, \theta_k]$. Our model assumes that θ_x remains static over time. This focus on a single user is in contrast with much of the work on collaborative filtering, and is motivated by the design requirements of the information filtering system we are building for arXiv.org, where concerns about fairness to authors make it especially important to avoid cascades and the Matthew effect (Easley and Kleinberg, 2010), in which popular items become more popular, irrespective of quality.

We model each θ_x as having been drawn independently for each x from a

Bayesian prior probability distribution, which is beta-distributed. We let α_{0x}, β_{0x} be the two parameters of this distribution, so that $\theta_x \sim \text{Beta}(\alpha_{0x}, \beta_{0x})$. In Section 2.4 we provide a method for estimating the parameters of this prior probability distribution from historical data. In Chapter 5 we exam details of independent assumptions and distributions made in the model.

For each item in the stream, our information filtering algorithm then decides whether to forward this item to the user, or to discard it. We let $U_n \in \{0, 1\}$ represent the decision made for the nth item, where U_n is 1 if the algorithm forwards this item, and 0 if it discards it. Each forwarded item is then seen by the user, who provides feedback on its relevance in the form of a Bernoulli random variable Y_n. In the application presented in Section 2.4, the feedback Y_n is provided implicitly, by whether or not the user clicked to view the forwarded item. In other applications, this feedback might be provided via an explicit rating inputted by the user. The sequence of random variables $(Y_n : n = 1, 2, \ldots)$ are conditionally independent given θ_x, with $P(Y_n = 1 | X_{1:n}, U_{1:n}, \theta) = \theta_{X_n}$. No feedback is provided on discarded items, and so Y_n is observed if and only if $U_n = 1$.

The decision of whether or not to forward the item may be made based only on the information available from previous forwarding decisions. That is, we require U_n to depend only on the current item's category X_n, and the history $H_{n-1} = (X_\ell, U_\ell, U_\ell Y_\ell : \ell \leq n - 1)$. In this definition of the history, we emphasize that Y_ℓ is only observable if $U_\ell = 1$.

In our model, we pay an explicit cost c for each item forwarded to the user, which models the cost of the user's time, and a reward of 1 for each relevant item forwarded. Thus, the total reward resulting from forwarding an item is

$U_n(Y_n - c)$.

At each time step n, there is a probability γ that the user will remain engaged through the next time step $n+1$, and a probability $1-\gamma$ that the user will abandon the system and never return, so N follows a geometric distribution with parameter $1-\gamma$. We have modeled the items as arriving in discrete time. Our approach can also be easily adapted to a continuous-time setting, where documents arrive according to a Poisson process and are reviewed instantly while the user has an exponential lifetime in the system. In this case, the discrete-time formulation is obtained by counting item arrivals. We discuss this further in Section 2.4 in the context of arXiv.org along with estimation of γ and validation of our modelling assumptions.

In Section 2.4, we discuss estimation of γ in our application to arXiv.org, and validation of our assumption. Our goal is to design an algorithm for making forwarding decisions that maximizes the expectation of the cumulative reward, $\sum_{n=1}^{N} U_n(Y_n - c)$. Our model is summarized by Figure 2.1.

To formalize this as a stochastic control problem, we define a policy π as a sequence of functions, $\pi = (\pi_1, \pi_2, \ldots)$, where each $\pi_n : (\{1, \ldots, k\} \times \{0,1\} \times \{0,1\})^{n-1} \times \{1, \ldots, k\} \mapsto \{0,1\}$ maps histories onto actions. We let Π be the space of all such policies. For each $\pi \in \Pi$, we define P^π to be the measure under which $U_{n+1} = \pi_{n+1}(H_n, X_{n+1})$ almost surely for each n, and we let E^π represent the expectation taken with respect to this measure. Our goal is then to solve

$$\sup_{\pi \in \Pi} E^\pi \left[\sum_{n=1}^{N} U_n(Y_n - c) \right]. \tag{2.1}$$

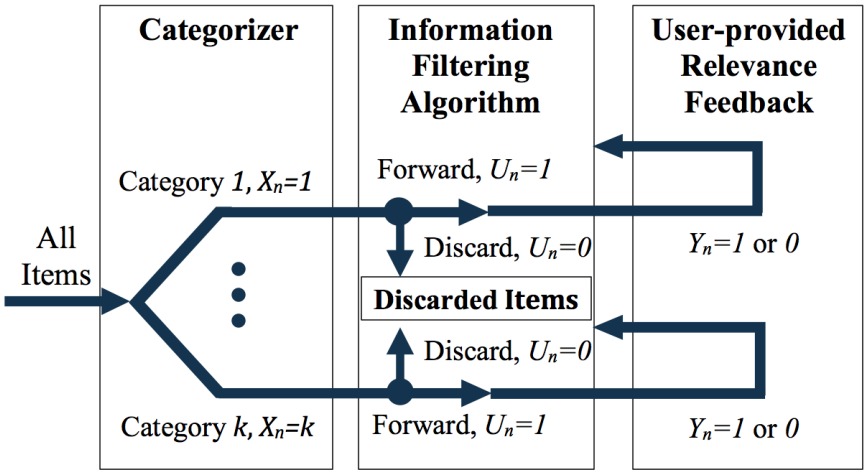


Figure 2.1: Schematic of the information filtering problem. Arriving items are categorized into one of k categories, and then are forwarded or discarded by an information filtering algorithm. This algorithm uses feedback on forwarded items to improve later forwarding decisions.

2.2 Solution Method and Structural Results

While (2.1) is a stochastic control problem, and could be solved via dynamic programming, the size of the state space of this dynamic program grows exponentially in k. This is the so-called "curse of dimensionality" (Powell, 2007). To circumvent this issue, we decompose the problem into a collection of single-category problems, each of which is a two-armed bandit problem and can be solved efficiently using dynamic programming. This is accomplished in Section 2.2.1, which performs the decomposition, and Section 2.2.2, which converts each single-category problem from its original form (undiscounted random horizon) to an easier-to-solve form (discounted infinite-horizon). We then provide the dynamic programming equations for these single category problems in Section 2.2.3, which we then use to show structural results in Section 2.2.4. Sections 2.2.2 and 2.2.3 follow standard arguments, but are included to support results in Section 2.2.4.

2.2.1 Decomposition into single-category subproblems

To decompose the problem (2.1) into a number of easily solved single-category problems, we first introduce some additional notation. Let $n_{\ell x} = \inf\{n : \sum_{i=1}^{n} \mathbb{1}_{[X_i=x]} = \ell\}$ be the index, in the overall stream of items, of the ℓth item from category x. We then define $U_{\ell x} = U_{n_{\ell x}}$ and $Y_{\ell x} = Y_{n_{\ell x}}$ to be the forwarding decision and relevance, respectively, of the ℓth item from category x. Finally, we let $N_x = \sup\{\ell : n_{\ell x} \leq N\}$ be the number of items from category x that arrive before the user leaves the system at step N. We let $H_{nx} = (U_{\ell x}, U_{\ell x} Y_{\ell x} : \ell \leq n)$ be the history of forwarding decisions and relevance feedback for items from category x.

For each category x, we then define a class of policies for making forwarding decisions for items *from that category only*, based only on the portion of the history arising from items in that category. Formally, we define a *single-category policy* for a given category x to be a sequence of functions, $\pi^{(x)} = (\pi_1^{(x)}, \pi_2^{(x)}, \ldots)$, where each function $\pi_{n+1}^{(x)} : \{0,1\}^{2n} \mapsto \{0,1\}$ maps H_{nx} onto $U_{n+1,x}$. We let $\Pi^{(x)}$ be the space of all such policies. For each $\pi^{(x)} \in \Pi^{(x)}$, we let $P^{\pi^{(x)}}$ be the probability measure under which $U_{n+1,x} = \pi_{n+1}^{(x)}(H_{nx})$ for each n, and we let $E^{\pi^{(x)}}$ be the expectation with respect to this measure.

We will write the value of the overall problem (2.1) in terms of the sum of the values of the solutions to forwarding problems for individual categories,

$$\sup_{\pi^{(x)} \in \Pi^{(x)}} E^{\pi^{(x)}} \left[\sum_{n=1}^{N_x} U_{nx}(Y_{nx} - c) \right]. \tag{2.2}$$

This theorem below gives us a way to construct the optimal policy for the multi-category problem from the solutions to single-category problems: to make a forwarding decision for a new item in the multi-category problem, we

identify the category X_{n+1} of that item, and then perform the forwarding decision that would have been made by the optimal policy for the single-category problem for that category. Because the objective is additive across categories, and because the prior on categories' relevance are independent of each other, relevance feedback from one category gives no information about the relevance of other categories. Thus, when considering whether or not to forward an item for a particular category, it is sufficient to consider the history of observations from that category alone.

Theorem 2.2.1.

$$\sup_{\pi\in\Pi} E^{\pi}\left[\sum_{n=1}^{N} U_n(Y_n - c)\right] = \sum_{x=1}^{k} \sup_{\pi^{(x)}\in\Pi^{(x)}} E^{\pi^{(x)}}\left[\sum_{n=1}^{N_x} U_{nx}(Y_{nx} - c)\right].$$

Moreover, if $\pi^{(x),*}$ attains the supremum in (2.2) *for each x, and if π^* is the policy constructed by setting $\pi^*_{n+1}(H_n, X_{n+1}) = \pi^{(X_{n+1}),*}_{n+1}(H_{nx})$ for each n, then π^* attains the supremum in* (2.1)*.*

Proof. Let V be equal to the value of (2.1), and for each x, let V_x be equal to the value of (2.2). (We have dropped the α and β from the notation V and V_x in this proof.) By construction, on each sample path,

$$\sum_{n=1}^{N} U_n(Y_n - c) = \sum_{x=1}^{k}\sum_{\ell=1}^{N_x} U_{\ell x}(Y_{\ell x} - c).$$

Thus, for each $\pi \in \Pi$,

$$E^{\pi}\left[\sum_{n=1}^{N} U_n(Y_n - c)\right] = E^{\pi}\left[\sum_{x=1}^{k}\sum_{\ell=1}^{N_x} U_{\ell x}(Y_{\ell x} - c)\right] = \sum_{x=1}^{k} E^{\pi}\left[\sum_{\ell=1}^{N_x} U_{\ell x}(Y_{\ell x} - c)\right], \qquad (2.3)$$

and we have

$$V = \sup_{\pi\in\Pi} E^{\pi}\left[\sum_{n=1}^{N} U_n(Y_n - c)\right] = \sup_{\pi\in\Pi} \sum_{x=1}^{k} E^{\pi}\left[\sum_{\ell=1}^{N_x} U_{\ell x}(Y_{\ell x} - c)\right] \le \sum_{x=1}^{k} \sup_{\pi\in\Pi} E^{\pi}\left[\sum_{\ell=1}^{N_x} U_{\ell x}(Y_{\ell x} - c)\right],$$

where the last inequality follows from the fact that the right-hand side potentially allows a different π to attain the supremum for each x.

Now fix an x and consider the term $\sup_{\pi\in\Pi} E^{\pi}\left[\sum_{\ell=1}^{N_x} U_{\ell x}(Y_{\ell x}-c)\right]$. This is equal to $V_x = \sup_{\pi^{(x)}\in\Pi^{(x)}} E^{\pi^{(x)}}\left[\sum_{\ell=1}^{N_x} U_{\ell x}(Y_{\ell x}-c)\right]$ because the conditional distribution of $(Y_{ix}, N_x : i \geq \ell)$ given the history available to any policy $\pi \in \Pi$ when making the decision $U_{\ell x}$ (this history is $(X_i, U_i, U_i Y_i : i < n_{\ell x})$ and $X_{n_{\ell x}} = x$) depends only upon the history available to a policy $\pi^{(x)} \in \Pi^{(x)}$, which is $(U_{ix}, U_{ix}Y_{ix} : i < \ell)$.

Thus, we have $\sup_{\pi\in\Pi} E^{\pi}\left[\sum_{\ell=1}^{N_x} U_{\ell x}(Y_{\ell x}-c)\right] = V_x$, and

$$V \leq \sum_{x=1}^{k} V_x. \tag{2.4}$$

We now show the opposite inequality, and the additional claim about constructing an optimal policy for (2.1). Let $\pi^{(x),*}$ and π^* be as described in the statement of the theorem. Then, for each x,

$$E^{\pi^*}\left[\sum_{\ell=1}^{N_x} U_{\ell x}(Y_{\ell x}-c)\right] = E^{\pi^{(x),*}}\left[\sum_{\ell=1}^{N_x} U_{\ell x}(Y_{\ell x}-c)\right] = V_x.$$

Summing over x and using (2.3), we have that the value of π^* in the original problem is

$$E^{\pi^*}\left[\sum_{n=1}^{N} U_n(Y_n-c)\right] = \sum_{x=1}^{k} E^{\pi^*}\left[\sum_{\ell=1}^{N_x} U_{\ell x}(Y_{\ell x}-c)\right] = \sum_{x=1}^{k} V_x. \tag{2.5}$$

Since $\pi^* \in \Pi$, and V is defined by taking the supremum over all policies π, we also have

$$E^{\pi^*}\left[\sum_{n=1}^{N} U_n(Y_n-c)\right] \leq V. \tag{2.6}$$

Combining (2.4), (2.5) and (2.6) we have,

$$\sum_{x=1}^{k} V_x = E^{\pi^*}\left[\sum_{n=1}^{N} U_n(Y_n-c)\right] \leq V \leq \sum_{x=1}^{k} V_x,$$

which implies both that $V = \sum_{x=1}^{k} V_x$, and that π^* attains the supremum defining V, as claimed. □

2.2.2 Conversion of finite-horizon single-category subproblems into infinite-horizon two-armed bandits

Below, it will be useful to transform this single-category problem (2.2) from a finite-horizon undiscounted problem, with a random horizon, into an infinite-horizon discounted problem. This transformation will make clear that the single subproblems are two-armed bandit problems. In performing this transformation, we use a standard geometric killing approach.

We first note that the number of items available for forwarding in a particular category has a geometric distribution.

Remark 2.2.1. *$N_x \sim$ Geometric $(1 - \gamma_x)$, where $\gamma_x = \frac{p_x\gamma}{p_x\gamma+1-\gamma}$. Here, by* Geometric$(1-q)$*, we mean the probability distribution supported on $\{0, 1, 2, \ldots\}$ that assigns probability mass $(1 - q)q^n$ to integer n.*

Proof. Fix any $n \in \{0, 1, 2, \ldots\}$,

$$P(N_x = n) = \sum_{m \geq n} P(N = m)\binom{m}{n} p_z^n (1 - p_z)^{m-n} = \sum_{m \geq n} \gamma^m (1 - \gamma)\binom{m}{n} p_z^n (1 - p_z)^{m-n}$$
$$= (1 - \gamma)(p_z\gamma)^n \sum_{m \geq n} \binom{m}{n} [(1 - p_z)\gamma]^{m-1} = \frac{(1 - \gamma)(p_z\gamma)^n}{[1 - (1 - p_z)\gamma]^{n+1}} = \gamma_x^n (1 - \gamma_x).$$

The third equality use the formula that $\frac{1}{(1-x)^s} = \sum_{k=0}^{\infty} \binom{s+k-1}{k} x^k = \sum_{k=0}^{\infty} \binom{s+k-1}{s-1} x^k$. With that, we show the claim. □

Using Remark 2.2.1, the following lemma writes the performance of any policy $\pi^{(x)}$ as an infinite-horizon discounted sum.

Lemma 2.2.1. *For each policy $\pi^{(x)} \in \Pi^{(x)}$, we have*

$$E^{\pi^{(x)}}\left[\sum_{n=1}^{N_x} U_{nx}(Y_{nx} - c)\right] = \gamma_x E^{\pi^{(x)}}\left[\sum_{n=1}^{\infty} \gamma_x^{n-1} U_{nx}(Y_{nx} - c)\right].$$

Proof. Using Fubini's theorem and $E^{\pi(x)}\left[\sum_{n=1}^{\infty} |\mathbb{1}_{[n\leq N_x]} U_{nx}(Y_{nx} - c)|\right] < \infty$, we have

$$E^{\pi^{(x)}}\left[\sum_{n=1}^{N_x} U_{nx}(Y_{nx} - c)\right] = E^{\pi^{(x)}}\left[\sum_{n=1}^{\infty} \mathbb{1}_{[n\leq N_x]} U_{nx}(Y_{nx} - c)\right] = \sum_{n=1}^{\infty} E^{\pi^{(x)}}\left[\mathbb{1}_{[n\leq N_x]} U_{nx}(Y_{nx} - c)\right]. \tag{2.7}$$

Now consider one of these terms, for any fixed n, we have that

$$\begin{aligned} E^{\pi^{(x)}}\left[\mathbb{1}_{[n\leq N_x]} U_{nx}(Y_{nx} - c)\right] &= E^{\pi^{(x)}}\left[E^{\pi^{(x)}}\left[\mathbb{1}_{[n\leq N_x]} U_{nx}(Y_{nx} - c) \mid U_{nx}, Y_{nx}\right]\right] \\ &= E^{\pi^{(x)}}\left[P^{\pi^{(x)}}\left(n \leq N_x \mid U_{nx}, Y_{nx}\right) U_{nx}(Y_{nx} - c)\right] \\ &= E^{\pi^{(x)}}\left[\gamma_x^n U_{nx}(Y_{nx} - c)\right]. \end{aligned}$$

Plugging this expression into (2.7) and applying Fubini's theorem again shows

$$E^{\pi^{(x)}}\left[\sum_{n=1}^{N_x} U_{nx}(Y_{nx} - c)\right] = \sum_{n=1}^{\infty} E^{\pi^{(x)}}\left[\gamma_x^n U_{nx}(Y_{nx} - c)\right] = \gamma_x E^{\pi^{(x)}}\left[\sum_{n=1}^{\infty} \gamma_x^{n-1} U_{nx}(Y_{nx} - c)\right],$$

which is the claimed expression. □

This lemma shows that we can find an optimal policy for the single category problem (2.2) by solving the stochastic control problem,

$$\sup_{\pi^{(x)} \in \Pi^{(x)}} E^{\pi^{(x)}}\left[\sum_{n=1}^{\infty} \gamma_x^{n-1} U_{nx}(Y_{nx} - c)\right]. \tag{2.8}$$

Any policy that is optimal for this problem is optimal for (2.2). We have dropped the strictly positive constant γ_x when stating this stochastic control problem, since this constant does not affect the ordering of the policies.

Equation (2.8) is a two-armed bandit, with one unknown Bernoulli arm (giving rewards of $1-c$ or $-c$) and one known arm (giving rewards of 0). By adding an additional reward c in all periods, this problem can be seen to be equivalent to a more conventional two-armed bandit, where the unknown Bernoulli arm gives rewards of 0 or 1, and the known arm gives rewards of c. Both two-armed and Bernoulli bandits have been studied extensively (Bellman, 1956; Berry and Fristedt, 1985), and a variety of methods have been proposed for computing optimal policies Katehakis and Veinott (1987); Gittins et al. (2011). One may use one of these techniques, or simply solve (2.8) directly using dynamic programming, as described in Section 2.3.

After Section 2.2.3 summarizes the dynamic programming equations for (2.8), Section 2.2.4 will prove structural results that apply beyond the forwarding problem more generally to Bernoulli bandits. These results do not, to the best of our knowledge, appear previously in the literature. We will also relate these new results to related results on Bernoulli bandits from Berry and Fristedt (1985), and for bandit rewards in location-scale families from Gittins et al. (2011).

2.2.3 Dynamic programming equations for the single-category subproblem

We now provide the dynamic programming equations for the infinite-horizon discounted version of the single-category problem (2.8), which we use in Section 2.2.4 to derive novel structural results. As noted above, the same optimal policy is optimal for the finite-horizon undiscounted problem (2.2), and is used

with Theorem 2.2.1 to provide an optimal policy for the original problem (2.1). The dynamic program considered in this section has an infinite state space, preventing an exact solution, and so we later provide a computable approximation and error bounds in Section 2.3 that can be used to solve it with any desired accuracy.

First, the conditional distribution of θ_x given history H_{nx} is

$$\theta_x \mid H_{nx} \ \sim \text{Beta}\,(\alpha_{nx}, \beta_{nx})\,,$$

where $\alpha_{nx} = \alpha_{0x} + \sum_{i=1}^{n} U_{ix}\, Y_{ix}$ is the sum of α_{0x} and the number of relevant items forwarded, and $\beta_{nx} = \beta_{0x} + \sum_{i=1}^{n}(1 - Y_{ix})U_{ix}$ is the sum of β_{0x} and the number of irrelevant items forwarded. This follows from standard results from Bayesian statistics on conjugate priors for Bernoulli observations. For details, see, e.g., DeGroot (2004). It will also be convenient to introduce the notation $\mu(\alpha, \beta) = \alpha/(\alpha + \beta)$ to refer to the mean of a Beta(α, β) distribution.

Moreover, since $Y_{\ell x}$ are conditionally i.i.d. given θ_x, the conditional distribution of the sequence $(Y_{\ell x} : \ell > n)$ given the history H_{nx} is completely determined by α_{nx}, β_{nx}. Thus, the solution to (2.8) will be given by a dynamic program whose state space includes all possible values of $(\alpha_{nx}, \beta_{nx})$.

We briefly review this use of dynamic programming, providing definitions and notations that will be used below when presenting and proving structural results. For any scalar real numbers $\alpha, \beta > 0$, we define the value function

$$V_x(\alpha, \beta) = \sup_{\pi^{(x)} \in \Pi^{(x)}} E^{\pi^{(x)}} \left[\sum_{n=1}^{\infty} \gamma_x^{n-1} U_{nx}(Y_{nx} - c) \mid \theta_x \sim \text{Beta}(\alpha, \beta) \right]. \tag{2.9}$$

The value function satisfies Bellman's equation

$$V_x(\alpha, \beta) = \max\{Q(\alpha, \beta, 0),\ Q(\alpha, \beta, 1)\}, \tag{2.10}$$

where we define the Q-factor $Q : (0,\infty)^2 \times \{0,1\} \mapsto \mathbb{R}$ by

$$Q_x(\alpha,\beta,0) = \gamma_x V_x(\alpha,\beta), \tag{2.11}$$

$$Q_x(\alpha,\beta,1) = E[Y_1 - c + \gamma_x V_x(\alpha_{1x},\beta_{1x}) | U_{1x} = 1, \alpha_{0x} = \alpha, \beta_{0x} = \beta] \tag{2.12}$$

$$= \mu - c + \gamma_x[\mu V_x(\alpha+1,\beta) + (1-\mu)V_x(\alpha,\beta+1)], \tag{2.13}$$

where $\mu = \mu(\alpha,\beta) = \alpha/(\alpha+\beta)$. Here, $Q_x(\alpha,\beta,0)$ and $Q(\alpha,\beta,1)$ are the value of discarding and forwarding the item, and proceeding optimally thereafter, when the posterior on θ_x is Beta(α,β).

An optimal policy is then any whose decisions attain the maximum in this recursion, breaking ties arbitrarily. That is, an optimal policy is any for which

$$U_{n+1,x} \in \text{argmax}_{u=0,1} Q_x(\alpha_{nx},\beta_{nx},u).$$

Thus, if we are able to compute the value function, from it we may compute the Q-factors, and then compute an optimal policy. While the set of possible values for (α_{nx},β_{nx}), and thus the size of the state space that must be considered, is countably infinite, preventing exact computation of the value function, we describe below in Section 2.3 a truncation method for computing upper and lower bounds on V_x, from which an approximation with explicit error bounds can be computed. The error in this approximation vanishes as the level of truncation grows.

2.2.4 Structural Results

In this section, we prove structural results that provide insight into the behavior of the optimal policy (that the optimal policy is a threshold policy, in Theorem 2.2.2, and behavior of this threshold, in Theorem 2.2.3), yield computational

benefits by reducing the amount of storage needed to implement the optimal policy (Theorem 2.2.2), and that form the foundation for computing approximations with explicit error bounds in Section 2.3 (Propositions 2.2.1 and 2.2.2). Similar properties are known to hold for location-scale families, and our work shows that they also hold in the Beta-Bernoulli setting, which is more relevant in the "click-based" forwarding problem studied here.

When solving the single-category sub-problem, we first notice that the Q-factor for the *discard* decision, $Q_x(\alpha, \beta, 0)$, has the following structure, stated in Remark 2.2.2.

Remark 2.2.2. *It is optimal to discard at* $(\alpha, \beta) \in (0, \infty)^2$ *if and only if* $V_x(\alpha, \beta) = 0$. *Equivalently,* $Q_x(\alpha, \beta, 0) = 0$.

Proof. When it is optimal to discard at (α, β), we have $V_x(\alpha, \beta) = Q_x(\alpha, \beta, 0) = \gamma_x \cdot V_x(\alpha, \beta)$ for some $0 < \gamma_x < 1$. This implies that $V_x(\alpha, \beta) = Q_x(\alpha, \beta, 0) = 0$. We now show the other direction. If $V_x(\alpha, \beta) = 0$, then $Q_x(\alpha, \beta, 0) = \gamma_x V_x(\alpha, \beta) = 0$ and furthermore $Q_x(\alpha, \beta, 1) \leq V_x(\alpha, \beta) = Q_x(\alpha, \beta, 0) = 0$. □

Remark 2.2.2 implies that we may rewrite Bellman's equation (2.10) as $V_x(\alpha, \beta) = \max\{0,\ Q(\alpha, \beta, 1)\}$. Next, we provide a lower and upper bound on the value function, and use them below to prove convergence of the value function in Proposition 2.2.3. We also use them when describing a truncation method with explicit error bounds that computes the solution to the single-category sub-problem in Section 2.3.

Since the value of any policy provides a lower bound on the value function, we consider a policy that ignores feedback, forwards all items if $\mu(\alpha_{0x}, \beta_0) \geq c$, and discards all items otherwise. The value of this policy is easy to com-

pute, and so provides a convenient lower bound. This is the basis for Proposition 2.2.1. Similarly, to obtain an upper bound on $V_x(\alpha,\beta)$, we consider an environment in which the true value of θ_x is revealed. The value of an optimal policy in this environment provides an upper bound on the value function. This is the basis for Proposition 2.2.2.

Proposition 2.2.1. *Define* $V_x^L(\alpha,\beta) = \frac{1}{1-\gamma_x}\max\{0,\ \mu(\alpha,\beta) - c\}$ *for any* $\alpha \in (0,\infty)$ *and* $\beta \in (0,\infty)$*. Then, we have the lower bound,* $V_x(\alpha,\beta) \geq V_x^L(\alpha,\beta)$*.*

Proof. To show the lower bound, we consider a policy that ignores learning and makes decisions based on the conditional expected reward, $E\left[Y|\theta \sim \text{Beta}(\alpha,\beta)\right] = \mu(\alpha,\beta) = \frac{\alpha}{\alpha+\beta}$. Let us define the policy, $\pi_L^{(x)}$, such that for all $n \geq 1$:

$$U_{nx} = \begin{cases} 1 & \text{if } \mu(\alpha,\beta) \geq c, \\ 0 & \text{otherwise.} \end{cases}$$

Then, this policy has value:

$$\begin{aligned} V_x^L(\alpha,\beta) &= E^{\pi_L^{(x)}}\left[\sum_{n=1}^{\infty}\gamma_x^{n-1}U_{nx}(Y_{nx}-c) \mid \theta_x \sim \text{Beta}(\alpha,\beta)\right] \\ &= \sum_{n=1}^{\infty}\gamma_x^{n-1}\ \max\left\{0,\mu(\alpha,\beta)-c\right\} = \frac{1}{1-\gamma_x}\max\{0,\ \mu(\alpha,\beta)-c\} \\ &\leq \sup_{\pi^{(x)}\in\Pi^{(x)}} E^{\pi^{(x)}}\left[\sum_{n=1}^{\infty}\gamma_x^{n-1}U_{nx}(Y_{nx}-c) \mid \theta_x \sim \text{Beta}(\alpha,\beta)\right] = V_x(\alpha,\beta). \ \square \end{aligned}$$

Proposition 2.2.2. *Define* $V_x^U(\alpha,\beta) = \frac{1}{1-\gamma_x}E[\max\{0,\theta_x - c\}]$ *for any* $\alpha \in (0,\infty)$ *and* $\beta \in (0,\infty)$*. Then, we have the upper bound* $V_x^U(\alpha,\beta) \geq V_x(\alpha,\beta)$*.*

Proof. We have

$$
\begin{aligned}
V_x(\alpha,\beta) &= \sup_{\pi^{(x)}\in\Pi^{(x)}} E^{\pi^{(x)}}\left[\sum_{n=1}^{\infty}\gamma_x^{n-1}(Y_{nx}-c)U_{nx} \mid \alpha_{0x}=\alpha,\beta_{0x}=\beta\right] \\
&= \sup_{\pi^{(x)}\in\Pi^{(x)}} \sum_{n=1}^{\infty}\gamma_x^{n-1}E^{\pi^{(x)}}\Big[(Y_{nx}-c)U_{nx} \mid \alpha_{0x}=\alpha,\beta_{0x}=\beta\Big] \\
&= \sup_{\pi^{(x)}\in\Pi^{(x)}} \sum_{n=1}^{\infty}\gamma_x^{n-1}E^{\pi^{(x)}}\Big[E^{\pi^{(x)}}[(Y_{nx}-c)U_{nx} \mid H_{n-1,x},\theta_x,\alpha_{0x}=\alpha,\beta_{0x}=\beta] \mid \alpha_{0x}=\alpha,\beta_{0x}=\beta\Big] \\
&= \sup_{\pi\in\Pi^{(x)}} \sum_{n=1}^{\infty}\gamma_x^{n-1}E^{\pi^{(x)}}\Big[(\theta_x-c)U_{nx} \mid \alpha_{0x}=\alpha,\beta_{0x}=\beta\Big] \\
&\le \sum_{n=1}^{\infty}\gamma_x^{n-1}E\Big[\max\{0,\theta_x-c\} \mid \alpha_{0x}=\alpha,\beta_{0x}=\beta\Big] \\
&= \frac{1}{1-\gamma_x}E\Big[\max\{0,\theta_x-c\} \mid \alpha_{0x}=\alpha,\beta_{0x}=\beta\Big] = V_x^U(\alpha,\beta).
\end{aligned}
$$

The first equality is the definition of the value function in equation (2.9), while the second equality is due to Fubini's Theorem since $E^{\pi(x)}\left[\sum_{n=1}^{\infty}|\gamma_x^{n-1}(Y_{nx}-c)U_{nx}|\middle|\alpha_{0x}=\alpha\right]$. Then applying the tower property of conditional expectation, we derive the third equation. Because the expected value of Y_{nx} conditioned on θ_x is θ_x and U_{nx} only depends on $H_{n-1,x}$, the inner conditional expectation reduces to $(\theta_x-c)U_{nx}$, which is smaller than $\max\{0,\theta_x-c\}$. So the inequality holds in the fifth line and we derive an upper bound for the value function. □

Combining Proposition 2.2.1 and Proposition 2.2.2, we show convergence of the value function as the prior converges to one in which we are certain about θ_x, as stated below in Proposition 2.2.3. This proposition is used in the proofs of Lemma 2.2.2 and 2.2.3, and of Theorem 2.2.3.

Proposition 2.2.3. *Let $(\alpha_{nx},\beta_{nx})_{n=1}^{\infty}$ be a sequence such that $\alpha_{nx},\beta_{nx} \ge 0$ with $\lim_{n\to\infty}\alpha_{nx}+\beta_{nx}=\infty$ and $\lim_{n\to\infty}\frac{\alpha_{nx}}{\alpha_{nx}+\beta_{nx}}=\mu_x$. Then, the value function has the limit,*

$$\lim_{n\to\infty} V_x(\alpha_{nx},\beta_{nx}) = \frac{1}{1-\gamma_x}\max\{0,\mu_x-c\}.$$

Proof. First, we show a lower bound on $\liminf_{n\to\infty} V_x(\alpha_{nx},\beta_{nx})$:

$$\begin{aligned}
\liminf_{n\to\infty} V_x(\alpha_{nx},\beta_{nx}) &\geq \liminf_{n\to\infty}\left[\frac{1}{1-\gamma_x}\max\left\{0,\frac{\alpha_{nx}}{\alpha_{nx}+\beta_{nx}}-c\right\}\right] \\
&= \lim_{n\to\infty}\left[\frac{1}{1-\gamma_x}\max\left\{0,\frac{\alpha_{nx}}{\alpha_{nx}+\beta_{nx}}-c\right\}\right] \\
&= \frac{1}{1-\gamma_x}\max\left\{0,\lim_{n\to\infty}\frac{\alpha_{nx}}{\alpha_{nx}+\beta_{nx}}-c\right\} \\
&= \frac{1}{1-\gamma_x}\max\{0,\mu_x-c\}
\end{aligned}$$

The first inequality uses Proposition 2.2.1, and the rest follows because the limit of $\frac{1}{1-\gamma_x}\max\left\{0,\frac{\alpha_{nx}}{\alpha_{nx}+\beta_{nx}}-c\right\}$ exists. Similarly, using Proposition 2.2.2, we show an upper bound on $\limsup_{n\to\infty} V_x(\alpha_{nx},\beta_{nx})$:

$$\begin{aligned}
\limsup_{n\to\infty} V_x(\alpha_{nx},\beta_{nx}) &\leq \limsup_{n\to\infty}\left\{\frac{1}{1-\gamma_x}E[\max\{0,\theta_x-c\}|\ \theta_x\sim \text{Beta}(\alpha_{nx},\ \beta_{nx})]\right\} \\
&\leq \frac{1}{1-\gamma_x}\max\{0,\mu_x-c\}
\end{aligned}$$

The second inequality holds because of the Portmanteau Theorem (Resnick (2005), page 264) since the function $0\leq\max\{0,\theta_x-c\}\leq 1$ is bounded and continuous and the sequence of probability measures $\text{Beta}(\alpha_{nx},\beta_{nx})$ converges in measure to one in which $\theta_x=\mu_x$ almost surely.

Combining the lower bound on $\liminf_{n\to\infty} V_x(\alpha_{nx},\beta_{nx})$ and the upper bound on $\limsup_{n\to\infty} V_x(\alpha_{nx},\beta_{nx})$, we have

$$\frac{1}{1-\gamma_x}\max\{0,\mu_x-c\}\leq\liminf_{n\to\infty} V_x(\alpha_{nx},\beta_{nx})\leq\limsup_{n\to\infty} V_x(\alpha_{nx},\beta_{nx})\leq\frac{1}{1-\gamma_x}\max\{0,\mu_x-c\}.$$

Therefore, the limit of $V_x(\alpha_{nx},\beta_{nx})$ exists and is equal to

$$\lim_{n\to\infty} V_x(\alpha_{nx},\beta_{nx})=\frac{1}{1-\gamma_x}\max\{0,\mu_x-c\}.\ \square$$

We now introduce a preliminary structural result for an stationary optimal policy, which shows that it is optimal to forward items until some stopping

time, after which we discard all items. This stopping time can be infinity, in which case we forward all items. The remark is a special case of the stopping rule shown in Theorem 5.2.2 from Berry and Fristedt (1985).

Remark 2.2.3. *Under any stationary optimal policy, if* $U_{nx} = 0$*, then* $U_{\ell x} = 0$ *for all* $\ell > n$*. Thus, this optimal policy forwards all items until the stopping time* $\inf\{n : U_{nx} = 0\}$*, and discards all subsequent items.*

Proof. To show the claimed characteristic of U^*_{nx}, it is enough to show that $U^*_{n,x} = 0$ implies $U^*_{n+1,x} = 0$. To keep the notation lighter, we suppose that $n = 0$, but the argument is the same for $n > 0$. Let $\pi^{(x)}_{0u}$ be the policy that chooses $U_{1x} = 0$ and $U_{2x} = u \in \{0, 1\}$, and then behaves optimally afterward. Suppose it is optimal to not forward at time 1, that is, $U^*_{1x} = 0$ with $Q_x(\alpha_{0x}, \beta_{0x}, 1) \leq Q_x(\alpha_{0x}, \beta_{0x}, 0)$. Then, since $(\alpha_{1x}, \beta_{1x}) = (\alpha_{0x}, \beta_{0x})$ under the optimal policy, either $\pi^{(x)}_{00}$ or $\pi^{(x)}_{01}$ (or both) is optimal. Suppose for contradiction $\pi^{(x)}_{00}$ is not optimal, then $\pi^{(x)}_{01}$ is strictly better than π_{00}. That is, $Q_x(\alpha_{1x}, \beta_{1x}, 1) > Q_x(\alpha_{1x}, \beta_{1x}, 0)$. But $(\alpha_{1x}, \beta_{1x}) = (\alpha_{0x}, \beta_{0x})$, thus

$$Q_x(\alpha_{0x}, \beta_{0x}, 1) = Q_x(\alpha_{1x}, \beta_{1x}, 1) > Q_x(\alpha_{1x}, \beta_{1x}, 0) = Q_x(\alpha_{0x}, \beta_{0x}, 0),$$

which contradicts the fact that $Q_x(\alpha_{0x}, \beta_{0x}, 1) \leq Q_x(\alpha_{0x}, \beta_{0x}, 0)$. □

The following lemmas (Lemma 2.2.2 and Lemma 2.2.3) state that $V_x(\alpha, \beta)$ is non-decreasing and convex as a function of $\mu(\alpha, \beta)$, holding $\alpha + \beta$ fixed. The proofs of these results use induction arguments to show that the value functions for a finite-horizon truncated version of the single-category problem is non-decreasing and convex. We then take the limit as the truncation point goes to infinity. We use these two lemmas below, in the proof of Theorem 2.2.2, to show that the optimal policy for the single-category sub-problem is a threshold policy, and in the proof of Theorem 2.2.3, to show this threshold is non-decreasing.

Lemma 2.2.2. *$\ell \mapsto V_x(\alpha+\ell, \beta-\ell)$ is non-decreasing for $-\alpha \le \ell \le \beta$ given any $\alpha, \beta > 0$.*

Proof. First, we show the non-decreasing property holds for a finite-horizon problem by induction. Let $V_x(\alpha, \beta, M')$ be the value function for a problem in which we stop at time M and receive terminal reward $\frac{1}{1-\gamma}\max\{0, \mu_{Mx} - c\}$, and our state at time M is $\alpha_{M'x} = \alpha, \beta_{M'x} = \beta$. For the base case at termination $M' = M$, we have $V_x(\alpha+\ell, \beta-\ell, M) = \frac{1}{1-\gamma}\max\left\{0, \frac{\alpha+\ell}{\alpha+\beta} - c\right\}$ is indeed a non-decreasing function of ℓ. Now suppose the property holds for some finite M', with $0 \le M' \le M$. That is, suppose $\ell \mapsto V_x(\alpha+\ell, \beta-\ell, M')$ is non-decreasing. Let us show that it also holds for $M'-1$. Let $\mu(\ell) = \frac{\alpha+\ell}{\alpha+\ell+\beta-\ell} = \mu(0) + \frac{\ell}{\alpha+\beta}$. Then

$$
\begin{aligned}
&V_x(\alpha+\ell, \beta-\ell, M'-1) \\
&\quad = \max\left\{0, \mu(\ell) - c + \gamma_x\left[\mu(\ell)V_x(\alpha+\ell+1, \beta-\ell, M') + (1-\mu(\ell))V_x(\alpha+\ell, \beta-\ell+1, M')\right]\right\}.
\end{aligned}
$$

Let $g(\ell) = V_x(\alpha+\ell, \beta-\ell+1, M')$, where g is non-decreasing by the induction hypothesis. To show that $\ell \mapsto V_x(\alpha+\ell, \beta+\ell, M')$ is non-decreasing, it is sufficient to show that $\ell \mapsto f(\ell) = \mu(\ell)g(\ell+1) + (1-\mu(\ell))g(\ell)$ is non-decreasing. Let $\Delta \ge 0$. Then,

$$
\begin{aligned}
f(\ell+\Delta) - f(\ell) &= \mu(\ell+\Delta)g(\ell+\Delta+1) + (1-\mu(\ell+\Delta))g(\ell+\Delta) - \mu(\ell)g(\ell+1) - (1-\mu(\ell))g(\ell) \\
&= \left[\mu(\ell+\Delta) - \mu(\ell)\right]g(\ell+\Delta+1) + \mu(\ell)g(\ell+\Delta+1) \\
&\quad + \left[(1-\mu(\ell+\Delta)) - (1-\mu(\ell))\right]g(\ell+\Delta) + (1-\mu(\ell))g(\ell+\Delta) \\
&\quad - \mu(\ell)g(\ell+1) - (1-\mu(\ell))g(\ell) \\
&= \left[\mu(\ell+\Delta) - \mu(\ell)\right]\left[g(\ell+\Delta+1) - g(\ell+\Delta)\right] \\
&\quad + \mu(\ell)[g(\ell+\Delta+1) - g(\ell+1)] + (1-\mu(\ell))[g(\ell+\Delta) - g(\ell)] \ge 0,
\end{aligned}
$$

since both $\mu(\ell)$ and $g(\ell)$ are non-decreasing functions of ℓ and $\mu(\ell) \in [0,1]$. We conclude that $V_x(\alpha+\ell, \beta-\ell, M'-1)$ is non-decreasing in ℓ. Thus, by induction,

$V_x(\alpha+\ell,\beta-\ell,M')$ is non-decreasing for all $0 \le M' \le M$. Because the limit of non-decreasing functions is also non-decreasing, we have that $\ell \mapsto V_x(\alpha+\ell,\beta-\ell) = \lim_{M\to\infty} V(\alpha+\ell,\beta-\ell,M)$ is non-decreasing, as stated in Lemma 2.2.2. □

Lemma 2.2.3. *$\ell \mapsto V_x(\alpha+\ell,\beta-\ell)$ is convex for $-\alpha \le \ell \le \beta$ given any $\alpha,\beta > 0$.*

Proof. Similar to the proof of Lemma 2.2.2, we show convexity of a M-step finite-horizon problem $V_x(\alpha,\beta,M)$ by induction, and use that the limit of convex functions is convex. Instead of α and β, let us rewrite the value function of the finite horizon problem as $f_M(\mu) \equiv V_x(\mu\cdot m,(1-\mu)\cdot m,M)$, in terms of $\mu = \frac{\alpha}{\alpha+\beta}$ and $m = \alpha+\beta$. By induction, we show that $f_\ell(\cdot)$ is convex for all $\ell \le M$. At the termination step, $f_M(\mu) = \frac{1}{1-\gamma_x}\max\{0,\mu-c\}$ is convex in μ. Assume that $f_{\ell+1}(\cdot)$ is convex, we show next that $f_\ell(\cdot)$ is convex.

Let $g_\ell(\mu) = \mu - c + \gamma_x\left[\mu\cdot f_{\ell+1}\left(\frac{\mu\ell+1}{\ell+1}\right) + (1-\mu)\cdot f_{\ell+1}\left(\frac{\mu\ell}{\ell+1}\right)\right]$, so $f_\ell(\mu) = \max\{0, g_\ell(\mu)\}$. Since the maximum of convex functions is convex, we only need to show that $g_\ell(\mu)$ is convex, or equivalently, $g_\ell\left(\frac{\mu+\nu}{2}\right) \le \frac{1}{2}(g_\ell(\mu)+g_\ell(\nu))$ for any $0 \le \mu \le 1$ and $0 \le \nu \le 1$ with $\mu \ne \nu$, or equivalently, after cancelling some terms (first subtract $\frac{\mu+\nu}{2} - c$ and then divide by γ_x),

$$\begin{aligned}&\frac{\mu+\nu}{2}\cdot f_{\ell+1}\left(\frac{\frac{\mu+\nu}{2}\ell+1}{\ell+1}\right) + \frac{2-\mu-\nu}{2}\cdot f_{\ell+1}\left(\frac{\frac{\mu+\nu}{2}\ell}{\ell+1}\right)\\ \le&\frac{\mu}{2}\cdot f_{\ell+1}\left(\frac{\mu\ell+1}{\ell+1}\right) + \frac{1-\mu}{2}\cdot f_{\ell+1}\left(\frac{\mu\ell}{\ell+1}\right) + \frac{\nu}{2}\cdot f_{\ell+1}\left(\frac{\nu\ell+1}{\ell+1}\right) + \frac{1-\nu}{2}\cdot f_{\ell+1}\left(\frac{\nu\ell}{\ell+1}\right).\end{aligned} \tag{2.14}$$

Notice that by convexity, we have $f_{\ell+1}\left(\frac{\frac{\mu+\nu}{2}\ell+1}{\ell+1}\right) \le \frac{1}{2}f_{\ell+1}\left(\frac{\mu\ell+1}{\ell+1}\right) + \frac{1}{2}f_{\ell+1}\left(\frac{\nu\ell+1}{\ell+1}\right)$, and $f_{\ell+1}\left(\frac{\frac{\mu+\nu}{2}\ell}{\ell+1}\right) \le \frac{1}{2}f_{\ell+1}\left(\frac{\mu\ell}{\ell+1}\right) + \frac{1}{2}f_{\ell+1}\left(\frac{\nu\ell}{\ell+1}\right)$, then the reminder on the left–hand–side of inequality (2.14) is less than or equal to

$$\frac{\mu+\nu}{4}f_{\ell+1}\left(\frac{\mu\ell+1}{\ell+1}\right) + \frac{\mu+\nu}{4}f_{\ell+1}\left(\frac{\nu\ell+1}{\ell+1}\right) + \frac{2-\mu-\nu}{4}f_{\ell+1}\left(\frac{\mu\ell}{\ell+1}\right) + \frac{2-\mu-\nu}{4}f_{\ell+1}\left(\frac{\nu\ell}{\ell+1}\right). \tag{2.15}$$

Let us take the difference between the right–hand–side of (2.14) and (2.15) and prove that it is nonnegative. That is, let us show

$$\frac{\mu-\nu}{4} f_{\ell+1}\left(\frac{\mu\ell+1}{\ell+1}\right) - \frac{\mu-\nu}{4} f_{\ell+1}\left(\frac{\nu\ell+1}{\ell+1}\right) - \frac{\mu-\nu}{4} f_{\ell+1}\left(\frac{\mu\ell}{\ell+1}\right) + \frac{\mu-\nu}{4} f_{\ell+1}\left(\frac{\nu\ell}{\ell+1}\right) \geq 0,$$

or, equivalently, dividing by $\frac{\mu-\nu}{4}$,

$$f_{\ell+1}\left(\frac{\nu\ell}{\ell+1}\right) + f_{\ell+1}\left(\frac{\mu\ell+1}{\ell+1}\right) \geq f_{\ell+1}\left(\frac{\mu\ell}{\ell+1}\right) + f_{\ell+1}\left(\frac{\nu\ell+1}{\ell+1}\right). \tag{2.16}$$

Without loss of generality, let us assume $\mu > \nu$. Let $\delta = \frac{1}{(\mu-\nu)\ell+1}$ with $0 < \delta \leq 1$, then we can represent, $\frac{\mu\ell}{\ell+1} = \delta\frac{\nu\ell}{\ell+1} + (1-\delta)\frac{\mu\ell+1}{\ell+1}$, and $\frac{\nu\ell+1}{\ell+1} = (1-\delta)\frac{\nu\ell}{\ell+1} + \delta\frac{\mu\ell+1}{\ell+1}$. Since $f_{\ell+1}(\cdot)$ is convex, we have

$$\delta \cdot f_{\ell+1}\left(\frac{\nu\ell}{\ell+1}\right) + (1-\delta) \cdot f_{\ell+1}\left(\frac{\mu\ell+1}{\ell+1}\right) \geq f_{\ell+1}\left(\frac{\mu\ell}{\ell+1}\right),$$

$$(1-\delta) \cdot f_{\ell+1}\left(\frac{\nu\ell}{\ell+1}\right) + \delta \cdot f_{\ell+1}\left(\frac{\mu\ell+1}{\ell+1}\right) \geq f_{\ell+1}\left(\frac{\nu\ell+1}{\ell+1}\right).$$

Combining them, we show equality (2.16) and thus Lemma 2.2.3 holds. □

Before stating the main theorem, we define a function $\mu^*(m)$ for $m > 0$. Let $\mu^*(m)$ be the infimum of $\mu(\alpha,\beta)$ over all states (α,β), $\alpha,\beta > 0$, with $m = \alpha+\beta$ such that it is still optimal to forward at the state. That is,

$$\mu^*(m) = \inf\left\{\mu(\alpha,\beta) : \alpha > 0, \beta > 0, m = \alpha+\beta \text{ and } Q_x(\alpha,\beta,1) > 0\right\}. \tag{2.17}$$

We can think of m as the effective number of observations of paper feedback, (indeed, after observing feedback on n items, the corresponding value of m is $m = \alpha_{nx}+\beta_{nx} = \alpha_{0x}+\beta_{0x}+n$). We can think of $\mu^*(m)$ as the smallest posterior mean such that we would be willing to forward. We see in the following theorem that it is optimal to forward when the posterior mean is above the threshold and discard when it is below.

Theorem 2.2.2. *Let $(\alpha_{nx}, \beta_{nx}) = (\alpha, \beta)$ be the state of category x at some step n with $m = \alpha + \beta$. Then it is optimal to forward the item if $\mu(\alpha_{nx}, \beta_{nx}) \geq \mu^*(\alpha_{nx} + \beta_{nx})$ and discard otherwise. In other words, the following policy is optimal:*

$$U_{nx} = \begin{cases} 1 & \text{if } \mu(\alpha_{nx}, \beta_{nx}) \geq \mu^*(\alpha_{nx} + \beta_{nx}), \\ 0 & \text{otherwise.} \end{cases}$$

This theorem shows that the optimal policy is a threshold policy. This provides a computational benefit, because we only need to store $\mu^*(m)$ for a one-dimensional array of possible values for m, rather than storing $V_x(\alpha, \beta)$ for a much larger two-dimensional array of possible values for (α, β). The proof of the theorem is based on the monotonicity of $V_x(\alpha, \beta)$ in $\mu_x(\alpha, \beta)$, shown in Lemma 2.2.2. Following the lemma, we can see that if it is optimal to forward the item at time n for $\mu^*(m)$, then it is also optimal to forward for any state (α, β) with $\mu(\alpha, \beta) \geq \mu^*(m)$, since the corresponding value function is non-decreasing in $\mu(\alpha, \beta)$.

We can compare Theorem 2.2.2 to the Gittins index policy (Gittins and Jones (1974)). This policy would compute the Gittins index $\nu(\alpha, \beta)$ for the (gross) value of forwarding, $\nu(\alpha, \beta) = \sup_\tau \frac{E[\sum_{n=1}^{\tau} \gamma^{n-1} Y_n | \alpha, \beta]}{E[\sum_{n=1}^{\tau} \gamma^{n-1} | \alpha, \beta]}$, and only forward when $\nu(\alpha, \beta) \geq c$. Both policies are optimal, and $\{(\alpha, \beta) : \nu(\alpha, \beta) \geq c\} = \{(\alpha, \beta) : \mu(\alpha, \beta) \geq \mu^*(\alpha, \beta)\}$, but $\mu^*(\cdot)$ is a function only of the effective number of samples while $\nu(\cdot, \cdot)$ depends on the full state. This offers a storage benefit, as described above. Also, $\mu^*(\cdot)$ can be computed by solving a single dynamic program, while computing $\nu(\cdot, \cdot)$ requires solving many dynamic programs, or doing other additional computation (Gittins, 1979; Katehakis and Veinott, 1987; Gittins et al., 2011).

If the prior distribution on the bandit's mean reward were from a location-scale family conjugate to the bandit reward distribution, as occurs for example

when the prior distribution is normal and the bandit reward is normal with known variance, then one can use linearity of the Gittins index in the posterior's location parameter (Gittins et al. (2011) Section 7 page 192, and discussed in more detail below) to show a results similar to Theorem 2.2.2: that the optimal policy is a threshold policy, and it is optimal to forward if and only if the posterior's location parameter is above the threshold. However, the family of Beta distributions is not a location-scale family.

Lastly, we provide structural properties of $\mu^*(m)$ in Theorem 2.2.3.

Theorem 2.2.3. *$\mu^*(m)$ has the following three properties:*

1. *$\mu^*(m) \leq c$ for any $m > 0$;*
2. *$\mu^*(m) \leq \mu^*(m+1)$ for any $m > 0$;*
3. *$\lim_{m\to\infty} \mu^*(m) = c$.*

To support the intuition behind these structural results, it is useful to decompose the net value of forwarding, $Q(\alpha,\beta,1) - Q(\alpha,\beta,0)$, into two terms: an immediate expected reward $\mu - c$, and the value of information (VOI) for the observed feedback, VOI $= \gamma_x[\mu V_x(\alpha+1,\beta) + (1-\mu)V_x(\alpha,\beta+1) - V_x(\alpha,\beta)]$, where we write $\mu = \mu(\alpha,\beta)$. We obtain these expressions through direct examination of (2.11) and (2.13). The optimal policy forwards whenever the sum of the immediate expected reward and the value of information is non-negative.

We contrast the optimal policy with the *pure exploitation* policy, which ignores the value of information, and considers only the immediate expected reward. The pure exploitation policy forwards if the immediate expected reward is non-negative, i.e., if $\mu(\alpha_{nx},\beta_{nx}) \geq c$, and discards otherwise. This is also a threshold policy, but with a threshold of c.

The first part of Theorem 2.2.3, that $\mu^*(m) \leq c$, shows that the optimal policy's threshold for forwarding $\mu^*(m)$ is at or below the pure exploitation policy's threshold c. Thus, whenever pure exploitation forwards, the optimal policy forwards as well. Moreover, when $\mu(\alpha, \beta)$ is in the range $[\mu^*(m), c)$, the optimal policy forwards even though the pure exploitation policy does not, exploring because the value of the feedback that will result overcomes a negative immediate expected reward.

The second part of Theorem 2.2.3, that $\mu^*(m)$ is non-decreasing, shows that this interval $[\mu^*(m), c)$ is widest, and thus also the optimal policy's willingness to forward is largest, when $m = \alpha + \beta$ is small and we have substantial uncertainty about the user's preference. As $m = \alpha + \beta$ increases, we have more feedback and less uncertainty, and the optimal policy is less willing to explore. A similar structural property for two-arm bandit indices was shown in Theorem 5.3.5 and Theorem 5.3.6 from Berry and Fristedt (1985). These results examine how Gittins index changes as we increase the number of successes, or the number of failures, but not both. Our result implicitly also examines how Gittins index changes as we change the effective number of measurements, but in contrast holds $\mu(\alpha, \beta)$ fixed, which requires changing both the number of successes and the number of failures.

The third part of Theorem 2.2.3, that $\lim_{m \mapsto \infty} \mu^*(m) = c$, shows that as we collect more and more feedback, and learn θ_x with greater and greater accuracy, it becomes optimal to behave like the pure exploitation policy. This is because the pure exploitation policy is optimal when θ_x is known.

Figure 2.2 shows an approximation to the optimal threshold $\mu^*(m)$, and the fixed threshold, c, of the pure exploitation policy in a setting with $\alpha_{0x} = 1$,

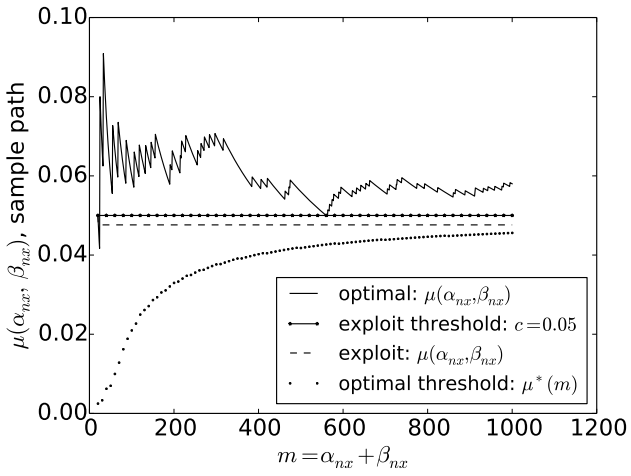


Figure 2.2: Illustration of the optimal policy's threshold, $\mu^*(m)$ (dotted line), and the pure exploitation policy's threshold, c (solid line with '$*$'), with user sample paths, $\mu(\alpha_{nx},\beta_{nx})$, under the optimal policy (solid line) and the pure exploitation policy (dashed line), with $\alpha_{0x} = 1$, $\beta_{0x} = 19$, $c = 0.05$ and $\gamma_x = 0.999$.

$\beta_{0x} = 19$, $c = 0.05$ and $\gamma_x = 0.999$. Although $\mu^*(m)$ is non-decreasing, this approximation to $\mu^*(m)$ oscillates. This is because it is computed by taking the infimum (2.17) only over values of α_{nx},β_{nx} reachable from a single α_{0x},β_{0x}, rather than over all values in $(0,\infty)^2$. We also plot a user sample path, $\mu(\alpha_{nx},\beta_{nx})$, under the optimal policy (solid line) and the pure exploitation policy (dashed line). The pure exploitation policy discards all items, because the initial value of $\mu(\alpha_{0x},\beta_{0x})$ is below c, but the optimal policy forwards items initially, discovers that the user's interest θ_x is larger than originally anticipated, and continues forwarding, thus earning a larger reward.

Conclusions similar to Theorem 2.2.3 may be derived using standard results from the literature when the prior and posterior reside in a location-scale family, as is the case when rewards are normal with known variance, and our prior and posterior on the rewards' unknown mean is also normal. In this case, letting μ and σ^2 be the mean and variance of the posterior, it is known (Gittins et al.

(2011) section 7 on page 192) that the Gittins index $\nu(\mu, \sigma^2)$ satisfies $\nu(\mu, \sigma^2) = \mu + \sigma\nu(0, 1)$ with $\nu(0, 1) \in (0, \infty)$. As noted above, this special structure implies a threshold policy is optimal, with threshold $\mu^*(\sigma^2) = \inf\{\mu : \nu(\mu, \sigma^2) \geq c\} = c - \sigma\nu(0, 1)$. We then immediately have $\mu^*(\sigma^2) \leq c$, $\mu^*(\sigma^2)$ decreasing in σ^2, and $\lim_{\sigma^2 \to 0} \mu^*(\sigma^2) = c$. Thus, Theorem 2.2.3 shows that properties already known for the Normal-Normal setting, and for other location-scale families, also hold in the Beta-Bernoulli setting, which is more natural for the forwarding problem.

2.3 Computation of the single-category value function

We cannot compute the single-category value function $V_x(\alpha, \beta)$ exactly through Bellman's recursion because storing $V_x(\alpha, \beta)$ for all possible values of α and β would require infinite storage. In this section we describe a method for computing an approximation with rigorous error bounds for $V_x(\alpha, \beta)$, from which an approximation to the optimal single-category policy can be computed.

The following lemma follows directly from Proposition 2.2.1 and 2.2.2, where we have loosened the upper bound for easier computation using the inequality $E[\max\{\theta_x - c, 0\}] \leq 1$.

Lemma 2.3.1. *For all $\alpha, \beta > 0$, we have $\frac{1}{1-\gamma_x} \max\{0, \mu(\alpha, \beta) - c\} \leq V_x(\alpha, \beta) \leq \frac{1}{1-\gamma_x}$.*

Proof. At each time step, the stepwise reward function $U_{nx}(Y_{nx} - c)$ is bounded above by 1. Then we have

$$V_x(\alpha, \beta) \quad = \sup_\pi E^\pi \left[\sum_{n=1}^{\infty} \gamma_x^{n-1} (Y_{nx} - c) U_{nx} \mid \theta_x \sim \text{Beta}(\alpha, \beta) \right] \leq \sup_\pi E^\pi \left[\sum_{n=0}^{\infty} \gamma_x \right] = \frac{1}{1-\gamma_x}.$$

For the other side, any policy π' provides a lower bound on the value of an

optimal policy. Thus,

$$V_x(\alpha,\beta) \geq E^{\pi'}\left[\sum_{n=1}^{\infty} \gamma_x^{n-1}(Y_{nx}-c)U_{nx} \mid \theta_x \sim \text{Beta}(\alpha,\beta)\right].$$

Take π' to be the (deterministic) policy that chooses, for each n, $U_{nx} = 1$ if $\mu(\alpha,\beta) > c$, and $U_{nx} = 0$ if $\mu(\alpha,\beta) \leq c$. If $\mu(\alpha,\beta) - c > 0$, its value is

$$E^{\pi'}\left[\sum_{n=1}^{\infty} \gamma_x^{n-1}(Y_{nx}-c) \mid \theta_x \sim \text{Beta}(\alpha,\beta)\right] = \sum_{n=1}^{\infty} \gamma_x^{n-1}(\mu(\alpha,\beta)-c) = \frac{1}{1-\gamma_x}(\mu(\alpha,\beta)-c).$$

If $\mu(\alpha,\beta) - c \leq 0$, its value is 0. Thus, putting both cases together, we see that its value is

$$E^{\pi'}\left[\sum_{n=1}^{\infty} \gamma_x^{n-1}(Y_{nx}-c) \mid \theta_x \sim \text{Beta}(\alpha,\beta)\right] = \frac{1}{1-\gamma_x}\max\{\mu(\alpha,\beta)-c, 0\}.$$

Combining this expression with the previously stated lower bound on $V_x(\alpha,\beta)$ shows that $V_x(\alpha,\beta) \geq \frac{1}{1-\gamma_x}\max\{\mu(\alpha,\beta)-c,0\}$. □

We now introduce an algorithm, Algorithm 1, that defines and computes two quantities, $V_x^U(\alpha,\beta;M)$ and $V_x^L(\alpha,\beta;M)$, which bound the value function $V_x(\alpha,\beta)$ above and below, as stated below in Lemma 2.3.2. $V_x^U(\alpha,\beta;M)$ and $V_x^L(\alpha,\beta;M)$ are value functions for a finite horizon truncated version of the problem, where after being presented with M items we are given a terminal reward, which is the lower bound $\frac{1}{1-\gamma_x}\max\{0,\mu(\alpha_{Mx},\beta_{Mx})\}$ from Lemma 2.3.1 when calculating V_x^L, and the upper bound when calculating V_x^U.

Lemma 2.3.2. *For each $M \geq 0$, $0 \leq \ell \leq M$, and $0 \leq i \leq \ell$ with $\alpha = \alpha_{0x} + i$ and $\beta = \beta_{0x} + \ell - i$, we have $V_x^L(\alpha,\beta;M) \leq V_x(\alpha,\beta) \leq V_x^U(\alpha,\beta;M)$.*

Proof. Consider the base case at termination M. By Lemma 2.3.1, we have

$$\frac{1}{1-\gamma_x}\max\left\{0, \frac{\alpha}{\alpha+\beta} - c\right\} = V_x^L(\alpha,\beta;M) \leq V_x(\alpha,\beta) \leq V_x^U(\alpha,\beta;M) = \frac{1}{1-\gamma_x}.$$

Algorithm 1 Computation of $V_x^L(\alpha,\beta;M)$ and $V_x^U(\alpha,\beta;M)$

Require: $\alpha_{0x}, \beta_{0x}, \gamma_x, c$, and M
  for $i = 0, ..., M$ **do**
    Let $\alpha = \alpha_{0x} + i, \beta = \beta_{0x} + M - i$.
    Let $V_x^L(\alpha,\beta;M) = \frac{1}{1-\gamma_x}\max\left\{0, \frac{\alpha}{\alpha+\beta} - c\right\}$ and $V_x^U(\alpha,\beta;M) = \frac{1}{1-\gamma_x}$.
  end for
  for $\ell = M-1, M-2, ..., 0$ **do**
    for $i = 0, ..., \ell$ **do**
      Let $\alpha = \alpha_{0x} + i, \beta = \beta_{0x} + \ell - i$.
      Let $V_x^L(\alpha,\beta;M) = \max\left\{0, \frac{\alpha}{\alpha+\beta} - c + \gamma_x\left[\frac{\alpha}{\alpha+\beta}V_x^L(\alpha+1,\beta;M) + \frac{\beta}{\alpha+\beta}V_x^L(\alpha,\beta+1;M)\right]\right\}$.
      Let $V_x^U(\alpha,\beta;M) = \max\left\{0, \frac{\alpha}{\alpha+\beta} - c + \gamma_x\left[\frac{\alpha}{\alpha+\beta}V_x^U(\alpha+1,\beta;M) + \frac{\beta}{\alpha+\beta}V_x^U(\alpha,\beta+1;M)\right]\right\}$.
    end for
  end for

Assume that the statement holds at some $0 \le \ell + 1 \le M$, we show that it also holds at ℓ:

$$\begin{aligned} V_x^U(\alpha,\beta;\ell) &= \max\left\{0, \frac{\alpha}{\alpha+\beta} - c + \gamma_x\left[\frac{\alpha}{\alpha+\beta}V_x^U(\alpha+1,\beta;M) + \frac{\alpha}{\alpha+\beta}V_x^U(\alpha,\beta+1;M)\right]\right\} \\ &\ge \max\left\{0, \frac{\alpha}{\alpha+\beta} - c + \gamma_x\left[\frac{\alpha}{\alpha+\beta}V_x(\alpha+1,\beta;M) + \frac{\alpha}{\alpha+\beta}V_x(\alpha,\beta+1;M)\right]\right\} \\ &= V_x(\alpha,\beta) \end{aligned}$$

Similarly,

$$\begin{aligned} V_x^L(\alpha,\beta;M) &= \max\left\{0, \frac{\alpha}{\alpha+\beta} - c + \gamma_x\left[\frac{\alpha}{\alpha+\beta}V_x^L(\alpha+1,\beta;M) + \frac{\alpha}{\alpha+\beta}V_x^L(\alpha,\beta+1;M)\right]\right\} \\ &\le \max\left\{0, \frac{\alpha}{\alpha+\beta} - c + \gamma_x\left[\frac{\alpha}{\alpha+\beta}V_x(\alpha+1,\beta;M) + \frac{\alpha}{\alpha+\beta}V_x(\alpha,\beta+1;M)\right]\right\} \\ &= V_x(\alpha,\beta) \end{aligned}$$

Combining the upper bound and the lower bound, we show that $V_x^L(\alpha,\beta;M) \le V_x(\alpha,\beta) \le V_x^U(\alpha,\beta;M)$ for each $M \ge 0$ and for each (α,β) pair. □

These two bounds provide a computable approximation to the value function, and thus to the optimal policy. The next lemma shows that the gap between these two bounds converges to 0 as the termination step M approaches infinity.

Lemma 2.3.3. *For each $M \geq 0$, $0 \leq \ell \leq M$, $0 \leq i \leq \ell$ with $\alpha = \alpha_{0x} + i$ and $\beta = \beta_{0x} + \ell - i$, we have $V_x^U(\alpha,\beta;M) - V_x^L(\alpha,\beta;M) \leq \gamma_x^{M-\ell}/(1-\gamma_x)$. In particular, $\lim_{M\to\infty} V_x^U(\alpha,\beta;M) - V_x^L(\alpha,\beta;M) = 0$.*

Proof. For any $M \geq 0$, let us first consider the base case at termination M. Then, by definition, for any $0 \leq i \leq M$ with $\alpha = \alpha_{0x} + i$ and $\beta = \beta_{0x} + M - i$, we have $V_x^U(\alpha,\beta;M) = \frac{1}{1-\gamma_x}$ and $V_x^L(\alpha,\beta;M) = \frac{1}{1-\gamma_x}\max\left\{0, \frac{\alpha}{\alpha+\beta} - c\right\} \geq 0$, thus, $V_x^U(\alpha,\beta;M) - V_x^L(\alpha,\beta;M) \leq \frac{1}{1-\gamma_x}$.

Assume that the statement holds for some $0 \leq \ell + 1 \leq M$. We show in the following that the inequality also holds for ℓ by induction. There are two cases to consider at step ℓ:

Case 1: Suppose $V_x^U(\alpha,\beta;M) = 0$. Since $V_x^L(\alpha,\beta;M) \geq 0$, the induction step at ℓ is obviously true.

Case 2: Suppose $V_x^U(\alpha,\beta;M) > 0$. Then,

$$V_x^U(\alpha,\beta;M) = \frac{\alpha}{\alpha+\beta} - c + \gamma_x\left[\frac{\alpha}{\alpha+\beta}V_x^U(\alpha+1,\beta;M) + \frac{\beta}{\alpha+\beta}V_x^U(\alpha,\beta+1;M)\right], \text{ and}$$

$$V_x^L(\alpha,\beta;M) \geq \frac{\alpha}{\alpha+\beta} - c + \gamma_x\left[\frac{\alpha}{\alpha+\beta}V_x^L(\alpha+1,\beta;M) + \frac{\beta}{\alpha+\beta}V_x^L(\alpha,\beta+1;M)\right].$$

Combining them leads to

$$\begin{aligned} V_x^U(\alpha,\beta;M) - V_x^L(\alpha,\beta;M) &\leq \gamma_x\left[\frac{\alpha}{\alpha+\beta}\left[V_x^U(\alpha+1,\beta;M) - V_x^L(\alpha+1,\beta;M)\right]\right. \\ &\qquad \left. + \frac{\beta}{\alpha+\beta}\left[V_x^U(\alpha,\beta+1;M) - V_x^L(\alpha,\beta+1;M)\right]\right] \\ &\leq \gamma_x\left[\left(\frac{\alpha}{\alpha+\beta} + \frac{\beta}{\alpha+\beta}\right)\frac{\gamma_x^{M-(\ell+1)}}{1-\gamma_x}\right] = \frac{\gamma_x^{M-\ell}}{1-\gamma_x}, \end{aligned}$$

while the second inequality is due to the inductive assumption that the statement holds for $\ell + 1$, that is, $V_x^U(\alpha+1,\beta;M) - V_x^L(\alpha+1,\beta;M) \leq \frac{\gamma_x^{M-(\ell+1)}}{1-\gamma_x}$ and

$V_x^U(\alpha, \beta + 1; M) - V_x^L(\alpha, \beta + 1; M) \leq \frac{\gamma_x^{M-(\ell+1)}}{1-\gamma_x}$. Therefore, we conclude that it holds for any $0 \leq \ell \leq M$. Moreover, as $M \to \infty$, the difference between $V_x^U(\alpha, \beta; M)$ and $V_x^L(\alpha, \beta; M)$ converges to 0 for any $0 < \gamma_x < 1$. □

2.4 Simulation Results

In this section we present idealized simulation results for single- and multi-category problems that compare the optimal policy with the pure exploitation policy and two competitive exploration policies: upper confidence bound (UCB) and Thompson sampling. We also evaluate the realism of these idealized simulations by comparing with more realistic trace-driven results. Idealized simulations generate user actions and item properties from the model in Section 2.1 using parameters estimated from historical arXiv.org data, while the trace-driven simulation uses real user histories from arXiv.org.

2.4.1 Idealized Simulation

In this section we present Monte Carlo simulation results under five policies—optimal, pure exploitation, tuned UCB, untuned UCB, and Thompson sampling—in single-category and multi-category problems. In this setting, user histories and item properties are generated according to the assumed model from Section 2.1 using parameters estimated from historical data from the arXiv. Because we simulate from the model, rather than using real user behavior on real items as we do in Section 2.4.2, we call this an "idealized simulation". Our numerical results demonstrate that exploration adds value in a wide variety of

settings, and identify problem characteristics that determine how much value exploration adds.

In the idealized setting, we first show the expected total reward for single-category subproblems $E^{\pi(x)}\left[\sum_{n=1}^{N_x} U_{nx}(Y_{nx} - c)\right]$. Each single-category expected total reward is estimated using $500,000$ independent simulated users, each of which is simulated for one sample path according to the model in Section 2.1. This simulation requires specifying four parameters: hyper-parameters α_{0x}, β_{0x} for the prior beta distribution, a unit forwarding cost c, and a discount factor γ_x. We set $\alpha_{0x} = 1$ and $\beta_{0x} = 19$, which are values typical among those obtained in Section 2.4.2 when fitting to historical data from arXiv.org, and which corresponds to an average user finding one out of 20 items to be relevant. We consider a range of values for γ_x consistent with those estimated from data in Section 2.4.2 for different arXiv categories. For c, we choose a range of values near $c = 0.05$. This value of c is consistent with being indifferent about viewing a stream with 1 relevant item out of every $\frac{1}{0.05} = 20$ shown.

The results of this simulation for the single-category problems under the five policies, optimal, pure exploitation, tuned UCB, untuned UCB, and Thompson sampling, are plotted in Figure 2.3, with 95% confidence intervals shown as error bars. UCB and Thompson sampling are two heuristic exploration policies based on common approaches to exploration vs. exploitation in the broader literature (Lai and Robbins, 1985; Auer et al., 2002; Kaufmann et al., 2012; Thompson, 1933; Agrawal and Goyal, 2011; Chapelle and Li, 2011; Russo and Van Roy, 2014). At each step the UCB policy computes the ρ-quantile, $Q(\rho, \theta_x)$, associated with the posterior distribution of θ_x and forwards the item if $Q(\rho, \theta_x) \geq c$, while the Thompson sampling policy draws a sample, $\hat{\theta}_x$, from the posterior and for-

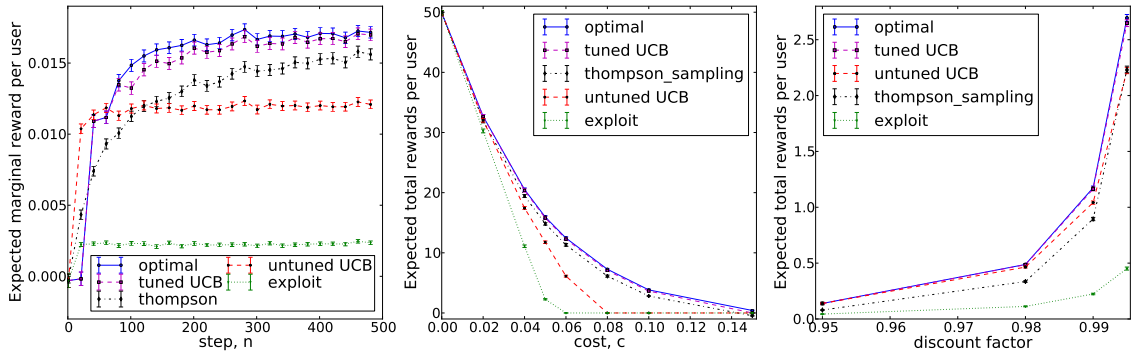


Figure 2.3: Idealized simulation results for the single-category sub-problem with $\alpha_{0x} = 1$ and $\beta_{0x} = 19$. The simulation compares the performance of five policies $\pi^{(x)}$: the optimal policy (denoted "optimal"), tuned UCB, untuned UCB at $\rho = 0.75$, Thompson sampling, and pure exploitation (denoted "exploit"). Tuned UCB runs the simulation for various ρ in the range $\{0.65, 0.7, 0.75, 0.8, 0.85, 0.9, 0.95, 0.99\}$ with the best results reported. Each error bar is a 95% confidence interval. *(a)* The left plot shows expected marginal reward, $E^{\pi(x)}[U_{nx}(Y_{nx} - c)]$, under each policy $\pi^{(x)}$ at each step n with a unit forwarding cost of $c = 0.05$ and a discount factor of $\gamma_x = 0.999$. *(b)* The middle plot shows expected total reward, $E^{\pi(x)}\left[\sum_{n=1}^{N_x} U_{nx}(Y_{nx} - c)\right]$, versus unit forward cost c ranging from 0 to 0.15 with a discount factor of $\gamma_x = 0.999$. *(c)* The right plot shows expected total reward versus discount factor, γ_x, ranging from 0.95 to 0.995 with a unit forwarding cost of $c = 0.05$.

wards if $\hat{\theta}_x \geq c$. The "tuned UCB" policy refers to the UCB policy where ρ is tuned for each given set of problem parameters (γ_x, α_{0x}, β_{0x}, c: $x = 1, ..., k$) by using simulation to try several ρ in the range $\{0.65, 0.7, 0.75, 0.8, 0.85, 0.9, 0.95, 0.99\}$ and using the value that performs best while in the "untuned UCB" policy $\rho = 0.75$ is chosen and fixed for all c and γ_x in Figure 2.3 since $\rho = 0.75$ is best for smaller unit costs and discount factors.

Figure 2.3(a) shows the expected marginal reward $E^{\pi(x)}[U_{nx}(Y_{nx} - c)]$ at each time step $n \in \{1, .., 500\}$ with a cost of $c = 0.05$ and a discount factor of $\gamma_x = 0.999$. In this setting, tuned UCB chooses $\rho = 0.95$ while untuned UCB sets $\rho = 0.75$. In the earlier steps, the expected marginal reward of the optimal policy and tuned UCB suffers from extensive exploration compared to pure exploitation,

Thompson sampling, and untuned UCB. But as we start collecting information about users over the next 50 steps, the expected marginal reward of the optimal policy and tuned UCB quickly recovers from the previous loss and rapidly surpasses that of the other policies, eventually stabilizing as feedback becomes abundant, and the optimal policy stops exploring and exploits the information it has gained.

In addition to the expected marginal reward, Figure 2.3(b) and Figure 2.3(c) show sensitivity plots of the expected total reward. Figure 2.3(b) shows the expected total reward against different unit costs c for a fixed discount factor $\gamma_x = 0.999$, while Figure 2.3(c) shows the expected total reward per user against different discount factors γ_x with a fixed unit cost $c = 0.05$. In the single-category subproblems, we observe that the optimal policy is almost identical to tuned UCB, and performs statistically better than Thompson sampling, untuned UCB, and pure exploitation. The difference between the optimal policy and tuned UCB becomes statistically significant in multi-category problems, as illustrated in Figure 2.3, because different optimal tunings of ρ are required for categories with different discounts. In Figure 2.3(b), the expected total reward decreases as cost increases for all five policies, which is intuitive because each forwarded item provides a smaller (net) reward when costs are higher. The optimal policy and tuned UCB performs at least as well as the other three policies for all cost values. At the two extremes when the cost is close to either 0 or 1, all five policies make the same forwarding decision, i.e. to forward all available items when the cost is near 0, or forward none when the cost is near 1. This explains why the five policies have the same expected total reward near the two extremes. As we move away from the extremes, the optimal policy provides a greater competitive advantage, by optimally balancing exploration and

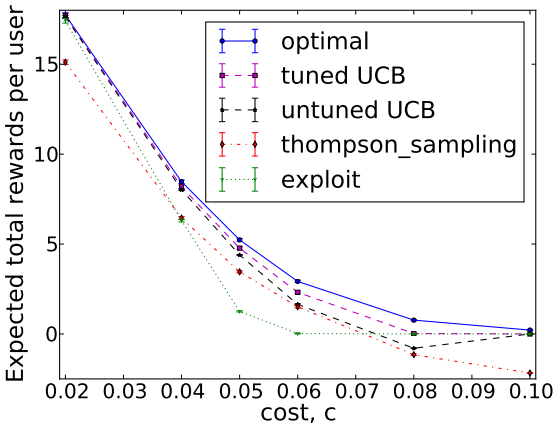


Figure 2.4: Idealized simulation results in a multi-category problem, with a mixture of 20 categories at $(\alpha_{0x}, \beta_{0x}, \gamma_{0x}) = (1, 19, 0.95)$ and one category at $(\alpha_{0x}, \beta_{0x}, \gamma_{0x}) = (1, 19, 0.995)$. The simulation compares the performance of five policies π: the optimal policy, tuned UCB, untuned UCB with $\rho = 0.85$, pure exploitation, and Thompson sampling. Each error bar is a 95% confidence interval. The plot shows expected total reward, $E^{\pi}\left[\sum_{n=1}^{N} U_n(Y_n - c)\right]$, versus unit forwarding cost c ranging from 0.02 to 0.1. In tuned UCB, simulations are run for a range of ρ-quantiles, $\{0.65, 0.7, 0.75, 0.8, 0.85, 0.9, 0.95, 0.99\}$, with the best expected total rewards reported in the figure for each cost, c. Untuned UCB sets $\rho = 0.85$ since it performs the best in the category with $\gamma_{0x} = 0.995$.

exploitation. It outperforms pure exploitation the most when the unit cost is near $c = 0.05$, because exploration provides the most benefit in the most ambiguous case when c is close to the expected value of θ_x, and our prior has mean $\mu(\alpha_{0x}, \beta_{0x}) = \frac{1}{1+19} = 0.05$. Compared to tuned UCB, fixing $\rho = 0.75$ for all c in untuned UCB deteriorates the performance.

In Figure 2.3(c), we observe that the expected total reward increases with the discount factor, γ_x, for all policies, but the optimal policy and tuned UCB have a much steeper positive slope as the discount factor approaches 1. This suggests that the optimal policy benefits the most from exploration when the discount factor is large, since the system has a longer time horizon to learn users' preferences and recover any losses suffered at the beginning when learning.

In Figure 2.4 we compare the performance of the optimal policy with the other four policies for the idealized simulation in a multi-category problem. The constructed multi-category problem consists of 20 categories at $(\alpha_{0x}, \beta_{0x}, \gamma_x) = (1, 19, 0.95)$ and one category at $(\alpha_{0x}, \beta_{0x}, \gamma_x) = (1, 19, 0.995)$. Untuned UCB sets $\rho = 0.85$ since it is best when $\gamma_{0x} = 0.995$. Figure 2.4 shows that tuned UCB, with the flexibility of tuning ρ, is comparable to the optimal policy when the unit forwarding cost is near 0, but its difference from the optimal policy enlarges and becomes statistically significant as the unit forward cost increases. Tuning UCB improves its performance over untuned UCB, but does not bring it as close to optimal as it did in the single-category case. The reason is that the right balance between exploration vs. exploitation differs across item category, preventing UCB from achieving this balance with a single value of ρ.

One could imagine UCB with a more elaborate tuning method, in which ρ is allowed to differ across item category, and is tuned separately using simulation optimization for each user and category based on α_{0x}, β_{0x}, γ_x, c, but using this heuristic approach would not offer any significant computational advantage over simply using the optimal policy.

2.4.2 Trace-driven Simulation

In this section, we present trace-driven simulation results using the web server logfile from arXiv.org for both the optimal policy and the pure exploitation policy. Instead of using simulated users and items as in Section 2.4.1, we use real historical user interactions and items extracted from this logfile. All identifiable information from users is hashed to protect user privacy. In contrast with the

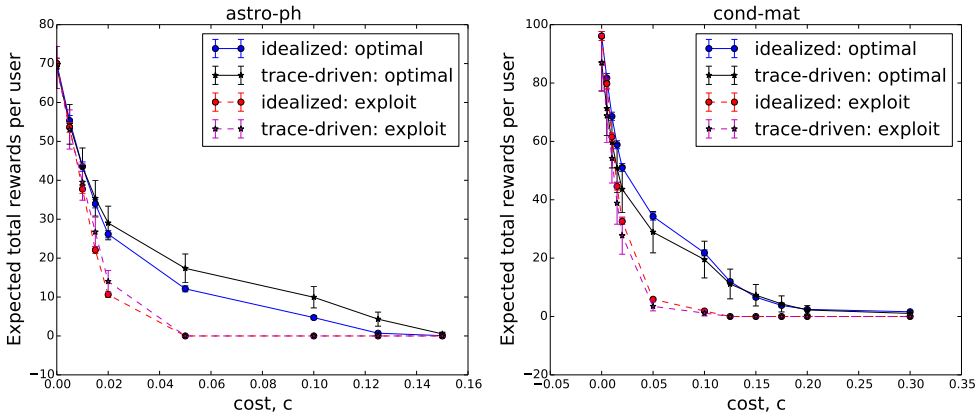


Figure 2.5: This figure plots expected cumulative reward against the unit cost of forwarding c under the trace-driven simulation and the idealized Monte-Carlo simulation for (a) astrophysics (astro-ph, left), and (b) condensed matter (cond-mat, right). Results are presented for both the optimal policy and the pure exploitation policy. 95% confidence intervals are shown as error bars.

focus on various policies in Section 2.4.1, we choose only the optimal policy and the pure exploitation policy in this section to test strategies at the two ends of the spectrum of performance, to understand better how exploration adds value in a more realistic setting and how predictions from our model match reality.

ArXiv items are categorized by their authors into one of 18 subjects (astrophysics, condensed matter, computer science, ...) and then again into one of several categories within the subject (for example, the condensed subject has nine categories: Disordered Systems and Neural Networks, Materials Science, Mesoscale and Nanoscale Physics, ...). In addition to this "primary" subject/category, authors may optionally provide one or more secondary subject/category labels.

Each subject has associated "new" and "recent" webpages, which show items submitted during the previous day and week, respectively. Some large subjects, e.g., astrophysics and condensed matter, have over 100 items submit-

ted per day. In our trace-driven simulation, we take items submitted to a single subject as our item stream, and use the author-provided primary category within this subject as our item category. We use this simple pre-existing categorization method to focus attention on the exploration vs. exploitation tradeoff rather than the categorization scheme, although one could easily use categories learned automatically from item content and/or historical co-access data (Manning et al., 2008).

In our experiments, we consider two separate item streams, astrophysics and condensed matter, and look at items submitted in 2009 and 2010. In each of these subjects, we consider an item to be presented to the user if he/she visited the subject's new or recent page during the period it was posted there. If the user clicked on the link to the full text or abstract from the new or recent page during this period, we consider the item to be relevant. If he/she did not click on this link, we consider the item to be irrelevant. If the user did not visit the new or recent page during the period the item was posted there, we consider the item's relevance to be unobserved. We then identify those users who visited the subject's new and recent pages a moderate number of items over the time period of interest (2009-2010), removing those who visited too infrequently (less than 30 visits) as not providing useful data, and those who visited too often (more than 510 visits) as likely robots.

In the next step, we randomly assign each extracted user into one of two groups: training and testing. The training users are used to estimate hyper–parameters α_{0x}, β_{0x}, γ_x for each category x, and then the simulation is executed among the testing users. For each training user u, we first estimate his/her $\theta_{u,x}$ associated with each category x (we have added u to the subscript for θ_x, and

below for N_x, to emphasize the dependence on the user) by calculating his/her click-through-rate on the items presented from that category. We then estimate $\alpha_{0,x}$ and $\beta_{0,x}$ for each category x by fitting a beta distribution to the histogram of estimated $\theta_{u,x}$ over all training users u. Similarly, to estimate γ_x for each category x, we first count the total number $N_{u,x}$ of items submitted to the category over each user u's lifetime in the system, and fit a geometric distribution with parameter $1 - \gamma_x$ to the histogram of $N_{u,x}$ among the training users.

After the parameter estimation, we then perform the trace-driven simulation for each testing user as follows. Iterating through those dates on which the user visited the category's new or recent page, we take all the items that were shown on those pages, and decide sequentially whether to forward that the item to the user according to one of the two policies. If the user clicked, according to the historical data, on the link to the full text or the abstract of the forwarded item over the period when the item was posted on the new or recent page, we conclude that the user found the item to be relevant; otherwise, if the user did not click, we conclude that the item was irrelevant to the user. Feedback is collected immediately and given to the policy, and the process is repeated for the next item, until the user leaves the system.

For a side-by-side comparison, we also perform an idealized simulation for the multi-category problem to check how trace-driven results differ from the idealized results. This idealized simulation follows the same framework as in Section 2.4.1 but uses multiple categories, and uses the same values for α_{0x}, β_{0x}, and γ_x as the trace-driven simulation.

Figure 2.5 shows the expected total reward per user versus the unit forwarding cost c under the two simulations, with error bars representing 95% confi-

dence intervals. The patterns observed are similar to those in Figure **??**(b): the two policies perform similarly when the unit cost is near 0 or 1; and the optimal policy substantially outperforms pure exploitation when c is away from the two extremes.

For each policy, the trace-driven results are close to the idealized simulation results, with much of the discrepancy explained by sampling error. However, some of the discrepancy is likely due to violations of our modeling assumptions by the historical data. In our model, we make the four main modeling assumptions: (1) arrivals of items follow a Poisson process and the lifetime of a user in the system is exponentially distributed, so that the total number of items in a category viewed by each user, N_x, follows a geometric distribution; (2) the prior distribution on θ_x in each category follows a Beta distribution; (3) the θ_x are independent across categories; (4) the number of items in the category x viewed by the user, N_x, and probability of relevance of an item from category x to the user, θ_x, are independent. We performed some empirical checks on the historical data to validate these assumptions. Assumptions (1) and (2) seem to be met reasonably well, but we saw some violations of (3) and (4). We leave extensions of our model to incorporate more general assumptions to future work.

Despite these violations of our modeling assumptions, we see that simulation model matches well with the behavior of the more realistic trace-driven simulation, and that the policy calculated to be optimal in our model provides similar improvements over pure exploitation in both trace-driven and idealized simulations.

CHAPTER 3

GENERALIZED INFORMATION FILTERING PROBLEM: PERIODIC REVIEWS

This chapter builds on Chapter 2, which considers an information filtering problem in a Bayesian setting, and uses dynamic programming to find the Bayes-optimal strategy for trading exploration and exploitation. There are two main differences between the model considered in that chapter, and the one considered in the current chapter. First, in Chapter 2, users provide *immediate* feedback on forwarded items, while in the current chapter, we allow items to queue in the system until the next user visit. It is only upon visiting the system that the user provides feedback. This "periodic review" assumption is more realistic in many information filtering systems. Second, Chapter 2 assumes that users provide a unit cost for forwarding to the information filtering system, while in the current chapter we provide a method for ranking results that allows this cost to be unknown.

In Section 3.1, we formulate the information filtering problem with periodic reviews and a fixed unit cost for forwarding an item. In Section 3.2, we consider the case where the unit cost is unknown and there is a budget constraint on the total number of items that a user can view during each review. We then show how to derive a ranking from this budgeted problem. Lastly, we show experimental results in Section 3.4.

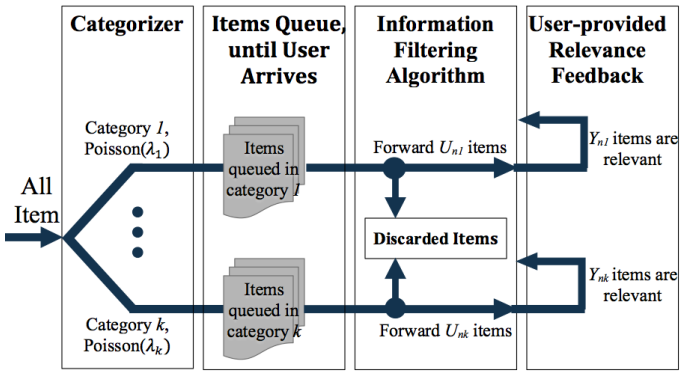


Figure 3.1: Schematic of the information filtering problem with periodic reviews.

3.1 Mathematical Model with a Unit Cost for Forwarding

We assume that items arrive to the system according to a Poisson process with rate $\lambda > 0$. Each item is categorized into (exactly) one of k categories, $\{1, ..., k\}$, and the category is observed as it enters the system. For systems without explicit categorization, the categories could be obtained by running a clustering algorithm on previously collected items in a pre-processing step. The availability of categories is also assumed in Chapter 2. See Chapter 4 for an example of categorization method and see Chapter 5 for the details of how it was applied to arXiv.org.

Using similar notation to Chapter 2, we let X_i denote the category of the i^{th} arriving item. We assume that the X_i are independent and identically distributed, and we let $p_x = P(X_i = x) > 0$. Thus, items in each category $x \in \{1, ..., k\}$ arrive according to a Poisson process with rate $\lambda_x = p_x\lambda$. In contrast to the discrete time case considered in Chapter 2, in this chapter we use Poisson processes to model item arrivals. The discrete-time problem considered in Chapter 2 can also be derived from a model in which items arrive in continuous time according to a

Poisson process and are reviewed instantly. In this case, the discrete-time formulation is obtained by counting item arrivals.

Similarly to Chapter 2, we focus on one user. For each user, we assume that each category x has some latent unobserved value $\theta_x \in [0, 1]$ measuring the probability that an item from category x is relevant to the user. Let $\theta = [\theta_1, ..., \theta_k]$. We place a Bayesian prior distribution on this θ, given by $\theta_x \sim \text{Beta}(\alpha_{0x}, \beta_{0x})$, with independence across x, for some parameters α_{0x} and β_{0x}, typically estimated using historical data from (other) long-time users on older items. In our model, θ is assumed to stay static over the user's lifetime.

The optimal tradeoff of exploration vs. exploitation will depend on how long the user interacts with our stream of items. Let T be the length of time that the user uses our information filtering system. T is unknown a priori, and we model it as an exponential random variable with parameter r.

The previous model in Chapter 2 provides a Bayes-optimal algorithm that analyzes the situation in which the user is always available to provide immediate feedback on each arriving item. However, in many real systems, users do not behave like this. Instead, they arrive periodically to review items that have queued in the system since their last visit.

Different from the "immediate response" assumed in Chapter 2, the model in this chapter assumes that the user visits the system at time points separated by exponentially distributed inter-arrival times, which are independent and have rate s. Let N be the number of user visits before T. Here, we assume that at each visit, he or she examines all items forwarded from the stream since the last visit and provides binary relevance feedback for each. In Section 3.2, we study

a problem variant in which the number of items the user is willing to examine on each visit is constrained.

At the n^{th} user visit, the posterior on θ_x is Beta(α_{nx},β_{nx}), for some α_{nx}, β_{nx}. All items that have arrived since the last user visit are queued, while all unforwarded items from before this last visit have already been discarded and are not queued. Let L_{nx} be the number of items queued in category x at the start of the n^{th} user visit, and let $\tilde{L}_{nx} = \min(\bar{L}, L_{nx})$, with a fixed $\bar{L} > 0$ that bounds the $\tilde{L}_{nx}$ for computational convenience. We are allowed to set $\bar{L}$ to be infinity in the theory, but have $\bar{L} < \infty$ in the computation. $\bar{L}$ can be taken to be large, in which case it has little impact in practice. Based on $(\alpha_{nx}, \beta_{nx})$ and the observed $\tilde{L}_{nx}$, we choose $U_{nx} \in \{0, ..., \tilde{L}_{nx}\}$, which is the number of items to forward to the user from category x.

With the decision U_{nx}, the user provides explicit feedback, denoted by Y_{nx}, that counts the number of relevant items forwarded. The posterior we get from observing this is the same as if we observed relevance of each item. Conditioning on U_{nx} and θ_x, Y_{xn} is binomial,

$$Y_{nx} \mid \theta_x, U_{nx} \sim \text{Binomial}\,(U_{nx}, \theta_x)\,.$$

The posterior we get from observing this is the same as if we observed relevance of each item.

We also assume there is a unit cost, c, for forwarding each item to the user and we get a reward of 1 for each relevant item shown. Thus, we collect reward $Y_{nx} - cU_{nx}$ in each step. We define a policy π as a sequence of functions, $(\pi_1, \pi_2, ...)$, where each $\pi_n : (\mathbb{N}^k \times \mathbb{N}^k)^{n-1} \times \mathbb{N}^k \mapsto \mathbb{N}^k$ maps history, $\{U_{\ell x}, Y_{\ell x} : \ell \leq n-1,\ x \in \{1, ..., k\}\}$, and the new observation $\{\tilde{L}_{nx} : x \in \{1, ..., k\}\}$ into actions. Let Π be the set of all such policies. Our objective is to find an optimal policy $\pi \in \Pi$ that

maximizes total expected reward:

$$\sup_{\pi\in\Pi} \mathbb{E}^{\pi}\left[\sum_{x=1}^{k}\sum_{n=1}^{N}(Y_{nx} - cU_{nx})\right]. \tag{3.1}$$

3.1.1 Solution and Computation Method

We can not solve (3.1) directly as the size of categories increases due to curse of dimensionality (Powell, 2007). However, because of the independence assumption across θ_x, we can decompose the original problem with k-categories into a sum of k independent sub-problems, each of which can be solved via stochastic dynamic programming. Equation 3.1 is rewritten as,

$$\sup_{\pi\in\Pi} E^{\pi}\left[\sum_{x=1}^{k}\sum_{n=1}^{N}(Y_{nx} - cU_{nx})\right] = \sum_{x=1}^{k}\sup_{\pi(x)\in\Pi(x)} E^{\pi(x)}\left[\sum_{n=1}^{N}(Y_{nx} - cU_{nx})\right],$$

where a single-category policy $\pi(x)$ is a sequence of functions, $(\pi_1(x), \pi_2(x), ...)$, associated with category x and $\Pi(x)$ is the set of all $\pi(x)$. Each $\pi_n(x) : (\mathbb{N}\times\mathbb{N})^{n-1}\times\mathbb{N} \mapsto \mathbb{N}$ maps the single-category history, $\{U_{\ell x}, Y_{\ell x} : \ell \leq n-1\}$, with the new observation $\tilde{L}_{nx}$ into actions U_{nx} for category x.

Similar to Section 2.2.2 in Chapter 2, we can convert each sub-problem from a problem with a random finite time horizon to one with an infinite time horizon as follows. First, N follows a geometric distribution with parameter $1-\gamma$, where $\gamma = \frac{s}{s+r}$. Then,

$$E^{\pi(x)}\left[\sum_{n=1}^{N}(Y_{nx} - cU_{nx})\right] = \gamma E^{\pi(x)}\left[\sum_{n=1}^{\infty}\gamma^{n-1}(Y_{nx} - cU_{nx})\right].$$

We now solve this Markov Decision process using stochastic dynamic programming. This is tractable because the state space of the single-category sub-

problem is smaller. We define the value function

$$V_x(\alpha,\beta) = \sup_{\pi(x)\in\Pi(x)} E^{\pi(x)}\left[\sum_{n=1}^{\infty}\gamma^{n-1}(Y_{nx}-cU_{nx})\Big|\alpha_{0x}=\alpha,\beta_{0x}=\beta\right] \tag{3.2}$$

$$= \sum_{l=0}^{\bar{L}} P(\tilde{L}_{nx}=\ell)V_x(\alpha,\beta|\ell) \tag{3.3}$$

where $V_x(\alpha,\beta|\ell)$ is the conditional value function when the size of available items in this period is ℓ. It is not hard to show that L_{nx} follows a geometric distribution with parameter $\xi_x = \frac{s}{\lambda_x+s}$, and thus the distribution of $\tilde{L}_{nx} = \min(\bar{L}, L_{nx})$ is given by:

$$P(\tilde{L}_{nx}=\ell) = \begin{cases} (1-\xi_x)^{\ell}\xi_x & \text{if } 0\le\ell<\bar{L}, \\ (1-\xi_x)^{\ell} & \text{if } \ell=\bar{L}. \end{cases}$$

The Bellman equation for the conditional value function $V_x(\alpha,\beta|\ell)$ is:

$$V_x(\alpha,\beta|\ell) = \max_{u\in\{0,1,\ldots,\ell\}} Q_x(\alpha,\beta,u), \tag{3.4}$$

and Q-factor, $Q(\alpha,\beta,u)$, is the expected reward when we forward u items and behave optimally afterwards:

$$\begin{aligned} Q_x(\alpha,\beta,u) =& E[Y_{1x}-cu+\gamma V_x(\alpha_{1x},\beta_{1x})|U_{1x}=u,\alpha_{0x}=\alpha,\beta_{0x}=\beta] \\ =& \left(\frac{\alpha}{\alpha+\beta}-c\right)u+\gamma E[V_x(\alpha_{1x},\beta_{1x})|U_{1x}=u,\alpha_{0x}=\alpha,\beta_{0x}=\beta]. \end{aligned} \tag{3.5}$$

The last term in the Q-factor equation (3.5) specifies the expected future reward for forwarding u items,

$$\begin{aligned} & E[V_x(\alpha_{1x},\beta_{1x})|U_{1x}=u,\alpha_{0x}=\alpha,\beta_{0x}=\beta] \\ = & \sum_{i=0}^{u} P\left(Y_{1x}=i|U_{1x}=u,\alpha_{0x}=\alpha,\beta_{0x}=\beta\right)V_x(\alpha+i,\beta+u-i), \end{aligned}$$

and one can show that $P\left(Y_{1x}=i \mid U_{1x}=u,\alpha_{0x}=\alpha,\beta_{0x}=\beta\right) = \binom{u}{i}\frac{B(\alpha+i,\beta+u-i)}{B(\alpha,\beta)}$, where $B(x,y)$ is the beta function.

Condition on $\tilde{L}_{nx} = \ell$, an optimal policy is then any decision that attains the maximum in the recursion, breaking ties arbitrarily. That is,

$$U_{n+1,x}(\ell) \in \text{argmax}_{u\in\{0,...,\ell\}} Q_x(\alpha_{nx}, \beta_{nx}, u). \tag{3.6}$$

Thus, if we are able to compute Q-factor, $Q_x(\alpha_{nx}, \beta_{nx}, u)$, we can then compute an optimal policy. Similar to the situation described in Section 2.2.3, the state space of $(\alpha_{nx}, \beta_{nx})$ in this single-category subproblem is theoretically countable infinite, which prevents us to compute the value function exactly. However, many approximation tools have been developed to approximate the value function (Katehakis and Veinott, 1987; Powell, 2007).

Algorithm 2 Computation of $V_x^L(\alpha, \beta; \tilde{T})$ and $V_x^U(\alpha, \beta; \tilde{T})$, where $V_x^L(\alpha, \beta; \tilde{T}) \leq V_x(\alpha, \beta) \leq V_x^U(\alpha, \beta; \tilde{T})$.

Require: γ, α_{0x}, β_{0x}, ξ_x, c, and N
for $i = \tilde{T} + \bar{L}, ..., 0$ **do**
  for $j = 0, ..., i$ **do**
    Let $\alpha = \alpha_{0x} + j, \beta = \beta_{0x} + i - j$, and $\mu = \frac{\alpha}{\alpha+\beta}$.
    for $\tilde{L}_{nx} = \{1, ..., \bar{L}\}$ **do**
      if $i > \tilde{T}$ **then**
        Let $V_x^L\left(\alpha, \beta; \tilde{T} \middle| \ell\right) = \ell \frac{\max\{0, \mu - c\}}{(1-\gamma)(1-\gamma\xi_x)}$,
        and $V_x^U\left(\alpha, \beta; \tilde{T} \middle| \ell\right) = \frac{\ell}{(1-\gamma)(1-\gamma\xi_x)}$.
      else
        Let $V_x^L(\alpha, \beta; \tilde{T}|\ell) = \max_{0\leq u\leq \ell} \left\{(\mu - c)u + \gamma E[V_x^L(\alpha_{1x}, \beta_{1x}; \tilde{T})|U_{1x} = u]\right\}$,
        and $V_x^U(\alpha, \beta; \tilde{T}|\ell) = \max_{0\leq u\leq \ell} \left\{(\mu - c)u + \gamma E[V_x^U(\alpha_{1x}, \beta_{1x}; \tilde{T})|U_{1x} = u]\right\}$.
      end if
    end for
    Let $V_x^L(\alpha, \beta; \tilde{T}) = \frac{1}{1-\gamma\xi_x} \sum_{l=1}^{\bar{L}} P(\tilde{L}_{1x} = \ell) V_x^L(\alpha, \beta; \tilde{T}|\ell)$,
    and $V_x^U(\alpha, \beta; \tilde{T}) = \frac{1}{1-\gamma\xi_x} \sum_{\ell=1}^{\bar{L}} P(\tilde{L}_{1x} = \ell) V_x^U(\alpha, \beta; \tilde{T}|\ell)$.
  end for
end for

Illustrated in Algorithm 2, we provide an approximation in solving the value function for the single-category subprobelm. Intuitively, we use backward induction to solve the dynamic program in equation 3.3 by considering a trun-

cated time-horizon problem terminated at $\tilde{T}$, and averaging an upper bound on the value function, $V_x^L(\alpha,\beta;\tilde{T})$ and a lower bound, $V_x^U(\alpha,\beta;\tilde{T})$. As the truncation point $\tilde{T}$ grows large, the gap $V_x^U(\alpha,\beta;\tilde{T}) - V_x^L(\alpha,\beta;\tilde{T})$ shrinks to 0, and so we can calculate $V_x(\alpha,\beta)$ to arbitrary accuracy.

3.1.2 Structural Results

In this section, we show structural results for the mathematical model with a unit cost for forwarding: first, $V_x(\alpha,\beta|\ell)$ is a non-decreasing function of ℓ; secondly, for any state (α,β), $V_x(\alpha+\ell,\beta-\ell)$ is non-decreasing and convex in ℓ for $0 \le \ell < \beta$.

Lemma 3.1.1. *Given* (α,β) *with* $\alpha,\beta \in \mathbb{R}_+$, $V_x(\alpha,\beta|\ell) \le V_x(\alpha,\beta|\ell')$ *for all* $\ell < \ell'$.

Proof. Given any $\ell' = \ell + 1$, the conditional value function is expressed as

$$\begin{aligned} V_x(\alpha,\beta|\ell') &= \max_{u\in\{0,\dots,\ell'\}} Q(\alpha,\beta,u) \\ &= \max\left\{V_x(\alpha,\beta|\ell),\ Q(\alpha,\beta,\ell+1)\right\} \ge V_x(\alpha,\beta|\ell). \ \square \end{aligned}$$

The above lemma, that $V_x(\alpha,\beta|\ell)$ is non-decreasing in ℓ, reduces the computational complexity of the value function. Since at every state (α,β), we can iterate through $\ell \in \{0,\dots,\bar{L}\}$ to compute $Q(\alpha,\beta,\ell)$, and then conditional value function $V_x(\alpha,\beta|\ell)$ is just maximum of two previous $V_x(\alpha,\beta|\ell-1)$ and $Q(\alpha,\beta,\ell)$. This reduces the computational complexity of $V(\alpha,\beta)$ on the size of arriving items, $\bar{L}$, from $O(\bar{L}^2)$ to $O(\bar{L})$.

Before stating the next two lemmas, we first reparametrize the value function $V_x(\alpha,\beta)$ in terms of the effective number of observations, $m = \alpha+\beta$, and

the posterior mean of θ_x, $\mu = \alpha/m$. That is, $V_x(\mu, m)$ is the value function at state (α, β). This is an abuse of notation, and which version ($V_x(\alpha, \beta)$ or $V_x(\mu, m)$) we mean will be clear from context later. In Section 3.1, we assume that $Y_{nx}|\theta_x, U_{nx} \sim$ Binomial(U_{nx}, θ_x) while the posterior of θ_x condition on data is a beta distribution, $\theta_x|\{U_{\ell x}\}_{\ell\in\{1,\ldots,n-1\}}, \{Y_{\ell x}\}_{\ell\in\{1,\ldots,n-1\}} \sim$ Beta$(\alpha_{nx}, \beta_{nx})$. Marginalizing the relevance probability θ_x, the observation Y_{nx} is indeed a beta-binomial distribution with U_{nx} number of Bernoulli trails, in which the probability of success is unknown and follows a beta distribution with parameters $(\alpha_{nx}, \beta_{nx})$. In the following lemmas, we use $Y \sim F(u, \alpha, \beta)$ to represent a beta-binomial distribution with a known number, u, of Bernoulli trails, in which the probability of success in each trial is unknown and follows a beta distribution with parameters (α, β). When parametrizing (α, β) by $m = \alpha + \beta$ and $\mu = \frac{\alpha}{\alpha+\beta}$, we have $Y \sim F(u, m\mu, m(1-\mu))$. Below, we show that the value function, $V_x(\mu, m)$, is non-decreasing in μ.

Lemma 3.1.2. *For each fixed $m \in \mathbb{R}_+$, $V_x(\mu, m)$ is non-decreasing in $\mu \in (0, 1)$.*

Proof. We first show the statement for a truncated T-finite-horizon problem, in which we stop collecting additional information at a large termination step T, and receive terminal reward 0. It does not matter what the termination reward is, because the termination reward shrinks to 0 exponentially as T grows. Given $0 \le t \le T$, let $V_x(\mu, m; t)$ denote the value function for the problem with t horizons left at the state (μ, m), where $m = \alpha + \beta$ and $\mu = \alpha/m$. The recursion of $V_x(\mu, m; t)$ is defined as:

$$\sum_{\ell=0}^{L} P(\tilde{L}_{nx} = \ell) \max_{u\in\{0,\ldots,\ell\}} \left\{(\mu - c)u + \gamma E\left[V_x\left(\frac{m\mu + Y}{m+u}, m+u; t+1\right)|Y \sim F(u, m\mu, m(1-\mu))\right]\right\} \quad (3.7)$$

Let us fix m. At the termination step when $t = 0$, $V_x(\mu, m; 0)$ is non-decreasing in μ since there are all zeros. We will use induction, assuming that $V_x(\mu, m; t')$ is

non-decreasing in μ for all $t < t' < T$ with $t \geq 0$, and then show that the statement holds for $V_x(\mu, m; t)$. By Bellman's equation, the value function $V_x(\mu, m; t)$ is expressed in (3.7). The first term, $(\mu - c)u$, in the maximization term of (3.7) is non-decreasing in μ with fixed m, so it is sufficient to show that

$$E\left[V_x\left(\frac{m\mu + Y}{m+u}, m+u; t+1\right) | Y \sim F(u, m\mu, m(1-\mu))\right] \quad (*)$$

is a non-decreasing function of μ. For a random variable Z, we can write $E[f(Z)] = \int_0^\infty P(f(Z) \geq w)dw$ if $f(Z)$ is non-negative. Then, given $\mu' \geq \mu$ and fix w,

$$\begin{aligned}
& P\left[V_x\left(\frac{m\mu' + Y}{m+u}, m+u; t+1\right) \geq w \,|Y \sim F(u, m\mu', m(1-\mu'))\right] \\
\geq\ & P\left[V_x\left(\frac{m\mu + Y}{m+u}, m+u; t+1\right) \geq w \,|Y \sim F(u, m\mu', m(1-\mu'))\right] \\
\geq\ & P\left[V_x\left(\frac{m\mu + Y}{m+u}, m+u; t+1\right) \geq w \,|Y \sim F(u, m\mu, m(1-\mu))\right].
\end{aligned}$$

The first inequality is due to the fact that by induction $V_x\left(\frac{m\mu+Y}{m+u}, m+u; t+1\right)$ is non-decreasing in μ when fixing $m + u > 0$. For any w, if $V_x\left(\frac{m\mu'+Y}{m+u}, m+u; t+1\right) \geq w$, then $V_x\left(\frac{m\mu+Y}{m+u}, m+u; t+1\right) \geq w$ as well. We now justify the second inequality. Proposition 3.3 in Kattuman and Yang (2014) states that Beta(α_1, β_1) has first-order stochastic dominance over Beta(α_2, β_2) for any $\alpha_1 \geq \alpha_2$ and $\beta_2 \leq \beta_2$. In addition, Remark 6.6 in Kattuman and Yang (2014) states that if Beta(α_1, β_1) has first order stochastic dominance over Beta(β_1, β_2), then Beta-Binomial(n, α_1, β_1) has first-order stochastic dominance over Beta-Binomial(n, α_2, β_2). Since $Y \sim F(u, m\mu', m(1-\mu'))$ has first order stochastic dominance over $Y \sim F(u, m\mu, m(1-\mu))$ and $V_x(\cdot, m+u; t+1)$ is non-decreasing, then $V_x\left(\frac{m\mu+Y}{m+u}, m+u; t+1\right) | Y \sim F(u, m\mu', m(1-\mu'))$ also has first order stochastic dominance over $V_x\left(\frac{m\mu+Y}{m+u}, m+u; t+1\right) | Y \sim F(u, m\mu, m(1-\mu))$. This justifies the second inequality, thus we show that (*) is non-decreasing function in μ.

As the sum of non-decreasing functions is non-decreasing, $V_x(\mu, m; T)$ is then non-decreasing in μ. Lastly, $V_x(\mu, m) = \lim_{T\to\infty} V_x(\mu, m; T)$ is also non-decreasing in μ since each single period reward is bounded and the limit of a sequence of non-decreasing function is non-decreasing. □

Lemma 3.1.3. *For fixed $m \in \mathbb{R}_+$, $V_x(\mu, m)$ is convex in $\mu \in (0, 1)$.*

Proof. Similar to the proof of Lemma 3.1.2, to show $V_x(\mu, m; t)$ for all $0 \le t \le T$ is convex in μ in a T-finite-horizon problem, it is sufficient to show that

$$E\left[V_x\left(\frac{m\mu + Y}{m + u}, m + u; t\right) | Y \sim F(u, m\mu, m(1-\mu))\right] \quad (**)$$

is convex in μ, which is proved using a few definition below.

Shaked and Shanthikumar (2007) define a set of random variable, $\{X(\theta) : \theta \in \Theta\}$ to be stochastic increasing if $E[\phi(X(\theta))]$ is increasing for all increasing functions ϕ, and $\{X(\theta) : \theta \in \Theta\}$ to be stochastic increasing and convex if $\{X(\theta) : \theta \in \Theta\}$ is stochastic increasing and $E[\phi(X(\theta))]$ is increasing and convex in θ for all increasing convex functions ϕ. Their use of the term "increasing" corresponds to our use of the term "non-decreasing".

For each $u \in \mathbb{N}_{++}$, let $\{X(u, p) : p \in (0, 1)\}$ be a set of binomial random variables with u number of trails and p as the probability of success in each Bernoulli trail. So a binomial random variable $X(u, p)$ has mean up and variance $up(1 - p)$. For each $m \in \mathbb{N}_{++}$, we define $\{Z(m\mu, m(1 - \mu)) : \mu \in (0, 1)\}$ to be a set of Beta random variables with parameters $\alpha = m\mu$ and $\beta = m(1 - \mu)$. That is, a Beta random variable $Z(m\mu, m(1-\mu))$ has mean $\frac{\alpha}{\alpha+\beta} = \mu$ and variance $\frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)} = \frac{\mu(1-\mu)}{m+1}$. Lastly, we define $\{Y(u, m\mu, m(1 - \mu)) : \mu \in (0, 1)\}$ to be a set of Beta-Binomial random variables with known number of Bernoulli trials, u, and a unknown probability

of success following a Beta distribution with a pair of parameters, $\alpha = m\mu$ and $\beta = m(1-\mu)$.

By Theorem 8.A.17 in Shaked and Shanthikumar (2007), $\{Y(u, m\mu, m(1-\mu) : \mu \in (0,1)\}$ is also stochastic increasing and convex because both $\{X(u,p) : p \in (0,1)\}$ and $\{Z(m\mu, m(1-\mu) : \mu \in (0,1)\}$ are stochastically increasing and convex (pages 360-366 in Shaked and Shanthikumar (2007)).

At the termination step with $t = 0$, $V_x(\mu, m; t)$ is convex in μ for fixed m since the termination reward is 0. Our induction hypothesis is that $V_x(\mu, m; t')$ is convex in μ for all $t < t' < T$. Since $Y \sim F(u, m\mu, m(1-\mu))$ is stochastically increasing and convex as defined above and $V_x(\cdot, m+u; t+1)$ is convex, we can conclude that (**) is convex in $\mu \in (0,1)$ by the definition of a family of random variable being stochastically increasing and convex, defined in (Shaked and Shanthikumar, 2007). Additionally, maximum of convex functions is convex and sum of convex functions is also convex. Thus, $V_x(\mu, m; T)$, which is a composition of sum and maximum of convex functions, is convex in μ for fixed m. Lastly, taking $T \to \infty$, the value function $V(\mu, m)$ is convex in $\mu \in (0,1)$. □

The proof techniques used in Lemma 3.1.2 and Lemma 3.1.3 can also be applied to give alternative proofs of Lemma 2.2.2 and Lemma 2.2.3 in Chapter 2 because $Y \sim F(1, m\mu, m(1-\mu))$, as a special case of $Y \sim F(u, m\mu, m(1-\mu))$, is stochastically increasing and convex.

3.1.3 Mathematical Model when decisions are made before observing the number of papers available for forwarding

In the preliminary work (Zhao and Frazier, 2014a), we considered a mathematical model similar to the one just presented, but where we have to devide the maximum number of papers we are willing to forward before observing number of arriving documents, L_{nx}. This could happen in an information filtering system where there is limited time for the system to react to the user's arrival, requiring the decision to be made beforehand to shorten the delay in the user's experience.

To formalize this scenario, we first redefine U_{nx} to be the maximum number of items to forward to the user from category x at the n^{th} user visit, and then show $\min(L_{nx}, U_{nx})$ items from category x to the user. For computational convenience, we require $U_{nx} \leq M$, where $M < \infty$. Conditioning on the decision U_{nx}, θ_x, and observation L_{nx}, Y_{nx} is again binomial with a probability θ_x, that is,

$$Y_{nx}|\theta_x, U_{nx}, L_{nx} \sim \text{Binomial}\Big(\min(U_{nx}, L_{nx}), \theta_x\Big).$$

Assuming a unit forwarding cost, c, in each step we generate reward, $Y_{nx} - c \cdot \min(U_{nx}, L_{nx})$. Our objective then becomes to find an optimal policy $\pi \in \Pi$ that maximizes the total expected reward:

$$\sup_{\pi\in\Pi} \mathbb{E}^{\pi}\left[\sum_{n=1}^{N}\sum_{x=1}^{k}\Big(Y_{nx} - c \cdot \min(U_{nx}, L_{nx})\Big)\right], \tag{3.8}$$

where a policy π is a sequence of functions, $(\pi_1, \pi_2, ..., \pi_N)$, where each $\pi_n : \Big(\{0, 1, ..., M\}^k \times \mathbb{N}^k \times \{0, 1, ..., M\}^k\Big)^{n-1} \mapsto \{0, 1, ..., M\}^k$ maps history, $\{U_{\ell x}, L_{\ell x}, Y_{\ell x} : \ell \leq n-1, \times \in \{1, ..., k\}\}$ into actions and Π is the set of all such policies.

Algorithm 3 Computation of $V_x^L(\alpha,\beta;\tilde{T})$ and $V_x^U(\alpha,\beta;\tilde{T})$

Require: γ, α_{0x}, β_{0x}, ξ_x, c, and N
  for $i = 0, ..., \tilde{T} + M$ **do**
    for $j = \max\{0, \tilde{T} + 1 - \alpha\}, ..., \tilde{T} + M - \alpha$ **do**
      Let $\alpha = \alpha_{0x} + i$, and $\beta = \beta_{0x} + j$.
      Let $V_x^L(\alpha,\beta;\tilde{T}) = \frac{E[\min(M,L_{1x})]}{(1-\gamma)(1-\gamma\xi_x)} \max\left\{0, \frac{\alpha}{\alpha+\beta} - c\right\}$ and $V^U(\alpha,\beta;\tilde{T}) = \frac{E[\min(M,L_{1x})]}{(1-\gamma)(1-\gamma\xi_x)}$.
    end for
  end for
  for $i = N_{ttl}, ..., 0$ **do**
    for $j = N_{ttl} - \alpha, ..., 0$ **do**
      Let $\alpha = \alpha_{0x} + i$, $\beta = \beta_{0x} + j$, and $\mu = \frac{\alpha}{\alpha+\beta}$.
      Let $V_x^L(\alpha,\beta;\tilde{T}) = \max_{0\le u\le M}\left\{(\mu - c)E(\min(u, L_{1x})] + \gamma E[V_x^L(\alpha_{1x},\beta_{1x};\tilde{T})|U_{1x} = u]\right\}$,
      $V_x^U(\alpha,\beta;\tilde{T}) = \max_{0\le u\le M}\left\{(\mu - c)E(\min(u, L_{1x})] + \gamma E[V_x^U(\alpha_{1x},\beta_{1x};\tilde{T})|U_{1x} = u]\right\}$,
    end for
  end for

The solution for this scenario is similar to the solution shown in Section 3.1.1. First, due to the independence assumption across θ_x, the problem expressed in (3.8) can also be decomposed into k subproblems, as the one in Section 3.1, but now each subproblem's state contains only the posterior (α,β) and is two-dimensional instead of three-dimensional. Next, we convert this problem from a random finite horizon problem into an infinite time horizon problem with a discount factor $\gamma \in (0, 1)$. A backward induction method for solving each subproblem is described in Algorithm 3.

The advantage of this model is that decisions only depend on (α,β), and not explicitly on the actual size of arriving items at each step. Thus, it provides storage advantage compared to the first model in Section 3.1 since decisions depend only on a two-dimensional quantity rather than a three-dimensional one. However, it is common to observe L in systems, and we may wish to make optimal forwarding decisions based on all available information, (α,β,L). In the rest of the chapter, we will switch back to the model where we first observe the

size of arriving items before we make decisions, U_{nx}.

3.2 Mathematical Model with a Constraint on Items Forwarded

In Section 3.1, we assumed that the system knows the unit cost, c, that the user incurs for reviewing each item. In reality, we often do not know this cost. In this section, we instead assume that the number of items forwarded in each step is constrained, $\sum_{x=1}^{k} U_{nx} \leq M \; \forall n$. Our objective is to maximize the expected number of relevant items forwarded, subject to this constraint:

$$\sup_{\tilde{\pi} \in \tilde{\Pi}} E^{\tilde{\pi}} \left[\sum_{n=1}^{N} \sum_{x=1}^{k} Y_{nx} \right], \tag{3.9}$$

where $\tilde{\Pi} = \{\pi \in \Pi : \sum_{x=1}^{k} U_{nx} \leq M \; \forall n\}$ and $\tilde{\pi}$ is a policy in $\tilde{\Pi}$ that satisfies the constraint. Similar to the previous section, we claim that

$$\frac{1}{\gamma} \sup_{\tilde{\pi} \in \tilde{\Pi}} E^{\tilde{\pi}} \left[\sum_{n=1}^{N} \sum_{x=1}^{k} Y_{nx} \right] = \sup_{\tilde{\pi} \in \tilde{\Pi}} E^{\tilde{\pi}} \left[\sum_{n=1}^{\infty} \gamma^{n-1} \sum_{x=1}^{k} Y_{nx} \right]. \tag{3.10}$$

In contrast with the previous problem in Section 3.1, computation in this problem scales exponentially in k due to the "curse of dimensionality", because we can no longer decompose equation (3.9) into multiple tractable sub-problems (Powell, 2007). Instead, we consider a Lagrangian relaxation, following developments Hu et al. (2014); Xie and Frazier (2013b), that provides a computationally tractable upper bound on the value of equation (3.10), and which motivates an index-based heuristic policy below in Section 3.2.1.

Let $\boldsymbol{\alpha}_0 = (\alpha_{01}, ..., \alpha_{0k})$ and $\boldsymbol{\beta}_0 = (\beta_{01}, ..., \beta_{0k})$. Let $\boldsymbol{\nu} = (\nu_1, \nu_2, ...)$ be a vector of Lagrange multipliers, with each $\nu_n \geq 0$ denoting a unit cost (or penalty) when we violate the constraint, $\sum_{x=1}^{k} U_{nx} \leq M$, at step n. We can then write the Lagrangian

relaxation of (3.10) as

$$\begin{aligned} V^{\boldsymbol{\nu}}(\boldsymbol{\alpha}_0, \boldsymbol{\beta}_0) &= \sup_{\pi \in \Pi} E^{\pi}\left[\sum_{n=1}^{\infty} \gamma^{n-1}\left[\sum_{x=1}^{k} Y_{nx} - \nu_n\left(\sum_{x=1}^{k} U_{nx} - M\right)\right]\right] \\ &= \sum_{x=1}^{k} \sup_{\pi(x) \in \Pi(x)} E^{\pi(x)}\left[\sum_{n=1}^{\infty} \gamma^{n-1}(Y_{nx} - \nu_n U_{nx})\right] + ME\left[\sum_{n=1}^{\infty} \gamma^{n-1}\nu_n\right]. \end{aligned} \tag{3.11}$$

The following Lemma shows that the value in equation (3.11) provides an upper bound on the value of equation (3.10).

Lemma 3.2.1. $V^{\boldsymbol{\nu}}(\boldsymbol{\alpha}_0, \boldsymbol{\beta}_0) \geq \sup_{\tilde{\pi} \in \tilde{\Pi}} E^{\tilde{\pi}}\left[\sum_{n=1}^{\infty} \gamma^{n-1} \sum_{x=1}^{k} Y_{nx}\right]$,

Proof.

$$\begin{aligned} V^{\boldsymbol{\nu}}(\boldsymbol{\alpha}_0, \boldsymbol{\beta}_0) &= \sup_{\pi \in \Pi} E^{\pi}\left[\sum_{n=1}^{\infty} \gamma^{n-1}\left[\sum_{x=1}^{k} Y_{nx} - \nu_n\left(\sum_{x=1}^{k} U_{nx} - M\right)\right]\right] \\ &\geq \sup_{\tilde{\pi} \in \tilde{\Pi}} E^{\tilde{\pi}}\left[\sum_{n=1}^{\infty} \gamma^{n-1}\left[\sum_{x=1}^{k} Y_{nx} - \nu_n\left(\sum_{x=1}^{k} U_{nx} - M\right)\right]\right] \geq \sup_{\tilde{\pi} \in \tilde{\Pi}} E^{\tilde{\pi}}\left[\sum_{n=1}^{\infty} \gamma^{n-1} \sum_{x=1}^{k} Y_{nx}\right] \end{aligned}$$

The first inequality holds because a larger optimal value is obtained when taking the supremum over a larger policy space, and $\tilde{\Pi} = \{\pi \in \Pi : \sum_{x=1}^{k} U_{nx} \leq M \ \forall n\} \subseteq \Pi$. Lastly, because of the constraint $\sum_{x=1}^{k} U_{nx} \leq M$ for all n is satisfied by $\tilde{\pi} \in \tilde{\Pi}$, and the non-negativity of the Lagrange multiplier, $\nu_n \geq 0$, $\nu_n \mathbb{E}^{\pi}\left[\sum_{x=1}^{k} U_{nx} - M\right] \geq 0$, and the last inequality holds. □

Given $\boldsymbol{\nu} \geq 0$, let $V_x^{\boldsymbol{\nu}}(\alpha_{0x}, \beta_{0x}) = \sup_{\pi(x) \in \Pi(x)} E^{\pi}\left[\sum_{n=1}^{\infty} \gamma^{n-1}(Y_{nx} - \nu_n U_{nx})\right]$. If we consider a special case $\boldsymbol{\nu} = \nu \boldsymbol{e}$ with $\nu \geq 0$ and $\boldsymbol{e} = (1, 1, ...)$, then $V_x^{\nu \boldsymbol{e}}(\alpha_{0x}, \beta_{0x})$ recovers the total expected reward in the information filtering problem described in Section 3.1, with a fixed unit cost, ν, for forwarding each item. $V_x^{\nu \boldsymbol{e}}(\alpha, \beta)$ can be computed efficiently using Algorithm 2.

Following equation (3.11), $V^{\boldsymbol{\nu}}(\boldsymbol{\alpha}_0, \boldsymbol{\beta}_0)$ can be decomposed into a sum of mul-

tiple sub-problems, $V_x^{\nu}(\alpha_{0x}, \beta_{0x})$, plus a constant,

$$V^{\nu}(\boldsymbol{\alpha}_0, \boldsymbol{\beta}_0) = \sum_{x=1}^{k} V_x^{\nu}(\alpha_{0x}, \beta_{0x}) + ME\left[\sum_{n=1}^{\infty} \gamma^{n-1} \nu_n\right] \geq \sup_{\tilde{\pi} \in \tilde{\Pi}} E^{\tilde{\pi}}\left[\sum_{n=1}^{\infty} \gamma^{n-1} \sum_{x=1}^{k} Y_{nx}\right] \quad (3.12)$$

This upper bound $V^{\nu}(\boldsymbol{\alpha}_0, \boldsymbol{\beta}_0)$ is useful because it can be computed efficiently, as the sum of independent and easy-to-solve stochastic control sub-problems. We can construct a tighter upper bound by taking the infimum of $V^{\nu}(\boldsymbol{\alpha}_0, \boldsymbol{\beta}_0)$ over sets of potential values for our Lagrange multipliers,

$$\mathrm{UB}(\boldsymbol{\alpha}_0, \boldsymbol{\beta}_0) = \inf_{\boldsymbol{\nu} \geq 0} V^{\nu}(\boldsymbol{\alpha}_0, \boldsymbol{\beta}_0) \geq \sup_{\tilde{\pi} \in \tilde{\Pi}} E^{\tilde{\pi}}\left[\sum_{n=1}^{\infty} \gamma^{n-1} \sum_{x=1}^{k} Y_{nx}\right]. \quad (3.13)$$

Given any $\boldsymbol{\nu} \geq \mathbf{0}$, Lemma 3.2.1 states that $V^{\nu}(\boldsymbol{\alpha}_0, \boldsymbol{\beta}_0)$ is an upper bound on equation (3.10). Then, the infimum of $V^{\nu}(\boldsymbol{\alpha}_0, \boldsymbol{\beta}_0)$ over the space of $\boldsymbol{\nu} \geq \mathbf{0}$ is still an upper bound on equation (3.10), thus (3.13) holds.

Lemma 3.2.2. *Given any $\alpha > 0, \beta > 0$, $V_x^{\nu}(\alpha, \beta)$ is non-increasing and convex in $\boldsymbol{\nu}$.*

Proof.

$$V_x^{\nu}(\alpha, \beta) = \sup_{\pi(x) \in \Pi(x)} E^{\pi(x)}\left[\sum_{n=1}^{\infty} \gamma^{n-1}(Y_{nx} - \nu_n U_{nx})\right] \quad (3.14)$$

$$= \sup_{\pi(x) \in \Pi(x)} E^{\pi(x)}\left[\sum_{n=1}^{\infty} \gamma^{n-1} Y_{nx}\right] - \sum_{n=1}^{\infty} \nu_n \gamma^{n-1} E^{\pi(x)}[U_{nx}] \quad (3.15)$$

Let $a(\pi(x)) = E^{\pi(x)}\left[\sum_{n=1}^{\infty} \gamma^{n-1} Y_{nx}\right]$ and $b_n(\pi(x)) = -\gamma^{n-1} E^{\pi(x)}[U_{nx}]$. Then

$$V_x^{\nu}(\alpha, \beta) = \sup_{\pi(x) \in \Pi(x)} a(\pi(x)) + \sum_{n=1}^{\infty} \nu_n b_n(\pi(x)). \quad (3.16)$$

The policy space $\Pi(x)$ does not depend on $\boldsymbol{\nu}$, nor does $a(\pi(x))$ or $b_n(\pi(x))$. Here, $a(\pi(x)) + \sum_{n=1}^{\infty} \nu_n b_n(\pi(x))$ is an affine function of $\boldsymbol{\nu}$ for each policy $\pi(x)$ and $V_x^{\nu}(\alpha, \beta)$ is the supremum over a collection of these affine functions. By equation (3.16), since decisions $U_{nx} \geq 0$ and $b_n(\pi(x)) = -\gamma^{n-1} E^{\pi(x)}[U_{nx}] \leq 0$ for all n, we can also conclude that $V_x^{\nu}(\alpha, \beta)$ is non-increasing in $\boldsymbol{\nu}$.

Next, to show that the value function is convex in $\boldsymbol{\nu}$, it is sufficient to show that $V_x^{\frac{\nu+\nu'}{2}}(\alpha,\beta) \leq \frac{1}{2}V^{\nu}(\alpha,\beta) + \frac{1}{2}V^{\nu'}(\alpha,\beta)$ given any $\boldsymbol{\nu}, \boldsymbol{\nu}' \geq \mathbf{0}$ with $\nu_n, \nu_n' < \infty$ for all n. The value function is bounded for any $\boldsymbol{\nu}, \geq \mathbf{0}$ with $\nu_n < \infty$ for all n. Then

$$\begin{aligned}
V_x^{\frac{\nu+\nu'}{2}}(\alpha,\beta) &= \sup_{\pi(x)\in\Pi(x)}\left[a(\pi(x)) + \sum_{n=1}^{\infty}\frac{\nu_n+\nu_n'}{2}b_n(\pi(x))\right] \\
&= \sup_{\pi(x)\in\Pi(x)}\left[\frac{1}{2}a(\pi(x)) + \sum_{n=1}^{\infty}\frac{\nu_n}{2}b_n(\pi(x)) + \frac{1}{2}a(\pi(x)) + \sum_{n=1}^{\infty}\frac{\nu_n'}{2}b_n(\pi(x))\right] \\
&\leq \sup_{\pi(x)\in\Pi(x)}\left[\frac{1}{2}a(\pi(x)) + \sum_{n=1}^{\infty}\frac{\nu_n}{2}b_n(\pi(x))\right] + \sup_{\pi(x)\in\Pi(x)}\left[\frac{1}{2}a(\pi(x)) + \sum_{n=1}^{\infty}\frac{\nu_n}{2}b_n(\pi(x))\right] \\
&= \frac{1}{2}V_x^{\nu}(\alpha,\beta) + \frac{1}{2}V_x^{\nu'}(\alpha,\beta).\square
\end{aligned}$$

Lemma 3.2.3. *$V^{\nu}(\boldsymbol{\alpha}_0, \boldsymbol{\beta}_0)$ is convex in $\boldsymbol{\nu}$.*

Proof. For each x, $V_x^{\nu}(\alpha_{0x}, \beta_{0x})$ is convex in $\boldsymbol{\nu}$ as shown in Lemma 3.2.2. The second term, $ME[\sum_{n=1}^{\infty}\gamma^{n-1}\nu_n]$ is also convex in ν. Therefore, $V^{\nu}(\boldsymbol{\alpha},\boldsymbol{\beta})$, which is sum of convex functions, is also convex in $\boldsymbol{\nu}$. □

Lemma 3.2.4. *Given $\boldsymbol{\nu}$, let $\pi^{*,\nu}(x)$ be an optimal policy for $V_x^{\nu}(\alpha,\beta)$. Let $\boldsymbol{g}^{\pi^{*,\nu}(x)} = \left(-\mathbb{E}^{\pi^{*,\nu}(x)}\left[\gamma^{n-1}U_{nx}\right] : n = 1, 2, ...\right)$. Then $\boldsymbol{g}^{\pi^{*,\nu}(x)}$ is a subgradient of $\nu \mapsto V_x^{\nu}(\alpha,\beta)$ at ν.*

Proof. Pick any $\boldsymbol{\nu}' \neq \boldsymbol{\nu}$, we have

$$\begin{aligned}
V_x^{\nu}(\alpha,\beta) + (\boldsymbol{\nu}' - \boldsymbol{\nu})^T\boldsymbol{g}^{\pi^{*,\nu}(x)} &= E^{\pi^{*,\nu}(x)}\left[\sum_{n=1}^{\infty}\gamma^{n-1}(Y_{nx} - \nu_n U_{nx})\right] - \sum_{n=1}^{\infty}(\nu_n' - \nu_n)E^{\pi^{*,\nu}(x)}[\gamma^{n-1}U_{nx}] \\
&= E^{\pi^{*,\nu}(x)}\left[\sum_{n=1}^{\infty}\gamma^{n-1}(Y_{nx} - \nu_n' U_{nx})\right] \\
&\leq \sup_{\pi(x)\in\Pi(x)} E^{\pi(x)}\left[\sum_{n=1}^{\infty}\gamma^{n-1}(Y_{nx} - \nu_n' U_{nx})\right] = V_x^{\nu'}(\alpha,\beta).
\end{aligned}$$

The first equality expands the definitions of the value function $V_x^{\nu}(\alpha,\beta)$ and the corresponding subgradient $\boldsymbol{g}^{\pi^{*,\nu}(x)}$ given that $\pi^{*,\nu}(x)$ is an optimal policy for

$V_x^{\boldsymbol{\nu}}(\alpha,\beta)$ at $\boldsymbol{\nu}$. We then derive the second equality after cancelling out the term $\boldsymbol{\nu}^T \boldsymbol{g}^{\pi^{*,\boldsymbol{\nu}}(x)} = -\sum_{n=1}^{\infty} \nu_n E^{\pi^{*,\boldsymbol{\nu}}(x)}[\gamma^{n-1} U_{nx}]$. The last inequality holds because the expected total reward, $E^{\pi^{*,\boldsymbol{\nu}}(x)}\left[\sum_{n=1}^{\infty} \gamma^{n-1}(Y_{nx} - \boldsymbol{\nu}_n' U_{nx})\right]$, at $\boldsymbol{\nu}'$ under an arbitrary policy $\pi^{*,\boldsymbol{\nu}}(x)$ is smaller than or equal to the supremum of the expected total reward under all possible policies, $\sup_{\pi(x)\in\Pi(x)} E^{\pi(x)}\left[\sum_{n=1}^{\infty} \gamma^{n-1}(Y_{nx} - \boldsymbol{\nu}_n' U_{nx})\right] = V_x^{\boldsymbol{\nu}'}(\alpha,\beta)$. □

We can compute this subgradient using the Markov Property. Before writing down equations, we first introduce the notation of Λ_{nx} as the feasible state space where (α_{nx},β_{nx}) takes values at the nth step when we start at the prior beta parameters with (α_{0x},β_{0x}). The subgradient at the nth entry is computed as follows

$$-\mathbb{E}^{\pi^{*,\boldsymbol{\nu}}(x)}\left[\gamma^n U_{n+1,x}\right] = -\gamma^n \sum_{(\alpha',\beta')\in\Lambda_{n,x}} P_n^{\pi^{*,\boldsymbol{\nu}}(x)}(\alpha',\beta') \sum_{\ell=0}^{\bar{L}} P(L_{n+1,x}=\ell)\cdot U_{n+1,x}^{*,\pi^{*,\boldsymbol{\nu}}(x)}(\alpha',\beta',\ell),$$

where $P_n^{\pi_x^*(\nu)}(\alpha',\beta')$ is defined recursively by

$$P_n^{\pi_x^*(\nu)}(\alpha',\beta') = \begin{cases} \mathbb{1}_{[\alpha_0=\alpha,\beta_0=\beta]} & \text{if } n=0, \\ \sum_{(\alpha,\beta)\in\Lambda_{n-1,x}} P(\alpha_{nx}=\alpha',\beta_{nx}=\beta'|\alpha_{n-1,x}=\alpha,\beta_{n-1,x}=\beta) P_{n-1}^{\pi_x^{*,\boldsymbol{\nu}}(x)}(\alpha,\beta) & \text{if } n\geq 1. \end{cases}$$

and the transition probability from the state $(\alpha_{n-1,x},\beta_{n-1,x})$ at the $(n-1)$th step to the state (α_{nx},β_{nx}) at the nth step is defined as

$$\begin{aligned} & P(\alpha_{nx}=\alpha',\beta_{nx}=\beta'|\alpha_{n-1,x}=\alpha,\beta_{n-1,x}=\beta) \\ = \;& P(U_{nx}^*=\alpha'+\beta'-\alpha-\beta, Y_{nx}=\alpha'-\alpha\,|\alpha_{n-1,x}=\alpha,\beta_{n-1,x}=\beta) \\ = \;& \sum_{\ell=0}^{\bar{L}} P(L_{nx}=\ell) P(U_{nx}^*=\alpha'+\beta'-\alpha-\beta, Y_{nx}=\alpha'-\alpha\,|\alpha_{nx}=\alpha,\beta_{nx}=\beta,L_{nx}=\ell) \\ = \;& \sum_{\ell=0}^{\bar{L}} P(L_{nx}=\ell)\cdot \mathbb{1}_{[U_{nx}^*=\alpha'+\beta'-\alpha-\beta\,|\alpha_{nx}=\alpha,\beta_{nx}=\beta,L_{nx}=\ell]} \\ & \quad \cdot P(Y_{nx}=\alpha'-\alpha|\alpha_{n-1,x}=\alpha,\beta_{n-1,x}=\beta,U_{nx}^*=\alpha'+\beta'-\alpha-\beta). \end{aligned}$$

Let us consider the set $\{\boldsymbol{\nu} : \boldsymbol{\nu} = \nu\boldsymbol{e} \text{ with } \nu \geq 0, \boldsymbol{e} = (1,1,1,...)\}$. Given any stepwise penalty $\nu \geq 0$, let $\pi^{*,\nu}(x)$ be the optimal policy for $V_x^{\nu}(\cdot,\cdot)$, and

$U^{*,\nu}(\alpha,\beta,\ell) \in \pi^{*,\nu}(x)$ be the optimal decision at state (α,β) conditioned on $L_{nx} = \ell$. For each $0 \le u \le \ell$, we define

$$\nu^*(u;\alpha,\beta,\ell) = \sup_{\nu\ge 0}\{\nu : U^{*,\nu}(\alpha,\beta,\ell) \ge u \text{ with } U^{*,\nu}(\alpha,\beta,\ell) \in \pi^{*,\nu}(x)\} \quad (3.17)$$

to be the largest reward achieved for forwarding (or the largest cost a user would pay to view) at least u items at state (α,β) conditioned on $L_{nx} = \ell$.

Based on the empirical observation of $\nu^*(u;\alpha,\beta,\ell)$ in various value functions, we conjecture that $\nu^*(u;\alpha,\beta,\ell)$ has this structural property: given (α,β,ℓ) and $u \le \ell$, $\nu^*(u;\alpha,\beta,\ell) = \nu^*(u;\alpha,\beta,\ell')$ for any $\ell' \ne \ell$. If this conjecture is true, then we only need to compute and store $\nu^*(u;\alpha,\beta,\bar{L})$ for all $0 \le u \le \bar{L}$, reducing the computational complexity from $O(\bar{L}^2)$ to $O(\bar{L})$.

Conjecture 3.2.1. *Given (α,β,ℓ) and $u \le \ell$, $\nu^*(u;\alpha,\beta,\ell) = \nu^*(u;\alpha,\beta,\ell')$ for any $\ell' \ne \ell$.*

We sketch a partial proof for the conjecture. Given ℓ, let us define the set $A(u,\alpha,\beta,\ell) = \{\nu : U^{*,\nu}(\alpha,\beta,\ell) \ge u \text{ with } U^{*,\nu}(\alpha,\beta,\ell) \in \pi^{*,\nu}(x)\}$. For any $\nu \in A(u,\alpha,\beta,\ell)$, we have $U^{*,\nu}(\alpha,\beta,\ell) \ge u$ by definition. By Lemma 3.1.1,

$$V_x^\nu(\alpha,\beta|\ell+1) = \max\{V_x^\nu(\alpha,\beta|\ell), Q(\alpha,\beta,\ell+1)\} \ge V_x^\nu(\alpha,\beta|\ell),$$

we have either $U^{*,\nu}(\alpha,\beta,\ell+1) = U^{*,\nu}(\alpha,\beta,\ell) \ge u$ or $U^{*,\nu}(\alpha,\beta,\ell+1) = \ell+1 \ge u$. Thus, $\nu \in A(u,\alpha,\beta,\ell+1)$ and we show that $A(u,\alpha,\beta,\ell) \subseteq A(u,\alpha,\beta,\ell+1)$. Let us now look at the other direction. Given $\nu \in A(u,\alpha,\beta,\ell+1)$, either $U^{*,\nu}(\alpha,\beta,\ell+1) = U^{*,\nu}(\alpha,\beta,\ell)$ holds or $U^{*,\nu}(\alpha,\beta,\ell) = \ell$ is true. In the first case, if $U^{*,\nu}(\alpha,\beta,\ell+1) = U^{*,\nu}(\alpha,\beta,\ell)$, then $U^{*,\nu}(\alpha,\beta,\ell) \ge u$. Then, to complete the conjecture, it is sufficient to show the later case that if $U^{*,\nu}(\alpha,\beta,\ell+1) = \ell+1$, then $U^{*,\nu}(\alpha,\beta,\ell) = \ell$.

Regardless whether this conjecture holds, we define $\nu^*(u,\alpha,\beta) := \nu^*(u;\alpha,\beta,\bar{L})$ for all $0 \le u \le \bar{L}$. Given $\bar{L}$ available items, $\nu^*(u,\alpha,\beta)$ represents the largest cost

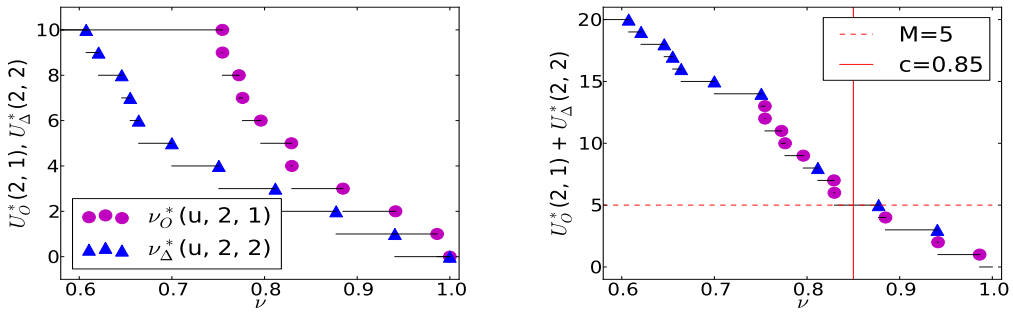
one would pay to view at least u items at state (α, β). This interpretation of costs associates the user's utility to viewing the next additional item, and it inspires a heuristic index-based policy for ranking available items as we illustrate in Section 3.2.1.

3.2.1 MDP-based Information Filtering (MDP-IF) Policy

In this section, we propose a heuristic index-based policy for ranking items queued at step n, disregarding whether the user cost is known. We call our policy the MDP-based Information Filtering Policy, abbreviated *MDP-IF*. We then discuss scenarios where the proposed policy is Bayes-optimal.

At each step n, we compute $\nu^*(u, \alpha_{nx}, \beta_{nx})$ for all possible $u \leq M$ for each category x, then rank items from the categories based on the computed $\nu_x^*(u, \alpha_{nx}, \beta_{nx})$, among all categories and all $u \leq M$. Recall that $\nu^*(u, \alpha_{nx}, \beta_{nx})$ defined in Section 3.2 reflects the highest reward one can achieve if we choose to forward at least u items from the category x at the current state, $(\alpha_{nx}, \beta_{nx})$. In the situation where we have a constraint that $\sum_{x=1}^{k} U_{nx} \leq M$, we would forward items from the ranked list until we exhaust M slots. This heuristic index policy is exactly the optimal policy corresponding to the Gittins index in a conventional multi-armed bandit problem (Gittins, 1979; Whittle, 1980), when there is at least one item queued in each category per step and one is allowed to forward one item per step, $M = 1$. On the other hand, if a user cost c is specified, then we would forward items from the ranked list with $\nu^*(u, \alpha_{nx}, \beta_{nx}) \geq c$.

Figure 3.2 demonstrates how this index policy works in a two-category problem (category "O" and category "Δ"). Category "O" is at state $(\alpha_O, \beta_O) = (2, 1)$



(a) Plot of $U_O^*(2,1)$ and $U_\Delta^*(2,2)$ against ν. (b) Plot of $U_O^*(2,1) + U_\Delta^*(2,2)$ against ν.

Figure 3.2: There are two categories: category O at state (2, 1) and category Δ at state (2, 2). Figure (a) shows optimal decision $U_O^*(2,1)$, $U_\Delta^*(2,2)$ against Lagrange multiplier ν, and also plots $\nu_O^*(u,2,1)$ (denoted in purple circles) and $\nu_\Delta^*(u,2,2)$ (denoted in blue triangles). Figure (b) plots $U_O^*(2,1) + U_\Delta^*(2,2)$ vs. ν, with ranked list of $\nu_O^*(u,2,1)$ and $\nu_\Delta^*(u,2,2)$ among all $u \in \{0,1,...,10\}$. When $M = 5$, the rank list would be: $\{O,O,\Delta,O,\Delta\}$. When the user cost is 0.85, the rank list would be the five right-most items, $\{O,O,\Delta,O,\Delta\}$.

and category "Δ" is at state $(\alpha_\Delta,\beta_\Delta) = (2,2)$. Figure (a) plots optimal decisions $U_O^*(2,1)$ and $U_\Delta^*(2,2)$ as a function of ν conditioned on that the number of available items is $L_{1x} = 10$. Additionally, Figure 3.2 also identifies and plots $\nu_O^*(u,2,1)$ and $\nu_\Delta^*(u,2,2)$ values for all $u \in \{1,...,10\}$. Then, Figure (b) plots $U_O^*(2,1)+U_\Delta^*(u,2,2)$ against ν, with a ranked list of $\nu_O^*(u,2,1)$ and $\nu_\Delta^*(u,2,2)$ among all $u \in \{1,...,10\}$. When $M = 5$, the ranked list would be items below the dashed line: $\{O,O,\Delta,O,\Delta\}$. When the user cost is 0.85, the ranked list would be the five items on the right of the vertical line (defined by $c = 0.85$): $\{O,O,\Delta,O,\Delta\}$.

3.3 Mathematical Model with a random stepwise Constraint on Items Forwarded

In many information filtering systems, including the one we are building for arXiv.org, we do not know aprior the number of items that a user wants to see at each visit. Moreover, this number is often random rather than a fixed value, M, defined in Section 3.2. When scrolling down a list of recommended items, the user's attention on the list often decays (e.g., attention-decay with a probability $1 - \kappa$) and then abandon the system from this visit. Thus, in these systems, we would rather present a ranking of items based on item relevance. In this way, the user browses pages from top to bottom and sees relevant information before s/he distract from the system (Dupret and Piwowarski, 2008). Below, we demonstrate how to extend Section 3.2 to a variant in which the user does not view all forwarded item, but with some decayed attention on the list.

We assume that the user's interest on viewing another item from the forwarded list decays with probability $1 - \kappa$. Let $M_n \in \{0, 1, ...\}$ denote the number of items viewed by a user, which is unknown until the user abandon the page during each visit. M_n follows a geometric distribution with parameter κ and mean of $(1 - \kappa)/\kappa$ items. Both the item-arrival model and the user-arrival model stay the same as one defined in Section 3.1. Recall that L_{nx} denotes the number of items arrives in category x during the user's nth visit. Let $L_n = \sum_{x=1}^{k} L_{nx}$ be the total number of available items to be presented to the user at his/her nth visit.

For all $1 \leq i \leq L_n$, let $X_{ni} = x \in \{1, ..., k\}$ denote a ranking decision for the available items in the queue, satisfying constraints that $\sum_{i=1}^{L_n} \mathbb{1}_{[X_{ni}=x]} = L_{nx}$ for every category $x \in \{1, ..., k\}$ (since we can not exceed the number of available items

in each category). With the ranking, X_{ni}, the user provides explicit feedback, Y_{nx}, counting the number of relevant items that users views. Conditioning on $\{X_{ni}\}_{1\leq i\leq L_n}$, M_n, and θ_x, Y_{nx} is binomial,

$$Y_{nx}|\theta_x, M_n, \{X_{ni}\}_{1\leq i\leq L_n} \sim \text{Binomial}\left(\sum_{i=1}^{M_n} \mathbb{1}_{[X_{ni}=x]},\ \theta_x\right).$$

The objective is still to maximize the expected number of relevant items viewed by the user, subject to constraints:

$$\sup_{\bar{\pi}\in\bar{\Pi}} E^{\bar{\pi}}\left[\sum_{n=1}^{N}\sum_{x=1}^{k} Y_{nx}\right], \tag{3.18}$$

where $\bar{\pi}$ is a sequence of functions, $(\bar{\pi}_1, \bar{\pi}_2, ...)$ with each $\bar{\pi}$ mapping historical observation, $\{(L_{\ell x}, X_{\ell i}, Y_{\ell x}, M_n) : \forall \ell \leq n-1, \forall i \leq L_\ell\}$, and new observations of item arrivals, L_{nx} for all x, into a ranking list of items. $\bar{\Pi}$ is the set of all such ranking policies that satisfies a stepwise constraint that $\sum_{i=1}^{L_n} \mathbb{1}_{[X_{ni}=x]} = L_{nx}$ for all x.

This stochastic dynamic problem is intractable. Instead, we can consider a relaxation of the problem, in which the policy gets extra information, M_n, before making the ranking decision. Similarly to the technique used in Section 3.2, we can then show that the value of this relaxed problem provides an upper bound on the value of the ranking problem in (3.18). Moreover, the MDP-IF policy proposed in Section 3.2.1 can be used to provide a heuristic index-based policy to rank all available items in the queue.

In the rest of the section, we assume that we observe extra information about the user's budget, M_n, before making decisions at the user's nth visit. We also assume that M_n and L_{nx} are independent. With that, our decision simply involves determining the number of available to forward from each category at each user visit. Conditioned on L_{nx} and M_n, let $U_{nx} \in \{0, ..., \min(L_{nx}, M_n)\}$ be decision variables that satisfy the conditions $\sum_{x=1}^{k} U_{nx} \leq M_n$ in each step n. This is

natural extension that adds flexibility into users' browsing behavior. Our objective is to maximize the expected number of relevant items forwarded, subject to constraints:

$$\sup_{\hat{\pi}\in\hat{\Pi}} E^{\hat{\pi}}\left[\sum_{n=1}^{N}\sum_{x=1}^{k} Y_{nx}\right], \tag{3.19}$$

where $\hat{\pi}$ is now a sequence of functions, $(\hat{\pi}_1, \hat{\pi}_2, ...)$, and each $\hat{\pi}_n$ maps historical observations, $\{(L_{\ell x}, M_\ell, U_{\ell x}, Y_{\ell x}) : \forall \ell \leq n-1\}$, with new observations $\{M_n, L_{nx} \forall x\}$ into actions with a stepwise constraint that $\sum_{x=1}^{k} U_{nx} \leq M_n$ for all n. The set of all such policies is denoted by $\hat{\Pi}$, whose relation with Π in Section 3.1 is $\hat{\Pi} = \{\pi \in \Pi : \sum_{x=1}^{k} U_{nx} \leq M_n \ \forall n\}$.

The problem is still intractable as a larger number of categories. As method used in Section 3.2, we can then consider a Lagrange relaxation of equation (3.19):

$$\begin{aligned} V^{\nu}(\boldsymbol{\alpha}_0, \boldsymbol{\beta}_0) &= \sup_{\pi\in\Pi} E^{\pi}\left[\sum_{n=1}^{N}\left[\sum_{x=1}^{k} Y_{nx} - \nu_n\left(\sum_{x=1}^{k} U_{nx} - M_n\right)\right]\right] && (3.20)\\ &= \sum_{x=1}^{k}\sup_{\pi\in\Pi} E^{\pi}\left[\sum_{n=1}^{N}(Y_{nx} - \nu_n U_{nx})\right] + E\left[\sum_{n=1}^{N}\nu_n M_n\right]. && (3.21) \end{aligned}$$

In this problem, U_{nx} takes values in the range $\{0, ..., \min(L_{nx}, M_n)\}$. Since L_{nx} and M_n are independent, the minimum of two independent geometric random variables is still geometric with parameter $1 - (1-\xi_x)(1-\kappa)$. This problem shares a similar structure as the relaxation problem studied in Section 3.2, but add flexibility to the user's browsing behavior. Lastly, the MDP-IF policy described in Section 3.2.1 can be applied in this problem to rank items among categories based on the computed indices, $\nu^*(u, \alpha, \beta)$, that reflects the user's cost to view at least u items from each category.

3.4 Experimental Results

In this section, we show numerical results of Monte Carlo simulations in different parameter settings to illustrate how this heuristic index-based policy performs compared to other competing methods, including the pure exploitation and upper confidence bound (UCB) policies, on the information filtering problem from Section 3.2, in which we consider periodic reviews, unknown user cost, and a fixed stepwise constraint on items forwarded. UCB is a popular heuristic exploration policy, related to the broader literature in Lai and Robbins (1985); Auer et al. (2002); Kaufmann et al. (2012); Agrawal and Goyal (2011). To our knowledge, we are not aware that UCB has been implemented in this problem but the idea is simple enough, so we implement it for comparison.

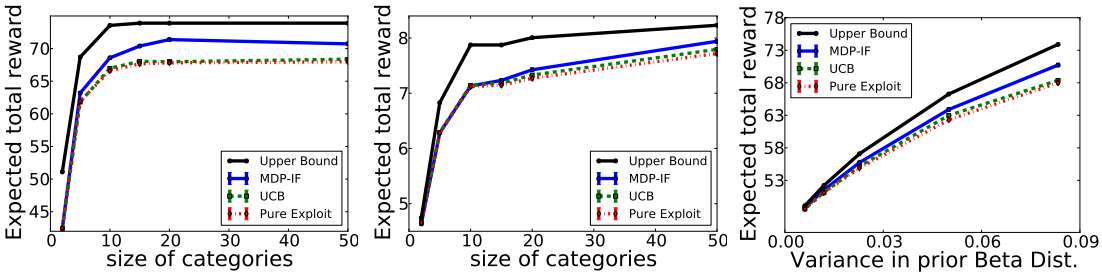
At each step n, pure exploitation ranks items based on the posterior means of user preference across categories, while UCB computes ρ-quantiles, $Q(\rho, \theta_x)$, for the posterior distributions, θ_x, of user preferences and ranks items by $Q(\rho, \theta_x)$. The parameter ρ is often tuned in each scenario to maximize its best performance. We also tried a different UCB approach, as one shown in the preliminary work (Zhao and Frazier, 2014a), that varies ρ_n at the nth step as a function of the total number of items shown. Since we did not observe its superiority over the general method described in the section, we drop implementing it in all scenarios.

We run experiments for five different scenarios, each with a chosen set of parameters, including γ and $\{\alpha_{0x}, \beta_{0x}, \xi_x\}_{x\in\{1,\ldots,k\}}$. Most browsers, as one we are building for arXiv.org, can fit only 7-10 items per page. So, in these experiments, we set $M = 10$ to constraint the maximum number of items allowed to forward

per user visit and let unit cost $c = 0.0$.

The first two scenarios in Figure 3.3 consist of simulations for a range of $k \in \{2, 5, 10, 15, 20, 50\}$, for us to understand how each policy behaves as category size, k, varies in two specified pairs of prior beta parameters, $(\alpha_{0x}, \beta_{0x})$. Specific parameters setting are described in sub-figure captions in Figure 3.3. $\gamma = 0.9$ is chosen based on the estimated parameter of the empirical distribution of user visits in astro-ph.GA and astro-ph.CO from the arXiv.org dataset in 2009-2010. Similarly, $\xi_x = 0.1$ is chosen based on the empirical distributions of item arrivals among categories. For convenience, we set $\alpha_{0x} = \beta_{0x} = 1$ in Figure 3.3(a) to have prior relevance probability of 0.5, while in Figure 3.3(b) we set $\alpha_{0x} = 1$ and $\beta_{0x} = 19$ to understand how the policy behaves as its prior relevance probability reduces to 0.05. Lastly, in Figure 3.3(c), we set the size of categories to be $k = 50$ and fix prior mean to be $\mu_{0x} = \frac{\alpha_{0x}}{\alpha_{0x}+\beta_{0x}} = \frac{1}{2}$, while we vary variance, $\text{Var}_{(\alpha,\beta)} = \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}$, of prior Beta parameters ($\alpha = \beta \in \{1, 2, 5, 10, 20\}$). This experiment helps to understand how the policy performs when we are more certain about user preferences.

With the chosen set of parameters in each scenario, we simulate a large set of users, with user visits, items arrivals between two visits, and user feedback on forwarded items generated according to the model formulated in Section 3.1. At each visit, each policy (MDP-IF, pure exploitation, or UCB) decides the top $M = 10$ items to forward. The reward is then collected to compute the expected total reward for each policy. UCB runs the simulations for various $\rho \in \{0.65, 0.7, 0.75, 0.8, 0.85, 0.9, 0.95\}$, with the best result reported in Figure 3.3. In addition to that, each sub-figure also plots an upper bound (in black solid



(a) $M = 10$, $c = 0$, $\gamma = .9$, $\alpha_{0x} = \beta_{0x} = 1, \xi_x = .1\ \forall x$

(b) $M = 10$, $c = 0$, $\gamma = .9$, $\alpha_{0x} = 1, \beta_{0x} = 19, \xi_x = .1\ \forall x$

(c) $M = 10$, $c = 0$, $\gamma = .9$, $k = 50, \xi_x = .1, \frac{\alpha_{0x}}{\beta_{0x}} = 1\ \forall x$

Figure 3.3: Each sub-figure shows plots of expected total reward of upper bounds (Upper Bound in black solid line), and expected total reward with 95% confidence intervals under each policy (MDP-IF in solid blue line, UCB in red dashed line, Pure Exploit in green dotted line) with a given parameter setting specified in its sub-caption. $100,000$ users are simulated in each scenario. UCB runs various simulations for $\rho \in \{0.65, 0.7, 0.75, 0.8, 0.85, 0.9, 0.95\}$, with the best results plotted. Figure (a) and (b) plot expected total reward against the size of categories, while we fix the prior mean $\frac{\alpha_{0x}}{\alpha_{0x}+\beta_{0x}} = \frac{1}{2}$ in Figure (c) and plot expected total reward against variances of various prior beta parameters, $\alpha_{0x} = \beta_{0x} \in \{1, 2, 5, 10, 20\}$, with the size of categories $k = 50$.

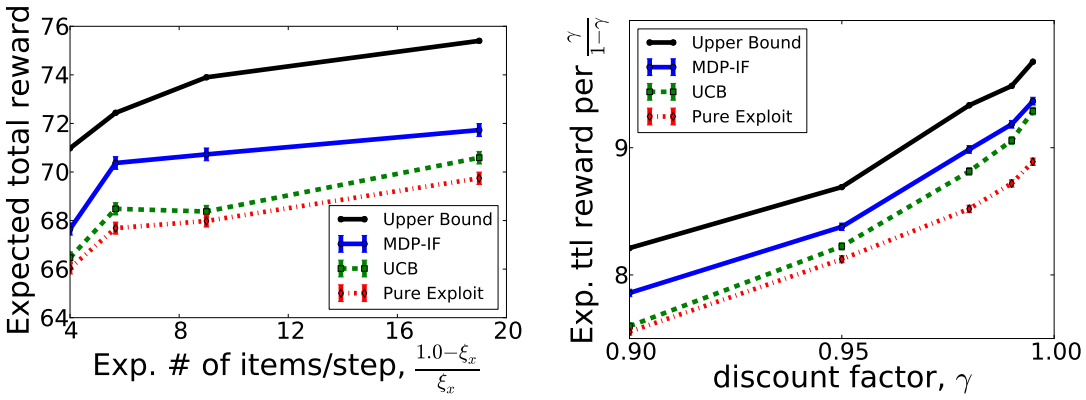
line) for each scenario described in the sub-figure. The upper bound,

$$\inf_{\{\nu \boldsymbol{e} \geq 0 : \boldsymbol{e} = (1,1,\ldots)\}} \gamma V^{\nu}(\boldsymbol{\alpha}_0, \boldsymbol{\beta}_0) = \inf_{\{\nu \boldsymbol{e} \geq 0 : \boldsymbol{e} = (1,1,\ldots)\}} \gamma \left\{ \sum_{x=1}^{k} V_x^{\nu \boldsymbol{e}}(\alpha_{0x}, \beta_{0x}) + ME\left[\sum_{n=1} \gamma^{n-1} \nu \right] \right\}, \quad (3.22)$$

is found using subgradient method described in Lemma 3.2.4. The factor γ in added because these simulation results compute the maximum expected total reward for random-finite horizon problems, as shown in equation (3.3), while $\inf_{\{\nu \boldsymbol{e} \geq 0 : \boldsymbol{e} = (1,1,\ldots)\}} V^{\nu}(\boldsymbol{\alpha}_0, \boldsymbol{\beta}_0)$ is computed based on the corresponding infinite horizon problems. Their ratio is γ. When providing the upper bound, we consider only the domain that the penalty (or cost) stays same over horizons, $\{\nu \boldsymbol{e} : \boldsymbol{e} = (1, 1, \ldots)$ and $\nu \geq 0\}$, rather than the whole set $\{\boldsymbol{\nu} \geq \mathbf{0}\}$ due to convenient computation in optimizing (3.22), but this upper bound still provides a reasonable tight bound on the value function, as shown in Figure 3.3 and Figure 3.4.

Figure 3.3(a) and (b) show the total expected reward with 95% confidence intervals against category size, k, for two different priors values: (a) $\alpha_{0x} = \beta_{0x} = 1$ for all x with a prior mean $\frac{\alpha_{0x}}{\alpha_{0x}+\beta_{0x}} = 0.5$, (b) $\alpha_{0x} = 1$ and $\beta_{0x} = 19$ for all x with a prior mean $\frac{\alpha_{0x}}{\alpha_{0x}+\beta_{0x}} = 0.05$. As the size of categories increases, the MDP-IF policy outperforms both UCB and pure exploitation, and their difference becomes more significant because efficient exploration in MDP-IF provides benefits in scenarios with large categories. Compared to Figure 3.3 (a), the MDP-IF policy in Figure 3.3 (b) outperforms UCB and pure exploitation for larger categories since the scenario in Figure 3.3 (b) has smaller prior mean and variance. When fixing the mean of the beta prior to be $\mu_{0x} = \frac{\alpha_{0x}}{\alpha_{0x}+\beta_{0x}} = \frac{1}{2}$ in Figure 3.3 (c), the plot shows that the gap between the expected total rewards of MDP-IF and ones from other policies enlarges as the variance of beta prior on user preferences increases. Across all scenarios, MDP-IF has the shortest distance to the upper bound among all policies, showing it has the smallest optimality gap.

Similarly to the simulation setup illustrated in Figure 3.3, we perform two additional experiments in Figure 3.4 to understand how MDP-IF behaves as item arrival rate and users' discount factor, γ, changes. Figure 3.4 (a) shows the expected total reward against $E[L_{\cdot x}] = \frac{1-\xi_x}{\xi_x}$, the average number of available stepwise items in each category. The expected total reward increases as the number of available items in each category increases. Moreover, MDP-IF outperforms UCB and pure exploitation more when the expected number of arriving items is larger. Figure 3.4 (b) shows the expected total reward per $\frac{\gamma}{1-\gamma}$ against various discount factors, γ. The MDP-IF policy outperforms UCB and pure exploitation at all discount values in $[0.90, 0.95, 0.98, 0.99, 0.995]$, but with a smaller gap from UCB as discount factor increases because users have more time to recover from exploration when they spend longer time in the system.



(a) $M = 10, c = 0, k = 50, \gamma = .9, \alpha_{0x} = \beta_{0x} = 1, \forall x$

(b) $M = 10, c = 0, k = 50, \xi_x = .1, \alpha_{0x} = \beta_{0x} = 1, \forall x$

Figure 3.4: Each sub-figure shows plots of expected total reward with 95% confidence intervals under each policy (MDP-IF in solid blue line, UCB in red dashed line, Pure Exploit in green dotted line) with a given parameter setting specified in its sub-caption. $100,000$ users are simulated in each scenario. UCB runs various simulations for $\rho \in \{0.65, 0.7, 0.75, 0.8, 0.85, 0.9, 0.95\}$, with the best results plotted. (a) shows expected total reward against average number of items, $E[L_{\cdot x}] = \frac{1-\xi_x}{\xi_x}$, while (b) plots expected total reward per $\frac{\gamma}{1-\gamma}$ against various discount factors, γ.

CHAPTER 4

CLUSTERING SCHEMES FOR CATEGORIZATION

Motivated by an application to the arXiv (arXiv.org, 2014) and a categorization method needed in Chapter 2 and Chapter 3, in this chapter we consider the problem of finding hard clusters of scientific articles in the presence of user-interaction data and document content. To this end, we develop a generative model of user-item interaction as well as item content, such that each document is associated with a single scalar latent variable, indicating its cluster membership. Our model is essentially a stochastic blockmodel applied to the bipartite user-item interaction graph, combined with a content distribution on the document vertices.

By the application to arXiv, we believe that there exists a finer-grained categorization of papers than currently exists, which will be used to offer users more fine-grained control over the daily and weekly feeds of newly submitted papers, and within a new information filtering system that will learn user preferences illustrated in Chapter 2 and Chapter 3. The primary user base will be those who frequently visit the website to stay up-to-date with the research community, as is common among physics researchers.

Below, in Section 4.1, we introduce topics on stochastic blockmodels and discuss the state-of-arts in the literature. In Section 4.2 we provide a detailed description of the model, and in Section 4.3 we describe how variational methods can be used for inference. In Section 4.4 we apply the model to two real-world datasets, from the arXiv, and compare them to a several baseline clusters.

4.1 Introduction

Much of the data being created on the web contains interactions between users and items. Stochastic blockmodels, and other methods for community detection and clustering of bipartite graphs, can infer latent user communities and latent item clusters from this interaction data. These models were introduced to discover latent community structure in graphs (Holland et al., 1983), typically formed by people or other entities interacting with each other (each person is a node, and edges indicate interactions), or by people interacting with text documents, images, videos, or some other object (each person is a node, each object is a node, and edges indicate interactions, forming a bipartite graph). In this second kind of application, interaction information is the only information typically used, and information from the documents themselves is ignored.

Traditional bipartite stochastic blockmodels assume that different communities tend to interact differently with each text document, with some communities tending to interact more frequently with a given document type, and other communities interacting more frequently with other document types. This differential preference of communities for documents induces a latent document clustering, with bipartite stochastic blockmodels attempting to learn this latent clustering from interaction information alone. Our model adds an additional assumption: that documents in each cluster have distinct characteristics, observable in the words that occur in them. When this assumption is satisfied, we argue that it can and should be used to improve performance.

While this assumption does not necessarily hold in all community detection applications, we argue that it holds in a wide variety of settings. In this chap-

ter, we apply this model to scientists interacting with scientific articles, where the words that tend to appear in articles preferred by a community vary considerably from community to community. Our model could also be applied to communities interacting with other kinds of items, e.g., videos, but in our empirical studies we focus on text.

Our model can also be seen as a co-clustering algorithm, because it provides a clustering of both users and documents. However, our model is distinguished from all other co-clustering algorithms of which we are aware, in that it uses not just the interaction information, but also co-variates observable in the documents. Thus, our model is distinguished from co-clustering approaches that use only document comments (e.g., based only the matrix of word co-occurrence) by the way it takes advantage of user co-access to find the mapping of contents to clusters that matches the communities' preferences. It is distinguished from co-clustering approaches that only use user interactions in that document contents are used to refine and improve the co-clustering.

An additional advantage of including document covariates into our model is new documents with no interaction history can be included into an appropriate document cluster, addressing the cold-start problem.

There has been growing interest in combining user interaction and item content in the context of recommender systems (Wang and Blei, 2011; Claypool et al., 1999; Balabanović and Shoham, 1997; Salter and Antonopoulos, 2006; Basilico and Hofmann, 2004; Melville et al., 2002), as well as combining citations and content in the context of community discovery (specifically on document networks) (Nallapati et al., 2008; Yang et al., 2009; Zhou et al., 2009). To our knowledge, this chapter is the first to approach the problem of clustering as

a community detection task on the network of user-document interactions.

4.2 Content-Augmented Stochastic Blockmodels

Suppose we are dealing with an application on the web such that there are D items and U users potentially interested in the items. Suppose that over time the users have been shown and provided feedback on a subset of items. We encode their feedback with the variable Y, defined by

$$Y_{ij} = \begin{cases} 1 & \text{if item } i \text{ was shown to user } j \text{ and relevant} \\ 0 & \text{if item } i \text{ was shown to user } j \text{ and irrelevant} \\ \triangle & \text{if item } i \text{ was not shown to user } j. \end{cases}$$

Note we only consider a binary response (relevant / irrelevant), and use the symbol $\triangle = Y_{ij}$ to denote the case where item i is not shown to user j.

The model proceeds by assuming there are k_d clusters such that each item i belongs to cluster $z_i \in \{1, \ldots, k_d\}$ and there are k_u communities such that each user j belongs to community $w_j \in \{1, \ldots, k_u\}$. We assume the community membership of a user and cluster membership of a paper completely determines the probability the user finds the paper relevant. Explicitly, the model assumes that

$$p(Y_{ij} = 1 | z_i = x, w_j = y, Y_{ij} \neq \triangle) = q_{xy} \tag{4.1}$$

for constants q_{xy} ranging over item clusters x and user communities y. For simplicity, we encode the q_{xy} as a matrix $Q = [q_{xy}]$. We assume the observed Y_{ij} are all sampled independently.

Finally, we endow the latent variables described above with the following Bayesian priors:

- The cluster z_i of item i follows a uniform distribution on k_d elements.
- The community w_j of user j follows a uniform distribution on k_u elements.
- The cluster-community interest probabilities q_{xy} follow a Beta(α, β) for some $\alpha, \beta > 0$.

We further simplify things by assuming that $k_u = k_d$, that is the number of user communities and document clusters is equal, and use the variable K to indicate this value.

The model described up to this point is a stochastic blockmodel on a bipartite graph, without any notion of item content.

To augment the model with content, we suppose that each item can be represented by an F-dimensional feature vector, such that the nth entry counts how many time the nth trait occurred, for some set of F traits. Let d_i represent the feature vector for the ith item.

In order to force that items in the same cluster should be similar in content, we assume that associated with each item cluster x is a probability vector $p_x \in [0,1]^F$, $\sum_\ell p_{x\ell} = 1$, such that if item i is in cluster x then the feature vector d_i is created from N_i samples of a Multinomial(p_x) distribution (the process by which N_i is chosen is unimportant). We assume the observed d_i are all sampled independently, and we place a Dirichlet(γ) prior on each p_x, for some $\gamma \in (\mathbb{R}_{>0})^F$.

This fully describes the content-augmented stochastic blockmodel (CASB). In the following section we describe how variational inference and Gibbs sampling can be used to infer the latent variables. A graphical depiction of the CASB can be seen in Figure 1, illustrating the dependencies between all latent variables.

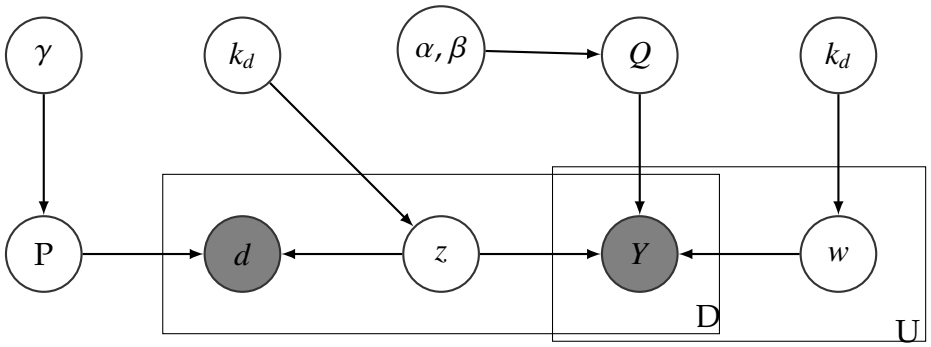


Figure 4.1: Graphical representation of the content-augmented model

4.2.1 Related Work

There is growing literature at the intersection of clustering and community-detection, as well as combining community-detection approaches with node attributes (content). In the context of document networks, the links considered are most often citations, as opposed to user interactions.

In Nallapati et al. (2008) the authors introduce two models which jointly describe text and citations. The first combines latent Dirichlet allocation Blei et al. (2003) with mixed-membership stochastic blockmodels Airoldi et al. (2009). They find this model leads to relatively intractable inference. In response, they introduce another model called Link-PLSA-LDA which associates a multinomial distribution with each article, from which the articles citations are drawn. Both models use the graph structure and article content to learn latent *vector representations* for each of articles. While powerful, these do not immediately lend themselves to hard clusters.

The problem of clustering an arbitrary graph with node attributes is studied in Zhou et al. (2009). Rather than taking a probabilistic approach the authors augment the underlying graph by adding a node for each attribute, with links

to each of the original nodes containing the attribute. Vertex closeness is given by a neighborhood random walk model, and the clusters are computed via the resulting distance function.

In Yang et al. (2009) the problem of community detection is approached by introducing a discriminative model combining link and content information. Initially they introduce the Popularity-based Conditional Link model (PCL). PCL assumes there is a latent variable describing each node's community membership, and the probability of a link between two nodes depends on each node's popularity and community membership. Content is introduced into the model by setting the probability of belonging to a specific community as the exponential of a linear function of the node's content vector.

4.3 Inference

It is intractable to directly optimize the likelihood of the latent variables. Instead, we appeal to variational inference techniques to find approximate estimates of the latent variables Wainwright and Jordan (2008). In variational inference, we associate to each latent variable a family of variational distributions each parameterized by a free variational parameter. These parameters are then optimized to find the closest member of the family to the posterior (in terms of KL-divergence). We will use mean-field variational inference, which assumes that the complete joint variational distribution factors.

We assume the following variational parameters and distributions:

$$z_i|\phi_i \sim \text{Multinomial}(\phi_i), \tag{4.2}$$

$$w_j|\varphi_j \sim \text{Multinomial}(\varphi_j), \tag{4.3}$$

$$q_{xy}|\alpha_{xy},\beta_{xy} \sim \text{Beta}(\alpha_{xy},\beta_{xy}), \tag{4.4}$$

$$p_x|\lambda_x \sim \text{Dirichlet}(\lambda_x). \tag{4.5}$$

Let q represent the distribution defined above. For notational convenience we will write expressions involving q with the understanding that the distribution is conditioned on the variational parameters.

Recall the evidence lower bound (ELBO) $\mathcal{L}(q)$ is defined as

$$\mathcal{L}(q) = \mathbb{E}_q[\log p(d, Y, z, w, P, Q)] - \mathbb{E}_q[\log q(z, w, P, Q)] \tag{4.6}$$

and is equal to the KL-divergence between q and the posterior, up to an additive constant. In order to find the optimal variational parameters we optimize $\mathcal{L}(q)$ using coordinate ascent. Following Hoffman et al. (2013); Wainwright and Jordan (2008) the update for each variational parameter in coordinate ascent equals the variational expectation of the natural parameter of the complete conditional corresponding to the relevant latent variable. The complete conditional distributions for each of the latent variables are:

$$\begin{aligned} p(z_i|z_{-i}, w, P, Q, d, Y) \propto &\prod_{\ell=1}^{F} p_{z_i\ell}^{d_{i\ell}} \prod_{j:Y_{ij}=1} q_{z_i w_j} \\ &\prod_{j:Y_{ij}=0} (1 - q_{z_i w_j}) \end{aligned} \tag{4.7}$$

$$p(w_j|w_{-j}, z, P, Q, d, Y) \propto \prod_{i:Y_{ij}=1} q_{z_i w_j} \prod_{i:Y_{ij}\neq 1} (1 - q_{z_i w_j}) \tag{4.8}$$

$$q_{xy}|Q_{-xy}, w, z, P, d, Y \sim \text{Beta}\,(\alpha', \beta') \tag{4.9}$$

Table 4.1: Expected Natural Parameters for Complete Conditionals.

Variable	Parameter	Expected Natural Parameter
z_i	$\phi_{i,x}, x \in \{1, ..., k_d\}$	$\mathbb{E}_q\left[\ln\left[\prod_{\ell=1}^{F} p_{z_i\ell}^{d_{i\ell}} \prod_{j:Y_{ij}=1} q_{z_i w_j} \prod_{j:Y_{ij}=0}(1 - q_{z_i w_j})\right]\right]$
w_j	$\varphi_{i,y}, y \in \{1, ..., k_u\}$	$\mathbb{E}_q\left[\ln\left[\prod_{i:Y_{ij}=1} q_{z_i w_j} \prod_{i:Y_{ij}\neq 1}(1 - q_{z_i w_j})\right]\right]$
q_{xy}	α_{xy}, β_{xy}	$\alpha - 1 + \mathbb{E}_q\left[\sum_{(i,j):z_i=x,w_j=y} Y_{ij}\right]$, $\beta - 1 + \mathbb{E}_q\left[\sum_{(i,j):z_i=x,w_j=y}(1 - Y_{ij})\right]$
p_x	$\gamma_{x\ell}, \ell \in \{1, ..., F\}$	$\gamma_\ell - 1 + \mathbb{E}_q\left[\sum_{i:z_i=x} d_{i\ell}\right]$

Table 4.2: Relevant Expectations. Ψ represents the digamma function.

Variable	Parameter	Relevant Expectation
z_i	$\phi_{i,x}, x \in \{1, ..., k_d\}$	$\Pr[z_i = x] = \phi_{i,x}$
w_j	$\varphi_{i,y}, y \in \{1, ..., k_u\}$	$\Pr[w_j = y] = \varphi_{j,y}$
q_{xy}	α_{xy}, β_{xy}	$\mathbb{E}[\ln q_{xy}] = \Psi(\alpha_{xy}) - \Psi(\alpha_{xy} + \beta_{xy})]$ $\mathbb{E}[\ln(1 - q_{xy})] = \Psi(\beta_{xy}) - \Psi(\alpha_{xy} + \beta_{xy})]$
p_x	$\gamma_{x\ell}, \ell \in \{1, ..., F\}$	$\mathbb{E}[\log p_{x\ell}] = \Psi(\lambda_{x\ell}) - \Psi(\sum_l \lambda_{xl})$

$$p_x | P_{-x}, w, z, Q, d, Y \sim \text{Dirichlet}(\gamma') \tag{4.10}$$

The parameters in equations (4.9) and (4.10) are given by:

$$\alpha' = \alpha + \sum_{(i,j):z_i=x,w_j=y} Y_{ij} \tag{4.11}$$

$$\beta' = \beta + \sum_{(i,j):z_i=x,w_j=y} 1 - Y_{ij} \tag{4.12}$$

$$\gamma'_\ell = \gamma_\ell + \sum_{i:z_i=x} d_{i\ell} \tag{4.13}$$

The expected natural parameter for each of these distributions is summarized in Table 4.1, while their relevant expectation is expressed in Table 4.2. To implement coordinate ascent, each update is simply given by the corresponding entry's expected value under q.

4.4 Evaluation

In this section we fit the CASB to two datasets taken directly from the arXiv, and compare the results to several baseline clusters.

The first dataset consists of all 7 819 papers uploaded to the arXiv in the astro-ph.CO (cosmology) category between 2009 and 2010, and all 621 users who visited the astro-ph.CO "/new" or "/recent" page at least 5 times and read at least 30 articles. We form a positive link ($Y_{ij} = 1$) between a user and all the papers they read, and a negative link ($Y_{ij} = 0$) between a user and all the papers that appeared on the astro-ph.CO "/new" or "/recent" pages on the days they visited that they did not read. For documents that appeared on days when the user did not visit we set $Y_{ij} = \triangle$, making for a total of 1 090 588 non-$\triangle$ links and 65 814 positive links.

We constructed the second dataset analogously. We selected all 6 677 papers uploaded to hep-th (theoretical high-energy physics) between 2009 and 2010, and all 1 449 users satisfying the criteria above, who additionally read at least 70 articles. We defined Y in the same manner as above, leading to 4 579 019 non-$\triangle$ links and 318 703 positive links.

We considered two standard methods of representing documents to illustrate the flexibility of the model. One representation was simply bag-of-words (BOW), treating the document as a vector counting each of the words in its abstract, so $d_{i\ell}$ represents how many times the ℓth word appeared in the abstract of the ith document, giving rise to document vector d_i. We truncated the abstract vocabulary to remove highly uncommon words, leading to a vocabulary of size 4 951 for astro-ph.CO documents and 3 282 for hep-th documents. (We

also did evaluations with the full text of the papers, but were forced to more significantly truncate the size of the vocabulary to improve speed. We found better performance using abstracts than with full texts, using a truncated vocabulary).

For the other representation we preprocessed the data by fitting latent Dirichlet allocation, and treating documents as counts over topics. That is, for the ith document with content vector d_i we let $d_{i\ell}$ represent the number of times a word from the ℓth topic appeared in the abstract. In our experiments we fit latent Dirichlet allocation with 50 topics.

We chose to focus on the astro-ph.CO and hep-th categories because physicists who study cosmology and high-energy physics are some of the heaviest users of the arXiv. Many of these users visit the website daily to stay up-to-date with the research community. As such, these categories have vast user data that is highly representative of the many subcommunities. These frequent users are also the primary target of the recommender system currently being built.

To validate the quality of the CASB clusters, we introduced several benchmarks.

In Nallapati et al. (2008) Nallapati et. al introduce the Link-PLSA-LDA (L-P-LDA) model. L-P-LDA is a graphical model that combines latent Dirichlet allocation (LDA) (Blei et al., 2003) and the mixed-membership stochastic blockmodel (MMSB) (Airoldi et al., 2009). It learns latent vector representations of the documents satisfying both the topic structure learned by LDA and the community structure learned by MMSB. Of all the benchmarks this one is most similar to ours, in that the latent variables capture both community structure and node content. We applied this model to both arXiv datasets, treating users as

documents with empty content. Once the vector representations were learned, KMeans was used to arrive at clusters.

In Gopalan et al. (2014), Gopalan et. al introduce CTPF, a generative model of document and reader preferences. CTPF learns user preference vectors and document topic vectors in the same latent space, for the purpose of recommendations. In this model a rating r_{ud} between a user u and document d is drawn according to

$$r_{ud} \sim \text{Poisson}(\eta_u^T(\theta_d + \epsilon_d)) \tag{4.14}$$

for user preference vector η_u, document topic vector θ_d and a small offset vector ϵ_d. The purpose of this model is to learn a latent space for effective document recommendation, this model is similar to ours in that it explicitly considers the interactions between users and items, as well as item contents. We fit this model to the two arXiv datasets and again used KMeans to go from vector representations to clusters.

We looked to Yang et al. (2009) for a benchmark that explicitly learns clusters, based on link data and document content. Yang et. al propose PCL, a model of link formation where the probability of a link depends on a node's latent cluster membership (similar to our setting with the CASB). They extend this model to PCLDC by discriminatively incorporating content: if $x_i \in \mathbb{R}^d$ is the ith node's content, they assume

$$p(z_i = k) = \frac{\exp(w_k^T x_i)}{\sum_l \exp(w_l^T x_i)} \tag{4.15}$$

where $w_k \in \mathbb{R}^d$ is a weight-vector associated with the kth cluster. Since this model requires that each node be associated with a content-vector, we trained this model on the arXiv data by assigning each user's content to be the 0-vector. Unfortunately this seriously hindered the performance of the PCLDC model

as a benchmark. It would be possible to modify this model to handle nodes without content, but we did not do so.

As another benchmark, we trained 50 dimensional article vectors with the PV-DBOW method described by Dai et al. (Dai et al., 2014). These article vectors are trained to be predictive of the text within the article using a hierachical softmax estimation of the log-linear objective,

$$P(w|a) = \frac{\exp(v_w \cdot v_a)}{Z_a} \quad Z_a = \sum_w \exp(v_w \cdot v_a). \tag{4.16}$$

Article vectors trained in this way have proven useful to semantic analysis tasks Le and Mikolov (2014) as well as retaining semantic similarity of the articles Dai et al. (2014). The article vectors were trained on the text extracted from the pdfs of the articles, after lowercasing the text and inserting word boundaries at each non-alphanumeric character. Any 'word' appearing less than 30 times was cut from the vocabulary. After training, the article vectors were clustered according to spherical k-means as described in Coates et al. Coates and Ng (2012).

To improve the utility of the article vectors, in the larger training example, the astro-ph.CO and hep-th articles were augmented with a set of 98 392 articles chosen to be representative of all of the categories on the arXiv.

Finally, we also used KMeans on the LDA vectors as a benchmark.

In order to select the proper value of K, we fit the CASB to each dataset for $K = 2, \ldots, 10$, and for each of the learned clusters we calculated the ELBO as in 4.6. We selected the largest value of K that contributed at least a 5% increase to the ELBO. For both datasets this resulted in $K = 6$.

In addition, we present a qualitative evaluation of the model applied to a third dataset. This dataset consists five years of conference proceeding data

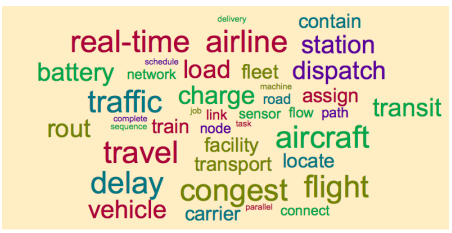
from the annual INFORMS conference INFORMS (2015), the largest conference of its kind for practitioners of operations research. Papers at INFORMS are presented in sessions, where a session chair will choose three or four papers relevant to the session's subject. We treat the set of authors as the users in our model, such that each author has a positive link to every paper presented in the same session as the author's own paper and a negative link to every other paper.

4.4.1 Community Discovery on INFORMS

The INFORMS dataset is rich for study because the conference represented has a large number of sessions and presented papers (more than 1 000 sessions, and just under 4 000 papers INFORMS (2015)). It includes many distinct research subcommunities, which CASB is designed to discover. It is also a field that is very applied in nature, and one in which the authors have expertise, making evaluation of the clusters easier than for the two physics datasets. We trained the CASB on this dataset, setting $K = 5$ arbitrarily.

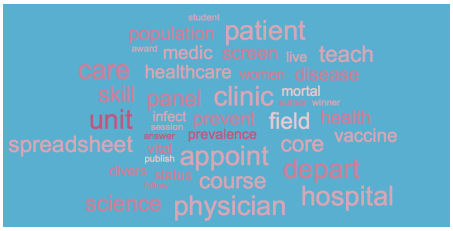
To evaluate the quality of these clusters, we formed word cloud visualizations of the frequently occurring words in each cluster. More specifically, for each word w we formed the scalars $w_1, \ldots, w_K$ such that w_i is the proportion of papers containing the word w belonging to the ith cluster. Then, in the word cloud corresponding to cluster i, the weight for the wth word is given by w_i. This weighting scheme ignores popular stop words since their distribution will be uniform across all clusters, whereas words that frequently occur in the ith cluster but do not occur in other clusters will have high weight. We limit our-

(a) Optimization word cloud

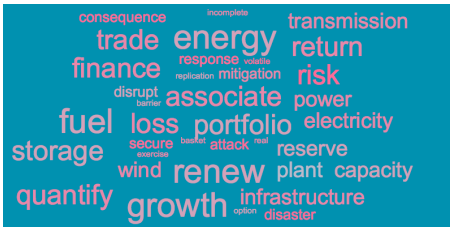


(b) Transporation word cloud

(c) Supply Chain word cloud



(d) Health Care word cloud



(e) Energy word cloud

Figure 4.2: Word clouds demonstrating the research communities learned by CASB within the INFORMS dataset. Each word cloud corresponds to a distinct community, and shows words whose relative frequency are high in that communities' papers. This qualitative result shows that CASB is able to distinguish meaningful research communities in the INFORMS dataset.

selves to words appearing in more than 50 papers.

Figure 4.2 displays five such word clouds. The first word cloud is clearly representative of the mathematical optimization research community with words such as "inequality", "separation" and "relax". The second word cloud is clearly representative of the transportation logistics community, with words such as "airline", "congest" and "real-time". The third word cloud focuses on supply chain and economics, with words such as "elasticity", "retail", and "assort". The fourth word cloud is a representative of Health-care communities, such as words in "vaccine", "hospital" and "clinic". Lastly, the fifth word cloud

is a representative of energy and infrastructure related industry, in which we see words such as "fuel", "transmission" and "electricity. We have set up a small website for a more complete view of the clusters[1].

While purely qualitative, these word clouds show that the CASB is able to retrieve real-word research communities with high accuracy.

4.4.2 Capturing Misplaced Papers

This evaluation focuses on two astrophysics subcategories on the arXiv: Cosmology and Nongalactic Astrophysics (astro-ph.CO); and Astrophysics of Galaxies (astro-ph.GA).

In creating these categories, the arXiv administrators' intention was for all papers about galactic astrophysics to go to astro-ph.GA. However, in the past, a significant portion of the astrophysics community had a different interpretation: astro-ph.GA was for papers discussing our galaxy, the Milky Way, while papers discussing other galaxies should go to astro-ph.CO.

In late 2013, arXiv.org's moderators began enforcing their interpretation of these two subcategories, recategorizing papers about galaxies from astro-ph.CO to astro-ph.GA Ginsparg (2014).

We hypothesized that the research communities interested in nongalactic and galactic papers differ, as do the words in their papers, and so the CASB should be able to separate older papers from astro-ph.CO into those nongalactic papers that were correctly submitted to astro-ph.CO, and those galactic papers

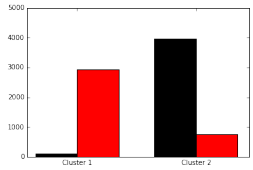
[1] peter-i-frazier.github.io/navigate-informs

that should have been submitted to astro-ph.GA. Moreover, it should be able to do this in an unsupervised way, based only on usage and item content, without being given examples of papers in each class.

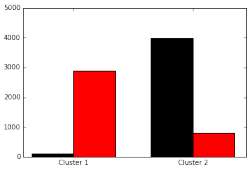
To test this hypothesis, we fit the CASB and each of the benchmarks to our cosmology dataset consisting of papers submitted to astro-ph.CO over 2009–2010, setting $K = 2$. We then compared each of these clusterings to a ground truth classification of papers (described below) into those that were properly submitted to astro-ph.CO, and those that should have been submitted to astro-ph.GA.

To create our ground truth, we used a Naive Bayes classifier trained on papers appearing in the arXiv in late 2013 and early 2014, which were manually reclassified by the arXiv moderators. We then ran this Naive Bayes classifier on the papers in our 2009–2010 dataset. Note that, although the Naive Bayes classifier is able to automatically classify papers as to whether they belong in astro-ph.GA or astro-ph.CO with high accuracy, this classifier required hand-curated training labels from the arXiv moderators, in the form of correct classifications of a large number of papers from 2013 and 2014. In contrast, in this evaluation, CASB and the benchmark methods do not have access to this training data, and instead must make a determination based only on what was available in 2009–2010.

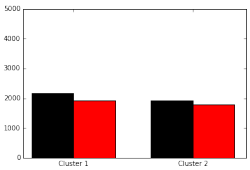
The distribution of cosmology-classified and galaxy-classified papers are presented in Figure 4.3. In Table 4.3 we present the number of misplaced papers for each clustering scheme. We see that the CASB applied to the bag-of-word representations have the fewest misclustered papers, followed closely by the CASB applied to the LDA representations.



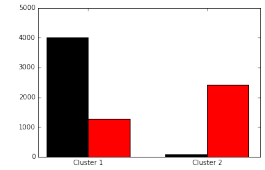
(a) CASB with bag-of-word representations



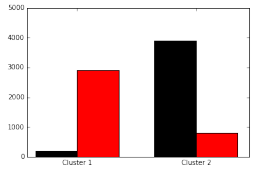
(b) CASB with LDA representations



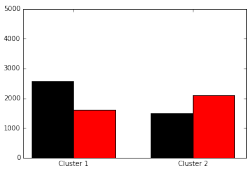
(c) PV-DBOW



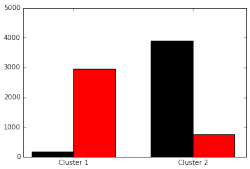
(d) PV-DBOW with auxiliary data



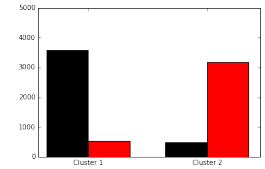
(e) KMeans on Pairwise-link-LDA vectors



(f) PCLDC



(g) KMeans on LDA vectors



(h) Kmeans on Poisson Factorization vectors

Figure 4.3: Distribution of galaxy and cosmology papers amongst clusters of the astro-ph.CO dataset. The red bars represent cosmology papers and the black bars represent galaxy papers. A method that performs well puts cosmology papers and galaxy papers in nearly distinct clusters, so that the red bar is much larger than the black bar in one of the clusters, and the black bar is much larger than the red in the other cluster.

Cluster Type	**Misplaced Papers**
KMeans PV-DBOW	3 829
PCLDC	3 100
KMeans PV-DBOW (auxiliary)	1 357
KMeans Poisson Factorization	1 024
KMeans Link-PLSA-LDA	985
KMeans LDA	936
CASB (LDA docs)	915
CASB (bag-of-words docs)	884

Table 4.3: Number of misplaced papers for each set of clusters. The number of misplaced clusters is taken to be the minimum of $g_1 + c_2$ and $c_1 + g_2$ where g_i and c_i are the number of galaxy and cosmology papers in cluster i, respectively.

Of note is the fact that the PCLDC clusters have a very high number of misclustered papers, despite considering link presence and document content. We hypothesize this is because their discriminative incorporation of content does not generalize well to nodes without content. When the node has zero content, the distribution from (4.15) will be uniform. Since the links in our datasets only exist between users and documents, PCLDC will have a hard time exploiting the structure of the graph when each user node is uniform across all clusters.

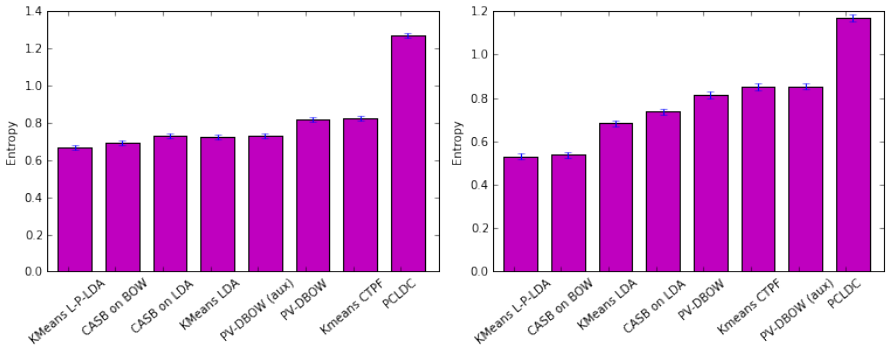
Interestingly, KMeans applied to the LDA representations also has very few misclassified documents. This suggests that there is a lot of signal in the content of the documents pointing to the ground truth communities. As the table shows, the CASB is able to exploit the user data to discover these communities even more accurately.

4.4.3 Author-based Evaluation

To further evaluate the quality of our clusters we looked to authorship data, as the papers a researcher writes are a strong indicator of the communities to which they belong.

Specifically, for each of our datasets we took the set of authors who had written two or more papers. For each of these authors a we formed the distribution $a_1, \ldots, a_K$ where a_i is the proportion of documents a has authored belonging to the ith cluster. (This is the same methodology as determining weights for the word clouds).

Now, if a clustering is representative of the underlying community structure,



(a) **astro-ph.CO** (b) **hep-th**

Figure 4.4: Average author-cluster distribution entropies of the various clusterings for the two arXiv datasets. **Lower is better.**

one would expect an author's distribution to be highly concentrated on one or maybe two clusters. This is because scientific researchers are experts in one or two fields which contain the majority of their work. They sometimes branch out and write papers in other communities, but this is an infrequent activity.

To measure this property, we compute the average entropy $\mathcal{H}$ of an author's distribution for each cluster. Entropy is a measure of the disorder of a distribution. On one extreme, if an author's publications all reside in one cluster the resulting entropy will be 0. On the other extreme, if an author's distribution is uniform the resulting entropy will be $\log_2(6) \sim 2.6$ since we fit the model with 6 clusters. The entropy for an author a is defined as

$$\mathcal{H}_a = -\sum_i a_i \log_2(a_i). \tag{4.17}$$

In Figure 4.4a the average author entropy is plotted for each clustering scheme. In this plot we see a slight reversal in the quality of the benchmarks compared to the partitioning evaluation. The PV-DBOW clusters trained with auxiliary data has almost identical entropy to CASB with LDA representations, and PV-DBOW trained solely on the astro-ph.CO dataset performs better than

all other benchmarks other than the Link-PLSA-LDA clusters. However, the quality of the CASB clusters remain the same: CASB clusters with BOW representations do better than all clusters, and CASB clusters with LDA representations are tied for second as previously mentioned.

In Figure 4.4b we see a similar pattern. The Link-PLSA-LDA clusters have marginally better performance than the CASB clusters with BOW representations, which are both in turn better than all other clusters. The LDA KMeans clusters come in third with slightly better performance than the CASB clusters with LDA representations.

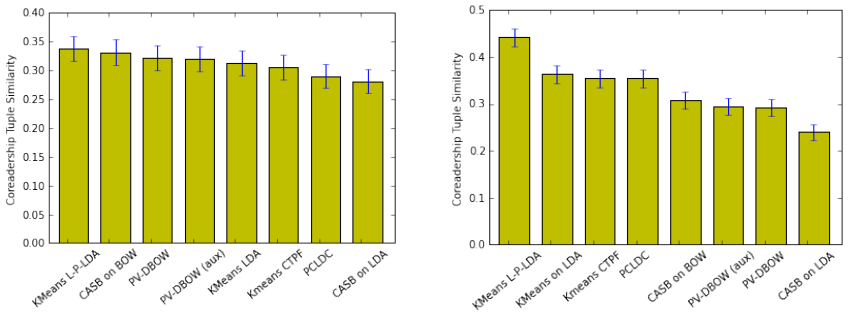
It's surprising that the LDA KMeans clusters have better performance than CASB LDA clusters and KMeans CTPF clusters, since the LDA clusters do not leverage user data at all. However, we still see that leveraging user interaction data is worthwhile, as the Link-PLSA-LDA and CASB with BOW representation clusters have better performance than LDA clusters.

The fact that CASBs learn clusters minimizing the average author-cluster distribution's entropy again shows that the clusters we are learning are truly representative of the underlying community structure in these subcategories.

4.4.4 Coreadership Similarities

To understand the CASB clusters better, we wanted to examine the extent to which high coreadership between two documents determines whether they belong to the same cluster.

Recall for documents d and b with readers $\mathcal{R}_d$ and $\mathcal{R}_b$ respectively, the Jaccard



(a) **astro-ph.CO** (b) **hep-th**

Figure 4.5: Coreadership Similarities of the various clusterings for the two arXiv datasets.

similarity between them is given by

$$J(d, b) = \frac{|\mathcal{R}_d \cap \mathcal{R}_b|}{|\mathcal{R}_d \cup \mathcal{R}_b|}. \tag{4.18}$$

The Jaccard similarity is a measure of overlap between two sets that is agnostic to their size. That is, if one paper has been read by every user, a large intersection between this paper's readership set with another paper's readership set would not contain as much signal as a large intersection between two papers with small overall readership.

To evaluate this criteria, we held out 100 users from each of the arXiv datasets, and reran inference for the CASB and all of the benchmarks. We then selected 3 000 documents from each of the arXiv datasets at random. For each of these selected documents, we construct another set consisting of all those documents whose Jaccard similarity with the original is at least 0.5. Some of the originally selected documents did not have Jaccard similarity greater than 0.5 with any other documents in the corpus so they were discarded, leaving us with 2 442 documents from hep-th and 1 750 documents from astro-ph.CO. From each of these similarity sets we selected one document at random, giving rise to a tuple containing the original document and another document, which

together possess Jaccard similarity of at least 0.5.

After arriving at these tuples with high-coreadership, we simply calculated the proportion which belonged to the same cluster. The results are summarized in Figure 4.5. Of note is that CASB with LDA representations has the lowest coreadership similarity proportion for both astro-ph.CO and hep-th. Not quite as extreme, the CASB with BOW representations had the second highest coreadership similarity in the astro-ph.CO dataset, and fifth highest in the hep-th dataset. These results show that the CASB clusters are not optimizing for high coreadership within clusters.

This can be explained due to the assumptions inherent in the model. Recall the variable q_{xy} represents the probability of a document in cluster x being clicked by a user in cluster y. Hence if two documents have high Jaccard similarity, it is not necessarily indicative that they arise from the same community. Rather, it is possible that both documents belong in separate clusters, but there is a community of users interested in both of these clusters. As we see in the author-based evaluation section, this assumption does not prevent the CASB from learning the true underlying community structure.

4.5 Conclusion

In this chapter we have presented the content-augmented stochastic blockmodel (CASB), a probabalistic model of user-item interactions and item content. The cornerstone assumption of this model is that users and items exist in communities such that their community memberships determine the probability of an interaction, and the content of items in the same cluster is generated from the

same distribution. We fit this model to two real-world datasets taken from the arXiv. We found that the learned clusters had the highest accuracy in distinguishing between two real-world communities contained in the dataset, and they gave rise to author-cluster distributions with low entropy. Both of these results indicate that the model's assumptions are valid, and the learned clusters are of high-quality.

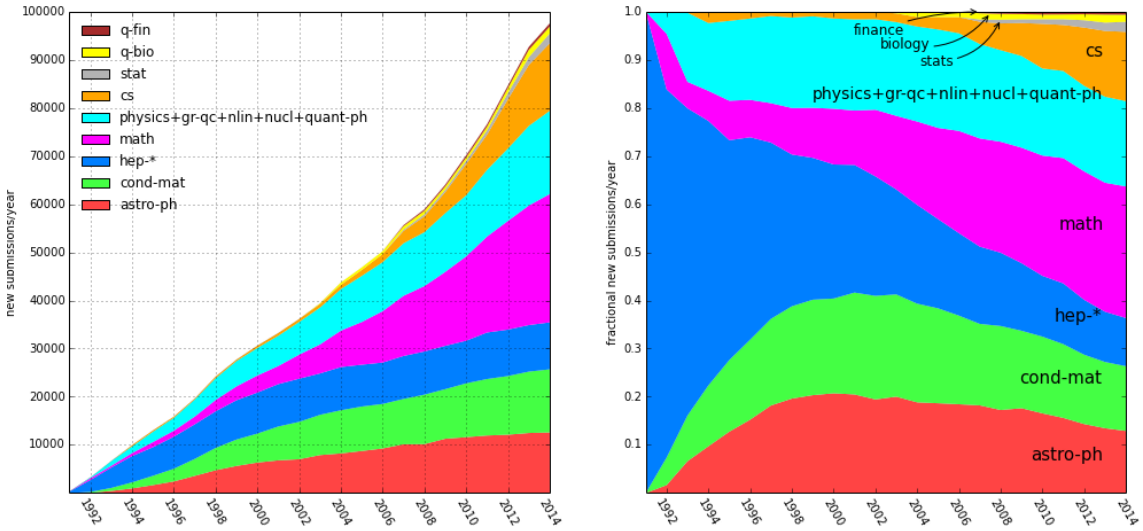
CHAPTER 5

APPLICATION TO ARXIV.ORG

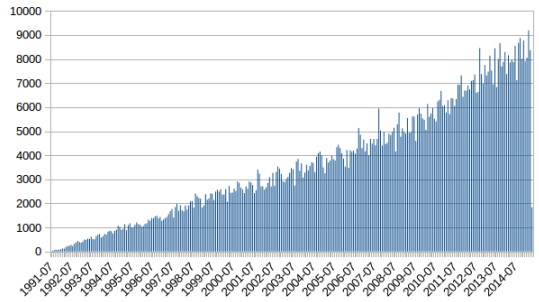
Our information filtering algorithm is motivated by the design of my.arXiv.org, a personalized article feed in arXiv.org, an open electronic repository of scientific articles started in the early 1990s and currently hosted by the Cornell University Library. As of early 2015, arXiv had accumulated 1 million user-submitted scientific articles in physics, mathematics, computer science, quantitative biology, quantitative finance, and statistics. At this time, arXiv was distributing over one million downloads of full-text articles to 400,000+ unique users each week, and every month there were thousands of new articles submitted to the repository (Ginsparg, 2011; Van Noorden, 2014). See Figure 5.1 for statistics of arXiv sumissions and downloads from 1991 to 2014.

Currently arXiv maintains 18 domains, which are further broken down into 140 subject classes. Popular domains, such as "astrophysics" or "condensed matter physics", may have over 70-100 new articles per day. This enormous number of new articles creates a challenge for regular researchers to keep track of the latest research in their fields. My.arXiv.org is created as an extension of arxiv.org to provide several types of personalized recommendations. One algorithm implemented for providing these recommendations to registered users is MDP-IF, shown in Chapter 3, with a categorization scheme from Chapter 4. Other personalization features are provided for anonymous users.

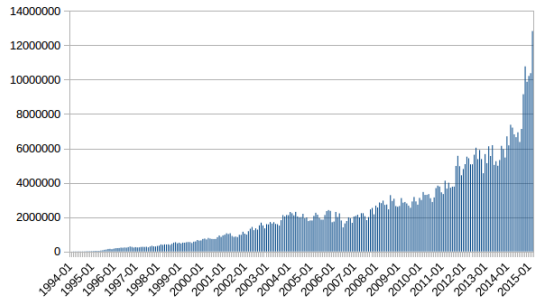
In Section 5.1, we examine the validity of model assumptions made in Chapter 2 and Chapter 3, and then introduce in Section 5.2 the application of the exploration vs. exploitation approach implemented in my.arXiv.org.



(a) The left plot shows the number of yearly arXiv submissions (1991-2014) for each of the following domains: "q-fin" (finance), "q-bio" (biology), "stat" (statistics), "cs" (computer science), "physic+gr-qc+nlin+nucl+quant-ph" (other physics), "math" (mathematics), "hep-*" (high energy physics), "cond-mat" (condensed matter physics), and "astro-ph". The right plot shows each domain's yearly submission rate[1].



(b) Monthly submission of new articles to arXiv.org from 1991 to early 2015[2].



(c) Number of downloads per month from 1994 to early 2015[3].

Figure 5.1: Statistics of arXiv article submissions and downloads from 1991-2014.

5.1 Evaluation of Model's Assumptions in arXiv.org

In the mathematical models described in Chapter 2 (and similar ones in Chapter 3), we made the four following main assumptions:

[1] `arxiv.org/help/stats/2014_by_area/index`, accessed on May 10, 2015
[2] `arxiv.org/stats/monthly_submissions`, accessed on May 10, 2015
[3] `arxiv.org/stats/monthly_downloads`, accessed on May 10, 2015

1. Arrivals of items follow a Poisson process and the lifetime of a user in the system is exponentially distributed, so that the number of available documents from each category to the user follows a geometric distribution;
2. The prior distribution of θ_x for each $x \in \{1, 2, ..., k\}$ follows a beta distribution;
3. User interests θ_x for $x \in \{1, 2, ..., k\}$ are independent across categories;
4. The number of items in the cluster x viewed by the user, N_x, and probability of relevancy of an item from cluster x to the user, θ_x, are independent.

We showed trace-driven simulation results in Section 2.4.2 using the web server logfile from arXiv, including real historical user interactions and items extracted from the logfile from 2009 to 2010. There we mentioned that there is a smaller discretion between the idealized simulation results and the trace-driven simulation results due to violations in Assumption (3) and (4). In the following, we evaluate these assumptions for arXiv's nine categories in the "condensed matter physics" domain[4].

Figure 5.2 shows histograms of the number of daily article arrivals for the nine categories in "cond-mat" for articles submitted from 2009 to 2010. A Poisson distribution is then fitted to each histogram via maximum likelihood estimation and then plotted to allow visual vilidation of the goodness of fit to a Poisson distribution. In Section 2.4.2, we set criteria to identify useful users in the arXiv logfile. For each of these users, if he/she visits the category's new or

[4]The "condensed matter physics" domain includes "cond-mat.dis-nn" (Disordered Systems and Neural Networks), "cond-mat.mtrl-sci" (Materials Science), "cond-mat.mes-hall" (Mesoscale and Nanoscale Physics), "cond-mat.other" (Other Condensed Matter), "cond-mat.quant-gas" (Quantum Gases), "cond-mat.soft" (Soft Condensed Matter), "cond-mat.stat-mech" (Statistical Mechanics), "cond-mat.str-el" (Strongly Correlated Electrons), and "cond-mat.supr-con" (Superconductivity).

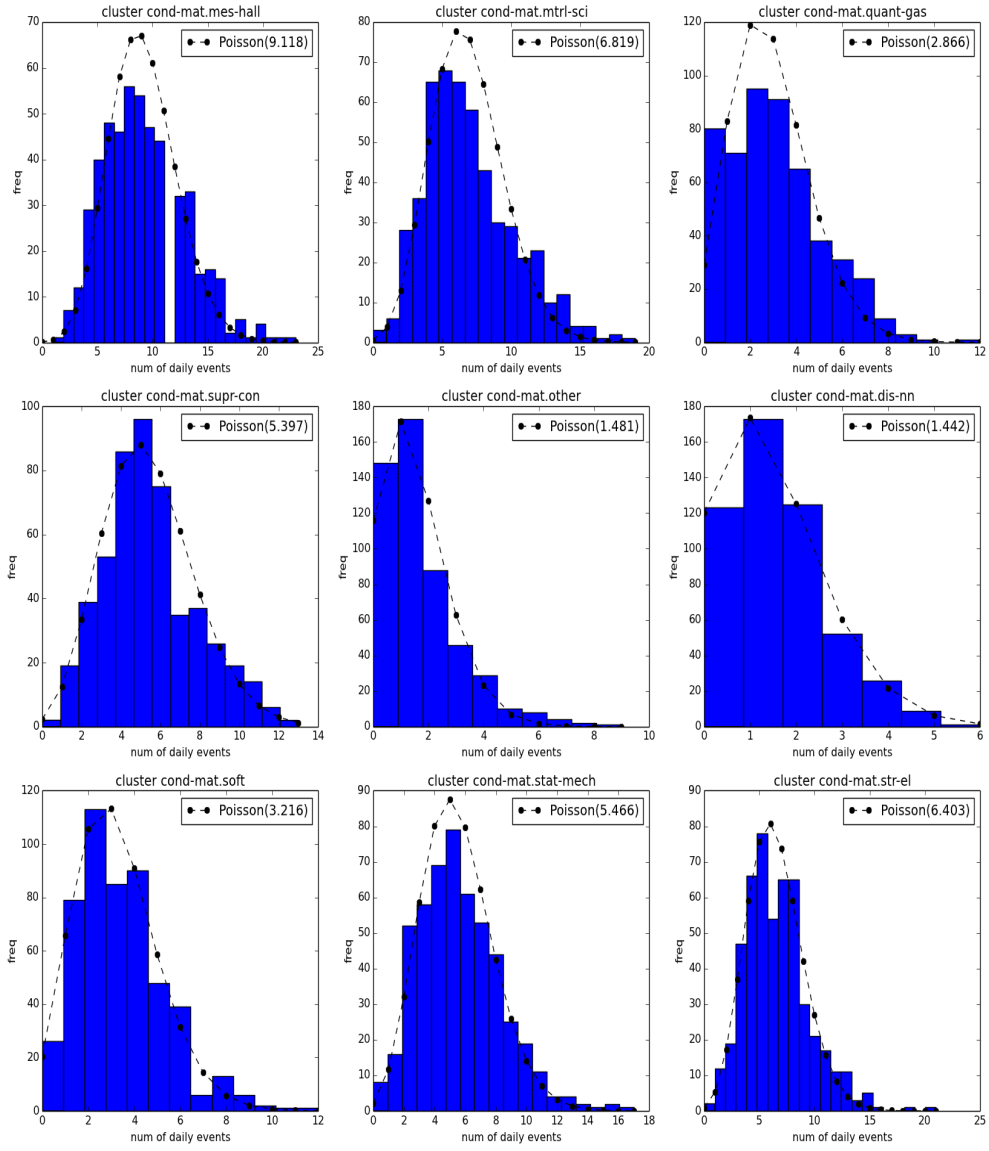


Figure 5.2: Each plot shows a histogram of item daily arrivals for each of nine categories in "cond-mat" (with category name given as the title in each plot) during the period of 2009-2010. Then a Poisson distribution is fitted to the histogram via maximum likelihood estimation.

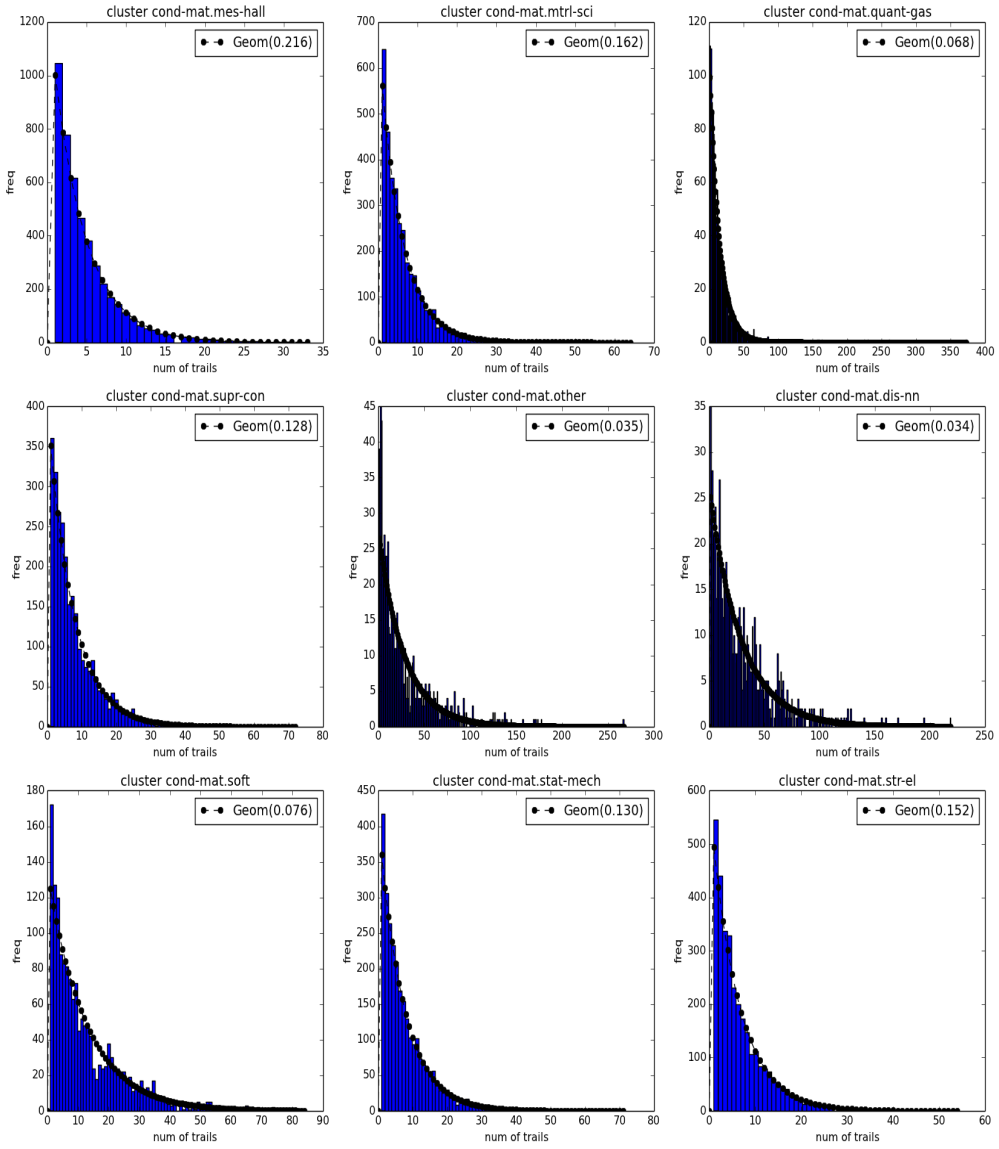


Figure 5.3: Each plot displays a histogram of N_x for each of nine categories in "cond-mat" (with category name given as the title in each plot), accompanied with estimated γ_x for the category through the method of moment. Then a geometric distribution with parameter γ_x is fitted to the histogram via maximum likelihood estimation.

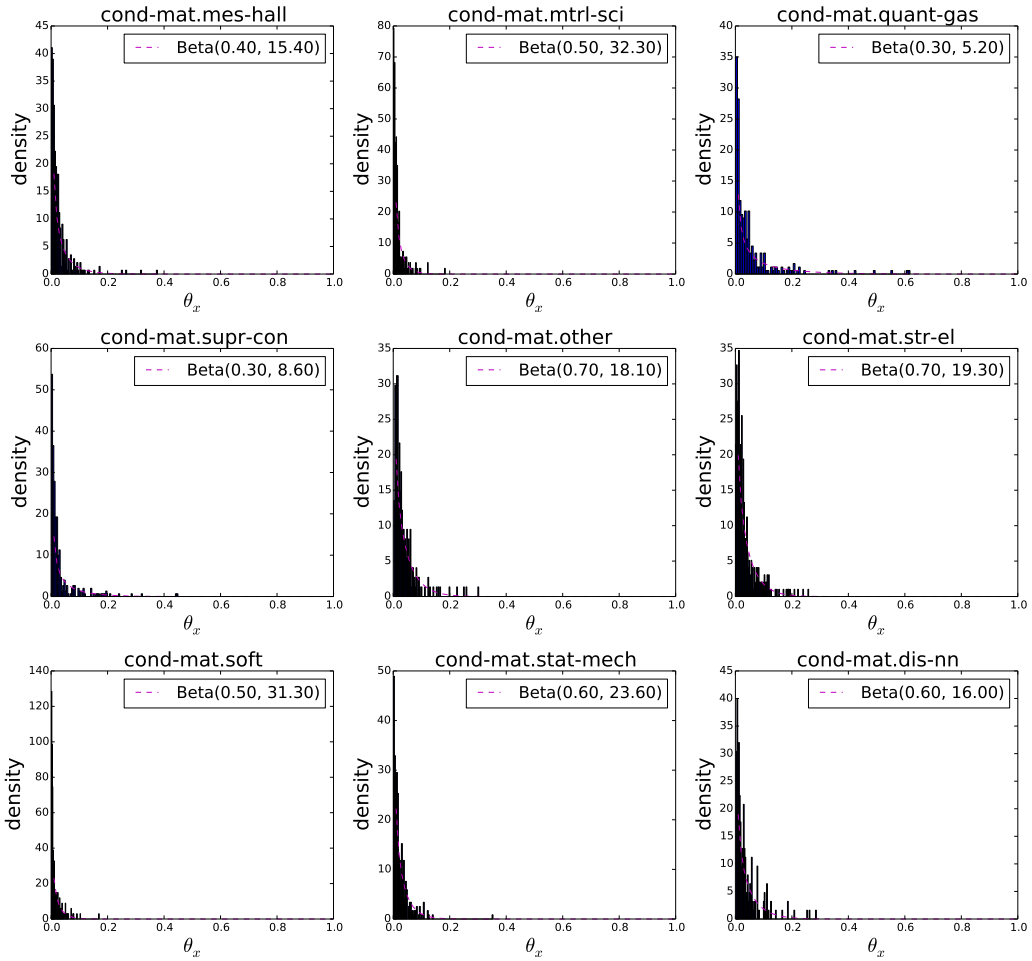


Figure 5.4: Each plot in the figure displays a sample histogram of θ_x for the associated category in condensed matter. Parameters of beta distributions are estimated from the samples through the method of moments.

recent page, we assume that items posted on his/her visit day were available to the user. With that, we can then estimate the number of items available from category x to each user, N_x. Figure 5.3 shows histograms of N_x for each category in "condensed matter physics". Additionally, in each plot we fit a geometric distribution with parameter γ_x, estimated from the histogram using maximum likelihood estimation. From Figure 5.2 and Figure 5.3, we can conclude that the first assumption is reasonable.

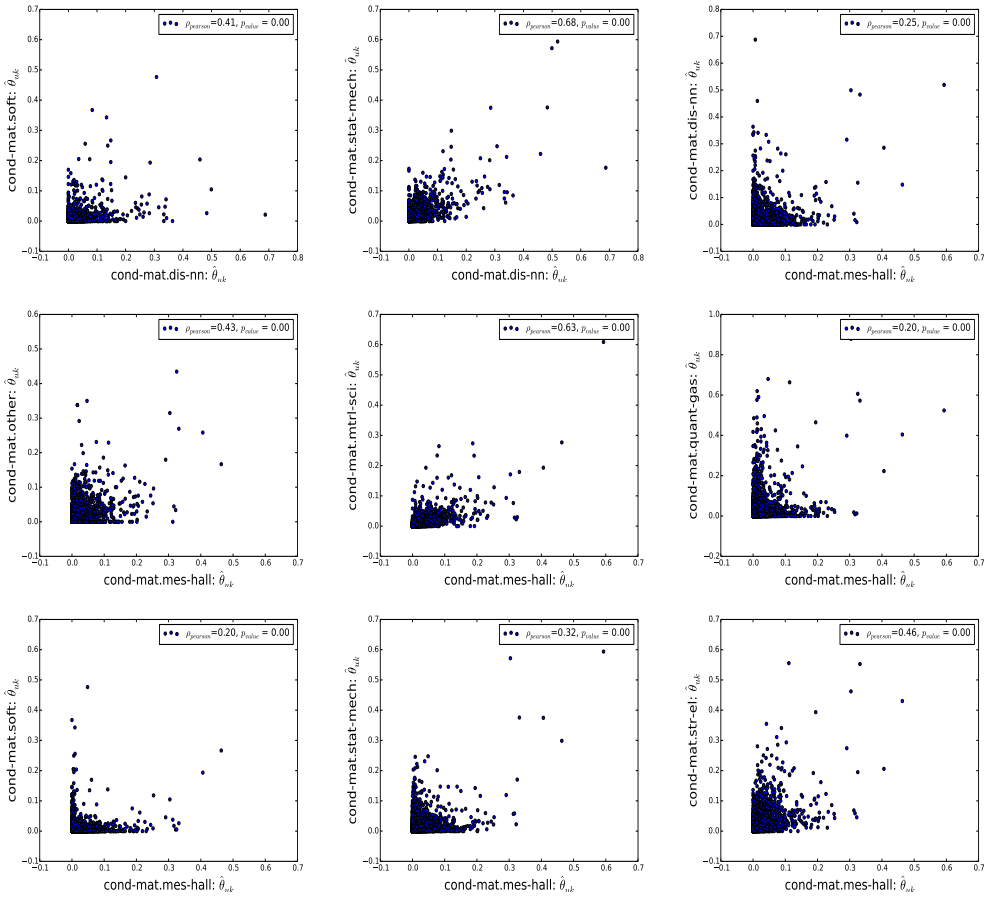


Figure 5.5: Each subplot shows a scatter plot of θ_x between a selected pair of two arXiv categories. In the legend, Pearson correlation along with its p–values are computed.

As described in Section 2.4.2, we can estimate each user's relevance probability, θ_x, for items in category x by calculating his/her click-through-rate on the presented items. Please see Section 2.4.2 for the detail procedure of computing θ_x. Figure 5.4 illustrates histograms of θ_x for each category of the "condensed matter physics" domain. With the method of moments, a beta distribution with parameters α_0 and β_0 is fitted to each histogram. From the histogram, we can see that users tend to have small relevant probabilities, but overall beta distributions fit the data reasonably well, and so the second assumption is reasonable.

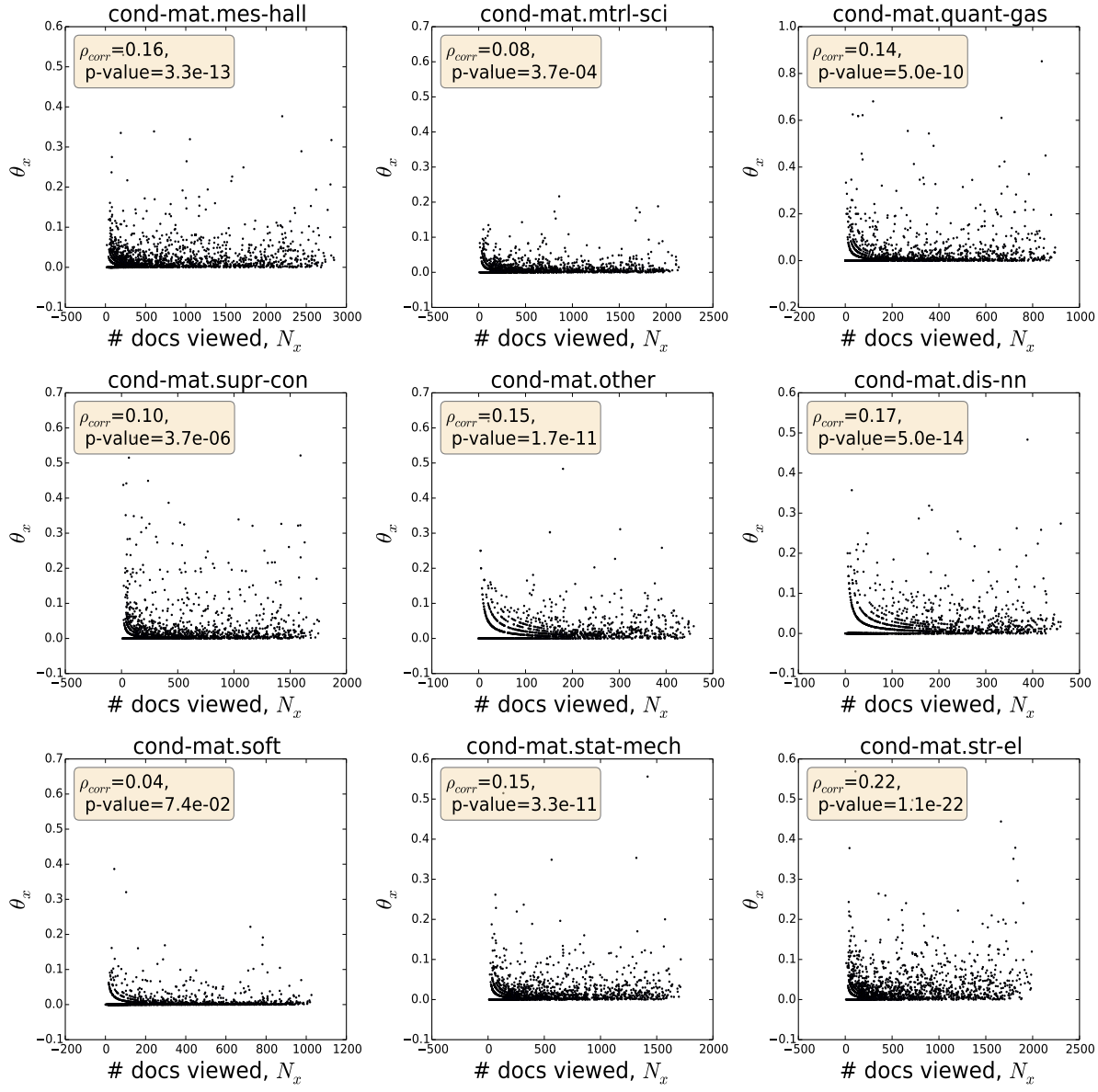


Figure 5.6: Each subplot shows a scatter plot of estimated $\hat{\theta}_x$ vs. $\hat{N}_x$ for the associated category in "cond-mat". In the text box, ρ_{corr} represents the estimated Pearson's correlation coefficient between $\hat{\theta}_x$ and $\hat{N}_x$.

Next, Figure 5.5 shows scatter plots of user preference θ_x among selected nine pairs of categories. In each plot, Pearson correlations and p-values are computed to test linear relationships for every pair of categories. Here, Pearson correlations assume that each of the two underlying datasets, which estimate θ_x across users for a pair of x, is normally distributed, and each p-value indicates the probability of producing two datasets having a correlation at least as extreme as the computed one, under the null hypothesis that these two datasets are uncorrelated. All pairs shown have relatively large correlations, and their correlations are statistically significant at a 1% α–level. This suggests that real user behavior in arXiv.org violates the third assumption.

Figure 5.6 shows scatter plots of θ_x, user preference, against N_x, the number of items shown to each user. Additionally, the Pearson correlation and its p–value between each pair of θ_x and N_x are estimated. All categories have small positive Pearson correlations between N_x and θ_x, but these correlations are statistically significant at 1% α–level. This suggest that real user behavior in arXiv.org modestly violates the fourth assumptions.

Using the exisitng categories defined in arXiv, we can conclude that assumption (1) and (2) are reasonable while violations occur in assumption (3) and (4) under this clustering scheme.

5.2 Current Implementation at my.arXiv.org

My.arXiv.org is a web application that serves as an extension of arXiv.org, containing personalization tools. It maintains a separate database for a replicate of arXiv's article database, and records interactions of users in the system to bet-

ter generate personalized recommendation lists of articles for my.arXiv users. There are two main types of receommendation in my.arXiv: session-based and daily.

Session-based recommendation is primarily designed for anonymous users who do not want to log into my.arXiv, but it also works for registered users. This recommendation only uses actions recorded for a user in the current session, in which a session begins from the moment that a user starts interacting with my.arXiv and ends 30 minutes after the user is inactive in the system. After viewing at least three articles in the system, a session-based recommender window will pop-up a list of similar articles recommended to the user, as shown by the screen-shot in Figure 5.7.

Several algorithms are embedded in the Session-Based recommendation to generate the recommendation list:

- subjects (baseline algorithm): list articles that have similar subject categories with the viewed articles.
- abstracts: find articles based on their similarity to titles and abstracts of the viewed articles.
- coaccess: generate recommendation using arxiv.org coaccess data through August 2014.
- abstracts + coaccess: team-draft merge of abstracts and coaccess (see team-draft interleaving method in Chapelle et al. (2012)).
- Collaborative Topic Poisson factorization (CTPF): a model developed in Gopalan et al. (2014).

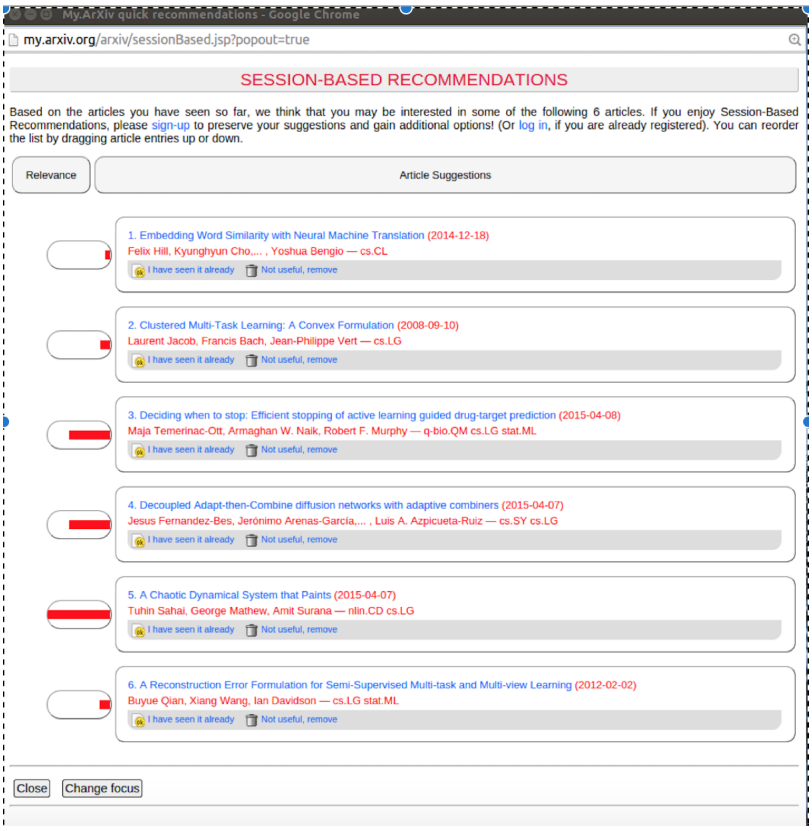


Figure 5.7: Screen-shot of Session-Based recommendation

- random: pick one of the algorithms, abstracts, coaccess, or abstracts+coaccess randomly, and use the recommendation from it. Only one is used at a time, and this can be configured.

In contrast to session-based recommendation, daily recommendations are generated nightly, are more expansive, and are designed for regular logged-in (registered) users. When a user creates a profile at my.arXiv, he/she can optionally provide further information, by designating interest in any of 140 predesigned arxiv categories or uploading personal papers or papers of interest written by others. See a screen-shot of the uploading-paper tool in Figure 5.8 and relevant papers on the method (Charlin et al., 2011; Charlin and Zemel, 2013). With the uploaded information, the daily recommendation system can

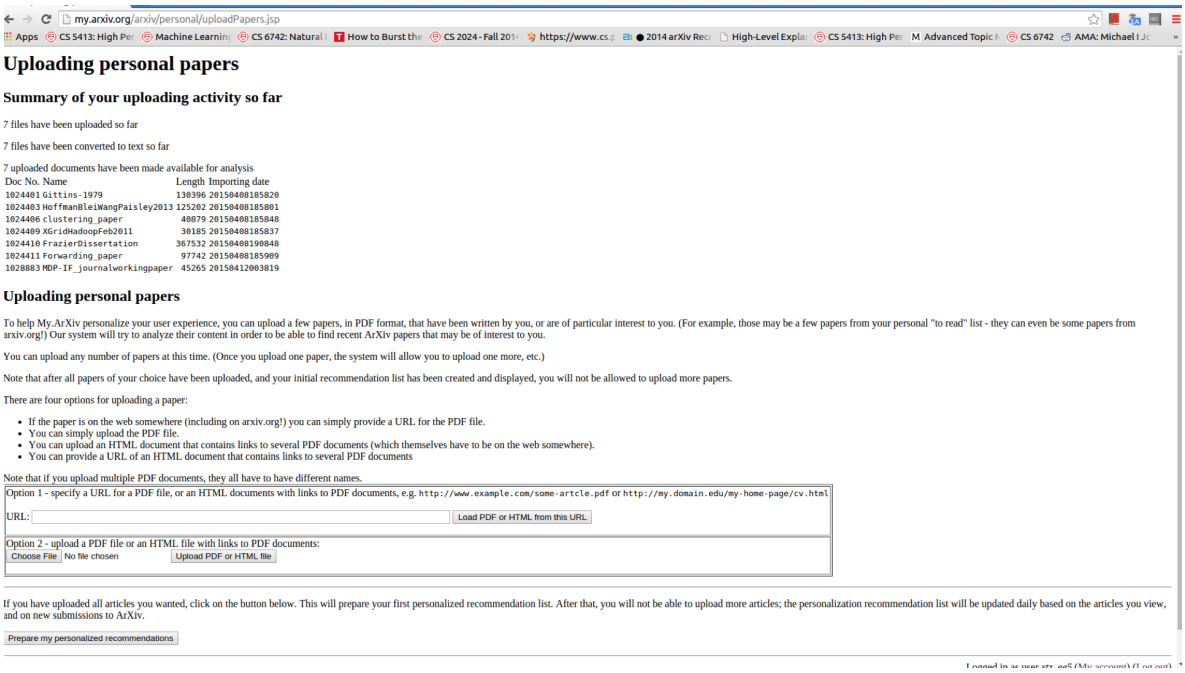


Figure 5.8: Screen shot of a functionality to upload personal papers on my.arXiv.org

analyze to find recent arXiv articles that might be relevant to the user. Afterwards, this information together with the user's interaction history will be used to make recommendation for the user, making this approaching more expansive than the session-based recommendation that only relies on shorter sessions of actions.

Algorithms used in daily recommendations are organized into several "experimental plans". Each my.arXiv user is enrolled into one of these plans, with each experimental plan having its particular method for generating recommendation lists. Supported experimental plans include the following:

- *SET_BASED*: Thorsten Joachim's recommender based on the user's previous activity and document content. This engine is mostly deprecated while its subscribers are rolled over to the refined algorithm, PPP.

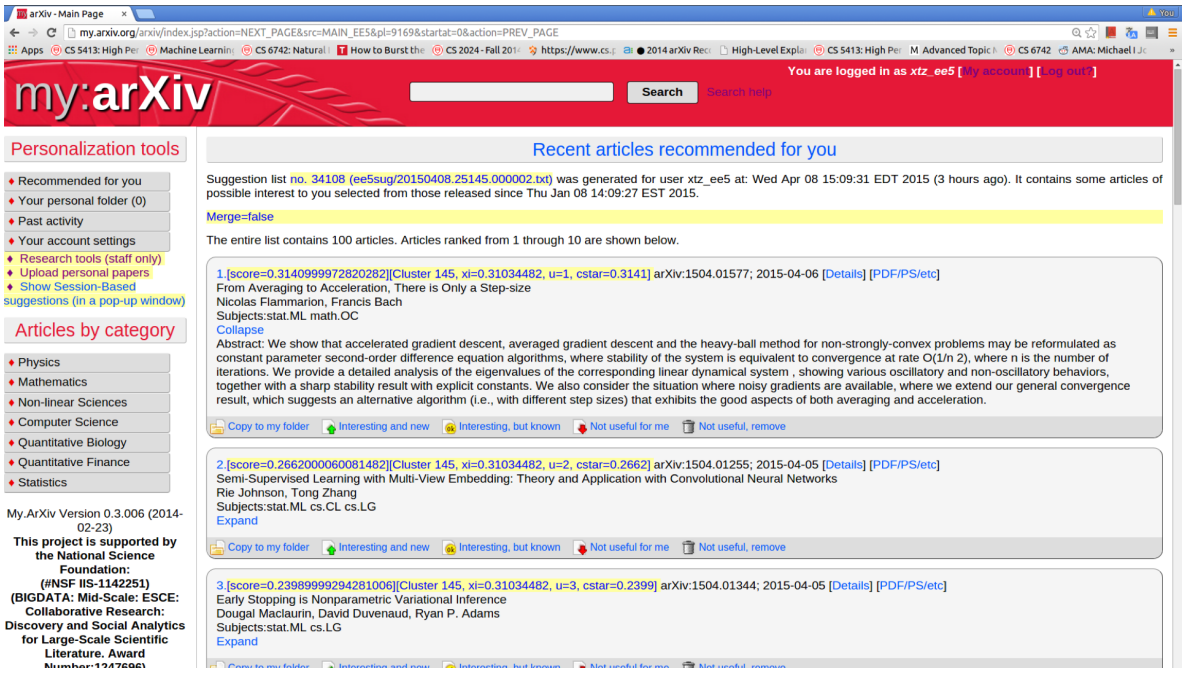


Figure 5.9: Screen shot of EE5 recommendation, using the MDP-IF policy, on my.arXiv.org

- *PPP*: Thorstem Joachim's Perturbed Perceptron Algorithm (3PR) which refines the "SET_BASED" algorithm and provides few safeguards to improves recommendation stability;
- *Exploration Engine version 3 (EE3)*: initial version of Exploration vs. Exploitation. This algorithm is not fully personalized since recommended articles are based on their similarity to older articles in subject categories that users expressed interests. This algorithm is now deprecated.
- *Exploration Engine version 4 (EE4)*: second version of Exploration vs. Exploitation algorithm using the policy developed in Chapter 2. The recommendation is based on cluster level (recommend all from the cluster or none) and is currently replaced by EE5.
- *Exploration Engine version 5 (EE5)*: latest version of Exploration vs. Exploitation algorithm based on the MDP-IF policy developed in Sec-

tion 3.2.1 and the clustering scheme developed in Chapter 4. See a screenshot in Figure 5.9.

5.2.1 Exploration vs. Exploitation 4 (EE4) in my.arXiv.org

This version of the exploration engine is appropriate for small number of users, or even just one user. It has the style of a "forwarding" or "filtering" algorithm, that classifies items into classes, and attempts to learn which classes each user is interested in. See theory of the method in Chapter 2.

The algorithm depends upon a classifier, which classifies each item into one of k different classes. Treating each of the arXiv categories (e.g., astro-ph.GA, stat.ML, cs.AI) separately, the classes of items that belong to the same primary arXiv category are created using an unsupervised learning algorithm (kmeans clustering) on historical data of some collection of document features. Here are steps involved in extracting document features in each arXiv category:

- extract items and meaningful users (who had a lot of interactions with the system) in this arXiv category;
- construct a 0-1 (0 for irrelevant, 1 for relevant) rating matrix, X, from the arXiv logfiles such that each row represents a user and each column represents an item;
- decompose the rating matrix by a singular value decomposition so that X is approximated by $U\Sigma V^*$ where V^* is a $r \times r$ unitary matrix used to represent the document features.

We then use k-means to cluster the document features, V^*, into classes with

a relatively small size, in which the size of classes is roughly proportional to the square root of the total size of items in this arXiv category. Next, we train a classifier, called multi-class Support-Vector-Machine, from item contents (tf-idf presentation of documents) to the derived class labels, and then use it to make prediction the class label for any new item in this category.

Given that the classifier is defined, we show next the ranking algorithm deployed in EE4. First, let us define $c^*(\alpha, \beta; \gamma)$ to be the largest c such that $\mu^*(\alpha + \beta; c, \gamma) < \frac{\alpha}{\alpha+\beta}$, in which $\mu^*(\alpha + \beta; c, \gamma)$ is $\mu^*(\alpha + \beta)$ defined in Equation (2.17) with a unit forwarding cost, c, and a discount factor γ. In other words, $c^*(\alpha, \beta; \gamma)$ is the largest cost, c, such that we would show the document to the user. By a bisection search shown in Algorithm 4, $c^*(\alpha, \beta; \gamma)$ can be computed to an given accuracy $\epsilon > 0$, because $\mu^*(\alpha + \beta; c, \gamma)$ is increasing in c, and $0 \leq c^*(\alpha, \beta; \gamma) \leq 1$.

Algorithm 4 Computation of $c^*(\alpha, \beta; \gamma)$

Require: ϵ (e.g., $\epsilon = 0.0001$)
  Let $L = 0$ be a lower bound on $c^*(\alpha, \beta)$.
  Let $U = 1$ be an upper bound on $c^*(\alpha, \beta)$.
  while $U - L > \epsilon$ **do**
    Let $c = (U + L)/2$
    if $\mu^*(\alpha + \beta; c, \gamma) > \frac{\alpha}{\alpha+\beta}$, let $U = c$.
    if $\mu^*(\alpha + \beta; c, \gamma) \leq \frac{\alpha}{\alpha+\beta}$, let $L = c$.
  end while

For each user in EE4, we present items to the user by sorting the clusters, which users express interests, in a decreasing order of $c^*(\alpha, \beta; \gamma)$ and showing all of the items in that cluster together. From the user's feedback on the presented items, we increment α_x or β_x for any relevant or irrelevant item, respectively, for the corresponding cluster x.

5.2.2 Exploration vs. Exploitation 5 (EE5) in my.arXiv.org

The ranking algorithm in EE4 is designed at the class level, so it suffers a great loss when too many items accumulate in one class on a given day. Thus, in EE5 we use the MDP-IF policy described in Section 3.2.1 to rank items among classes that a user express interests when he/she first registers. In the current setting, we have $\alpha_x = 1$ and $\beta_x = 19$ for each user's interested classes so that all classes are treated equal at the beginning. Additionally, ξ_x for each class is estimated using average number of item arrivals in the class, while γ is set to be 0.99 now.

In the EE5 experimental plan, we use *Content-Augmented Stochastic Block-models* (CASB) described in Chapter 4 to cluster items in each of 140+ arXiv categories. With an exception that some arXiv categories has only one class (due to their small sizes of items), most arXiv categories are chosen to have class sizes, K, in the set $\{5, ..., 10\}$, in which each cluster size is directly proportional to the category's size of items. In each subject category, we train a CASB model to uncover hidden probability vector $p_x \in [0, 1]^F$, $\sum_\ell p_{x,\ell} = 1$ for each item cluster $x \in \{1, ..., K\}$.

Next, we describe a classifier for a new item. For any new item, i, in the arXiv category, let $d_i = [d_{i,1}, ..., d_{i,F}]$ represent item features. Here, each $d_{i,\ell} \in \mathbb{N}$ represents the number of times that words (multigram) from the ℓ^{th} word cluster (word2vec) appear in this document. Then, we determine the nearest cluster for this item by computing the argmax of the log-likelihood function:

$$x^* = \text{argmax}_{x \in \{1,...,K\}} \log \prod_{\ell=1}^{F} p_{x,\ell}^{d_{i,\ell}}.$$

This method is re-trained every few months (e.g., six) to update probability vector associated with each cluster x in each arXiv category.

CHAPTER 6

CONCLUSION

In this thesis we considered variants of the information filtering problem where we face a voluminous stream of items and need to decide sequentially on which batch of items to forward to a user so that the total reward, the number of relevant items shown minus the cost of forwarded items, is maximized. With a focus on limited historical data, we formulated the problem as as a variant of Markov Decision Processes in a Bayesian setting, and then provided computationally tractable algorithms.

Chapter 2 considered the simple information filtering problem with immediate reviews on forwarded items and a unit forwarding cost. With these assumptions, we provided the optimal policy for adaptively forwarding items from multiple categories to maximize the expected total reward via balancing exploration against exploitation. With an independence property, the original k-category problem can be decomposed to an aggregation of k single-category subproblems, avoiding the so-called "curse of dimensionality". Moreover, we show that the optimal policy for each sub-problem is a threshold policy, and that this threshold has intuitive structural properties. To compute this threshold, we provide an approximation to the optimal policy with rigorous error bounds, whose error converges geometrically to 0 as truncation increases. Lastly, results from both idealized Monte Carlo and trace-driven simulations show that this optimal policy provides value in practice.

Inspired by the real operation from arXiv, Chapter 3 generalized the model from Chapter 2 to consider periodic reviews and unknown costs. In the "periodic review" setting, we also derived a similar Bayes-optimal algorithm through

solving a stochastic dynamic program. With the unknown costs, we formulated the problem as a constrained MDP where the total number of item shown is constrained. Due to the curse of dimensionality in this constrained MDP, we then considered a Lagrangian relaxation and provided an index-based policy that ranks items. As shown in the numerical section, our index-based policy outperforms UCB and the pure exploitation policy, and provides magnified benefits in many settings.

In both Chapter 2 and Chapter 3, we assumed there is a pre-processing step that categorizes incoming items. In Chapter 4 we developed and presented one of such categorization methods, called the content-augmented stochastic block-model (CASB), to learn hidden user communities, item cluster, and community-cluster interaction from a probabilistic model of user-item interactions and item content.

The methods described in the thesis have been deployed to my.arXiv.org as one of daily algorithms in recommending new articles. The system will soon be rolled out for real-user testing. The author sincerely hopes that exploration vs. exploitation method could diversify suggested articles to arxiv researchers, reduce their time in finding relevant and interesting articles, and provide them positive user experience.

APPENDIX A

ADDITIONAL PROOFS IN CHAPTER 2

Proof of Theorem 2.2.3

Part 1: we want to show that $\mu^*(m) \leq c$. As in the proof of Lemma 2.2.3, we write the value function in terms of $\mu = \frac{\alpha}{\alpha+\beta}$ and $m = \alpha + \beta$ instead of α, β. First, $\mu V_x(\mu m + 1, (1-\mu)m) + (1-\mu)V_x(\mu m, (1-\mu)m + 1) \geq 0$ since both $V_x(\mu m + 1, (1-\mu)m)$ and $V_x(\mu m, (1-\mu)m + 1)$ are non-negative and any convex combination of two non-negative points is also non-negative. Pick any $\mu \in (c, 1]$, we have $Q_x(\mu m, (1-\mu)m, 1) = \mu - c + \gamma_x \left[\mu V_x(\mu m + 1, (1-\mu)m) + (1-\mu)V_x(\mu m, (1-\mu)m + 1)\right] \geq \mu - c > 0$, so $\mu \in A(m) = \left\{\mu(\alpha, \beta) : \alpha \in (0, \infty), \beta \in (0, \infty), m = \alpha + \beta \text{ and } Q_x(\alpha, \beta, 1) > 0\right\}$. Therefore, $A(m) \supseteq (c, 1]$, which implies $\mu^*(m) = \inf A(m) \leq c$.

Part 2: we want to show that $\mu^*(m_0) \leq \mu^*(m_0+1)$ for any $m_0 > 0$. It is sufficient to show that for each μ, $V_x(\mu m_0, (1-\mu)m_0) \geq V_x(\mu(m_0 + 1), (1-\mu)(m_0 + 1))$. As defined in the proof of Lemma 2.2.3, we first show the statement for a M–step finite–horizon problem where $M - m_0 \geq 1$ is an integer. Fix μ, let $g_m(\mu) \equiv V_x(\mu m, (1-\mu)m, M)$ for any $0 < m \leq M$. We will show $g_m(\mu) \geq g_{m+1}(\mu)$ for all $m \in \{M-1, M-2, ..., m_0\}$ by backward induction. In the base case, $m = M - 1$, we have

$$\begin{aligned} g_{M-1}(\mu) &= \max\left\{0, \mu - c + \gamma_x \left[\mu \cdot g_M\left(\frac{\mu(M-1)+1}{M}\right) + (1-\mu) \cdot g_M\left(\frac{\mu(M-1)}{M}\right)\right]\right\} \\ &\geq \max\left\{0, \mu - c + \gamma_x \cdot g_M(\mu)\right\} = \max\left\{0, \mu - c + \gamma_x[\max\{0, (\mu - c)/(1-\gamma_x)\}]\right\}, \end{aligned}$$

where the inequality holds by Jensen's inequality because $g_M(\mu) = \max\{0, \mu - c\}/(1-\gamma_x)$ is convex in μ. If $\mu - c > 0$, then $g_{M-1}(\mu) \geq \frac{\mu - c}{1-\gamma_x} = g_M(\mu)$. Otherwise, if $\mu - c \leq 0$, then $g_{M-1}(\mu) \geq 0 = g_M(\mu)$. This shows the base case $g_{M-1}(\mu) \geq g_M(\mu)$.

Let $m \in \{M-2, M-3, ..., m_0\}$ and assume $g_m(\mu) \geq g_{m+1}(\mu)$. Next we show that $g_{m-1}(\mu) \geq g_m(\mu)$. There are two cases to consider. First, if $g_m(\mu) = 0$, we have that $g_{m-1}(\mu) \geq g_m(\mu) = 0$. Second, if

$$g_m(\mu) = \mu - c + \gamma_x \left[\mu \cdot g_{m+1}\left(\tfrac{\mu m+1}{m+1}\right) + (1-\mu) \cdot g_{m+1}\left(\tfrac{\mu m}{m+1}\right)\right],$$

then

$$\begin{aligned}
&\frac{g_{m-1}(\mu) - g_m(\mu)}{\gamma_x} \\
\geq\ & \mu \cdot g_m\left(\frac{\mu(m-1)+1}{m}\right) + (1-\mu) \cdot g_m\left(\frac{\mu(m-1)}{m}\right) - \mu \cdot g_{m+1}\left(\frac{\mu m+1}{m+1}\right) - (1-\mu) \cdot g_{m+1}\left(\frac{\mu m}{m+1}\right) \\
\geq\ & \mu \cdot g_m\left(\frac{\mu(m-1)+1}{m}\right) + (1-\mu) \cdot g_m\left(\frac{\mu(m-1)}{m}\right) - \mu \cdot g_m\left(\frac{\mu m+1}{m+1}\right) - (1-\mu) \cdot g_m\left(\frac{\mu m}{m+1}\right),
\end{aligned}$$

in which the first inequality is because g_{m-1} is greater than or equal to its value at forwarding, and the second inequality is due to the induction hypothesis that $g_{m+1}(\mu) \leq g_m(\mu)$. Notice that $\frac{\mu(m-1)}{m} \leq \frac{\mu m}{m+1} \leq \mu \leq \frac{\mu m+1}{m+1} \leq \frac{\mu(m-1)+1}{m}$. Define a function $\hat{g}_m(x)$ by replacing $g_m(x)$ with a linear interpolation between $g_m\left(\frac{\mu m}{m+1}\right)$ and $g_m\left(\frac{\mu m+1}{m+1}\right)$ over the interval between these two points,

$$\hat{g}_m(x) = \begin{cases} g_m(x), & \text{if } x \geq \frac{\mu m+1}{m+1} \text{ or } x \leq \frac{\mu m}{m+1}, \\ \frac{g_m\left(\frac{\mu m+1}{m+1}\right) - g_m\left(\frac{\mu m}{m+1}\right)}{1/(m+1)}\left(x - \frac{\mu m}{m+1}\right) + g_m\left(\frac{\mu m}{m+1}\right), & \text{if } \frac{\mu m}{m+1} \leq x \leq \frac{\mu m+1}{m+1}. \end{cases} \tag{A.1}$$

g_m is convex by Lemma 2.2.3, and so $\hat{g}_m$ is also convex. Moreover, by Jensen's inequality, $\hat{g}_m \geq g_m$. Thus, using this inequality in the first line, and then Jensen's

inequality on $\hat{g}_m$ in the second line,

$$\begin{aligned}(1-\mu)\cdot g_m\left(\frac{\mu(m-1)}{m}\right) \quad &+ \quad \mu\cdot g_m\left(\frac{\mu(m-1)+1}{m}\right) \geq (1-\mu)\cdot \hat{g}_m\left(\frac{\mu(m-1)}{m}\right)+\mu\cdot \hat{g}_m\left(\frac{\mu(m-1)+1}{m}\right)\\ &\geq \quad \hat{g}_m\left((1-\mu)\frac{\mu(m-1)}{m}+\mu\frac{\mu(m-1)+1}{m}\right)\\ &= \quad \hat{g}_m(\mu)=\hat{g}_m\left((1-\mu)\frac{\mu m}{m+1}+\mu\frac{\mu m+1}{m+1}\right)\\ &= \quad \frac{g_m\left(\frac{\mu m+1}{m+1}\right)-g_m\left(\frac{\mu m}{m+1}\right)}{1/(m+1)}\left(\mu-\frac{\mu m}{m+1}\right)+g_m\left(\frac{\mu m}{m+1}\right)\\ &= \quad (1-\mu)\cdot g_m\left(\frac{\mu m}{m+1}\right)+\mu\cdot g_m\left(\frac{\mu m+1}{m+1}\right).\end{aligned}$$

Thus, $g_{m-1}(\mu)\geq g_m(\mu)$. This completes the induction step.

We have shown that $V_x(\mu m_0,(1-\mu)m_0,M)=g_m(\mu)\geq V_x(\mu(m_0+1),(1-\mu)(m_0+1),M)$ for each μ in a M–step finite–horizon problem. Taking the limit as M goes to infinity, we conclude that $V_x(\mu m_0,(1-\mu)m_0)\geq V_x(\mu(m_0+1),(1-\mu)(m_0+1))$. Equivalently, we conclude that $\mu^*(m)\leq\mu^*(m+1)$.

Part 3: we want to show that $\lim_{m\to\infty}\mu^*(m)=c$**.** It is sufficient to show that, for each $\mu<c$, there exists $N>0$ large enough that $V_x(\mu m,(1-\mu)m)=0$ for all $m\geq N$. Let $\alpha=\mu m$, and $\beta=(1-\mu)m$. By definition, $V_x(\alpha,\beta)=\max\{0,\mu-c+\gamma_x[\mu V_x(\alpha+1,\beta)+(1-\mu)V_x(\alpha,\beta+1)]\}$, so if $\mu-c+\gamma_x[\mu V_x(\alpha+1,\beta)+(1-\mu)V_x(\alpha,\beta+1)]\leq 0$, then $V_x(\alpha,\beta)=0$. Using the upper bound from Proposition 2.2.2, $V_x(\alpha,\beta)\leq V_x^U(\alpha,\beta)=\frac{1}{1-\gamma_x}E[\max\{0,\theta_x-c\}|\theta_x\sim\text{Beta}(\alpha,\beta)]$. So,

$$\begin{aligned}&\mu V_x(\alpha+1,\beta)+(1-\mu)V_x(\alpha,\beta+1)\leq \mu V_x^U(\alpha+1,\beta)+(1-\mu)V_x^U(\alpha,\beta+1)\\ &\quad=\mu E[\max\{0,\theta_x-c\}|\theta_x\sim\text{Beta}(\alpha+1,\beta)]+(1-\mu)E[\max\{0,\theta_x-c\}|\theta_x\sim\text{Beta}(\alpha,\beta+1)]\\ &\quad=E[E[\max\{0,\theta_x-c\}|\theta_x\sim\text{Beta}(\alpha,\beta),Y]|\theta_x\sim\text{Beta}(\alpha,\beta)]\\ &\quad=E[\max\{0,\theta_x-c\}|\theta_x\sim\text{Beta}(\alpha,\beta)]=V_x^U(\alpha,\beta),\end{aligned}$$

where the first and fourth equalities are justified through the definition of $V_x^U(\alpha,\beta)$, the second equality rewrites each term in terms of a conditional expectation given $Y|\theta_x \sim \text{Bernoulli}(\theta_x)$, and the third equality is due to the tower property of conditional expectation. Then, recalling that α, β implicitly depend on m,

$$\lim_{m\to\infty} V_x^U(\alpha,\beta) = \lim_{m\to\infty} E[\max\{0,\theta_x - c\}|\theta_x \sim \text{Beta}(\alpha,\beta)] = \max\{0,\mu - c\} = 0.$$

So, $\limsup_{m\to\infty} \mu - c + \gamma_x[\mu V_x(\alpha+1,\beta) + (1-\mu)V_x(\alpha,\beta+1)] \leq \mu - c < 0$. Thus, there exists $N > 0$ such that $\mu - c + \gamma_x(\mu V_x(\alpha+1,\beta) + (1-\mu)V_x(\alpha,\beta+1)] < 0$ for all $m \geq N$. This implies that $\mu^*(m) \geq \mu \ \forall m > N$. Then, $\liminf_{n\to\infty} \mu^*(m) = \lim_{N'\to\infty} \inf\{\mu^*(m) : m \geq N'\} \geq \mu$. Since this is true for all $\mu < c$, we have $\liminf_{m\to\infty} \mu^*(m) \geq c$. Combining with part 1, which showed that $\limsup_{m\to\infty} \mu^*(m) \leq c$, we have that

$$c \leq \liminf_{m\to\infty} \mu^*(m) \leq \limsup_{m\to\infty} \mu^*(m) \leq c.$$

Therefore, the limit exists and $\lim_{m\to\infty} \mu^*(m) = c$.

BIBLIOGRAPHY

Adomavicius, G., A. Tuzhilin. 2005. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on knowledge and data engineering* **17**(6) 734–749.

Agarwal, D., B.C. Chen, P. Elango. 2009. Explore/exploit schemes for web content optimization. *Proceedings of 2009 Ninth IEEE International Conference on Data Mining*. ICDM '09, IEEE Computer Society, Washington, DC, USA, 1–10.

Agarwal, D., B.C. Chen, B. Pang. 2011a. Personalized recommendation of user comments via factor models. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. EMNLP '11, Association for Computational Linguistics, Stroudsburg, PA, USA, 571–582.

Agarwal, D., L. Zhang, R. Mazumder. 2011b. Modeling item–item similarities for personalized recommendations on yahoo! front page. *The Annals of Applied Statistics* **5**(3) 1839–1875.

Agrawal, S., N. Goyal. 2011. Analysis of thompson sampling for the multi-armed bandit problem. *CoRR* **abs/1111.1797**.

Agrawal, S., N. Goyal. 2012. Thompson sampling for contextual bandits with linear payoffs. *CoRR* **abs/1209.3352**.

Airoldi, E.M., D.M. Blei, S.E. Fienberg, E.P. Xing. 2009. Mixed membership stochastic blockmodels. *Advances in Neural Information Processing Systems*. 33–40.

Araman, V.F., R. Caldenty. 2009. Dynamic pricing for perishable products with demand learning. *Operations Research* **57**(5) 1169 – 1188.

arXiv.org. 2014. arxiv.org e-print archive. `http://arxiv.org`. Accessed: 2014-06-30.

Auer, P., N. Cesa-Bianchi, P. Fischer. 2002. Finite-time analysis of the multiarmed bandit problem. *Machine learning* **47**(2) 235–256.

Auer, P., N. Cesa-Bianchi, Y. Freund, R.E. Schapire. 1995. Gambling in a rigged casino: The adversarial multi-armed banditproblem. *Foundations of Computer Science, 1995. Proceedings., 36th Annual Symposium on*. 322–331.

Balabanović, M., Y. Shoham. 1997. Fab: content-based, collaborative recommendation. *Communications of the ACM* .

Basilico, J., T. Hofmann. 2004. Unifying collaborative and content-based filtering. *Twenty-first International Conference on Machine learning - ICML '04*. ACM Press, New York, New York, USA, 9.

Bechhofer, R.E. 1954. A single-sample multiple decision procedure for ranking means of normal populations with known variances. *The Annals of Mathematical Statistics* **25**(1) 16–39.

Bechhofer, R.E., J. Kiefer, M. Sobel. 1968. *Sequential Identification and Ranking Procedures*. University of Chicago Press, Chicago.

Bellman, R. 1956. A problem in the sequential design of experiments. *Sankhy: The Indian Journal of Statistics (1933-1960)* **16**(3/4) pp. 221–229.

Berry, D.A., B. Fristedt. 1985. *Bandit Problems: Sequential Allocation of Experiments*. Chapman & Hall, London.

Besbes, O., A. Zeevi. 2009. Dynamic pricing without knowing the demand

function: Risk bounds and near-optimal algorithms. *Operations Research* **57**(6) 1407–1420.

Blei, D.M., A.Y. Ng, M.I. Jordan. 2003. Latent dirichlet allocation. *The Journal of Machine Learning Research* **3** 993–1022.

Box, G.E.P., W.G. Hunter, J.S. Hunter. 1978. *Statistics for Experimenters: An Introduction to Design, Data Analysis, and Model Building*. John Wiley & Sons, New York.

Callan, J. 1996. Document filtering with inference networks. *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '96, ACM, New York, NY, USA, 262–269.

Cashore, M., X. Zhao, A. Alemi, Y. Liu, P.I. Frazier. 2015. Clustering via content-augumented stochastic blockmodels .

Chapelle, O., T. Joachims, F. Radlinski, Y. Yue. 2012. Large-scale validation and analysis of interleaved search evaluation. *ACM Trans. Inf. Syst.* **30**(1) 6:1–6:41.

Chapelle, O., L. Li. 2011. An empirical evaluation of thompson sampling. *Advances in Neural Information Processing Systems 24*. Curran Associates, Inc., 2249–2257.

Charlin, L., R. Zemel, C. Boutilier. 2011. A framework for optimizing paper matching. *Proceedings of the Twenty-Seventh Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-11)*. AUAI Press, Corvallis, Oregon, 86–95.

Charlin, L., R.S. Zemel. 2013. The toronto paper matching system: An automated paper-reviewer assignment system. *In the ICML Workshop on Peer Reviewing and Publishing Models (PEER)*.

Claypool, M., A. Gokhale, T. Miranda. 1999. Combining content-based and collaborative filters in an online newspaper. *Proceedings of ACM SIGIR workshop on recommender systems.* **60**.

Coates, A., A.Y. Ng. 2012. Learning feature representations with k-means. *Neural Networks: Tricks of the Trade*. Springer, 561–580.

Dai, A.M., C. Olah, Q.V. Le, G.S. Corrado. 2014. Document embedding with paragraph vectors. *NIPS Deep Learning Workshop*.

DeGroot, M.H. 2004. *Optimal Statistical Decisions*. John Wiley & Sons, Hoboken, NJ.

den Boer, A.V., B. Zwart. 2013. Simultaneously learning and optimizing using controlled variance pricing. *Management Science* **60**(3) 770–783.

Ding, X., M.L. Puterman, A. Bisi. 2002. The Censored Newsvendor and the Optimal Acquisition of Information. *Operations Research* **50**(3) 517–527.

Dupret, G.E., B. Piwowarski. 2008. A user browsing model to predict search engine click data from past observations. *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '08, ACM, New York, NY, USA, 331–338.

Easley, D., J. Kleinberg. 2010. *Networks, Crowds, and Markets: Reasoning about a Highly Connected World*. Cambridge University Press, New York, NY, USA.

Frazier, P.I. 2011. *Wiley Encyclopedia of Operations Research and Management Science*, chap. Learning with Dynamic Programming. Wiley, Hoboken, NJ.

Frazier, P.I., W.B. Powell, S. Dayanik. 2008. A knowledge gradient policy for

sequential information collection. *SIAM Journal on Control and Optimization* **47**(5) 2410–2439.

Ginsparg, P. 2011. Arxiv at 20. *Nature* **476** 145–147. doi:10.1038/476145a.

Ginsparg, P. 2014. personal communication.

Gittins, J., K. Glazebrook, R. Weber. 2011. *Multi-Armed Bandit Allocation Indices*. John Wiley & Sons, Ltd.

Gittins, J. C., D. M. Jones. 1974. A dynamic allocation index for the sequential design of experiments. J. Gani, ed., *Progress in Statistics*. North-Holland, Amsterdam, 241–266.

Gittins, J.C. 1979. Bandit processes and dynamic allocation indices. *Journal of the Royal Statistical Society, Series B* 148–177.

Gopalan, P.K., L. Charlin, D. Blei. 2014. Content-based recommendations with poisson factorization. *Advances in Neural Information Processing Systems 27*. Curran Associates, Inc., 3176–3184.

Hartmann. 1991. An improvement on Paulson's procedure for selecting the population with the largest mean from k normal populations with a common unknown variance. *Sequential Analysis* **10**(1-2) 1–16.

Hoffman, M.D., D. Blei, C. Wang, J. Paisley. 2013. Stochastic variational inference. *Journal of Machine Learning Research* 1303–1347.

Hofmann, K., S. Whiteson, M. Rijke. 2013. Balancing exploration and exploitation in listwise and pairwise online learning to rank for information retrieval. *Information Retrieval* **16**(1) 63–90.

Holland, P.W., K.B. Laskey, S. Leinhardt. 1983. Stochastic blockmodels: First steps. *Social networks* **5**(2) 109–137.

Hu, W., P.I. Frazier, J. Xie. 2014. Parallel bayesian policies for finite-horizon multiple comparisons with a known standard. *Proceedings of the 2014 Winter Simulation Conference*. Institute of Electrical and Electronics Engineers, Inc., Piscataway, New Jersey.

INFORMS. 2015. Annual meeting - informs. https://www.informs.org/Attend-a-Conference/Annual-Meeting. Accessed: 2015-02-20.

Jaksch, T., R. Ortner, P. Auer. 2010. Near-optimal regret bounds for reinforcement learning. *Journal of Machine Learning Research (JMLR)* **11** 1563–1600.

Joachims, T. 1998. Text categorization with support vector machines: Learning with many relevant features. *European Conference on Machine Learning (ECML)*. Springer, Berlin, 137–142.

Kaelbling, L.P., M.L. Littman, A.R. Cassandra. 1998. Planning and acting in partially observable stochastic domains. *Artificial Intelligence* **101**(1-2) 99–134.

Katehakis, M. N., A. F. Veinott, Jr. 1987. The multi-armed bandit problem: decomposition and computation. *Mathematics of Operations Research* **12**(2) 262–268.

Kattuman, P., W Yang. 2014. Reinforced random processes in competitive systems. Accessed March 25, 2015.

Kaufmann, E., A. Garivier, T. Paristech. 2012. On bayesian upper confidence bounds for bandit problems. *In AISTATS*.

Kim, Seong-Hee, Barry L. Nelson. 2001. A fully sequential procedure for indifference-zone selection in simulation. *ACM Trans. Model. Comput. Simul.* **11**(3) 251–273. doi:http://doi.acm.org/10.1145/502109.502111.

Lafferty, J., C. Zhai. 2001. Document language models, query models, and risk minimization for information retrieval. *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*. SIGIR '01, ACM, New York, NY, USA, 111–119.

Lai, TL, H. Robbins. 1985. Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics* **6**(1) 4–22.

Langford, J., T. Zhang. 2007. The epoch-greedy algorithm for contextual multi-armed bandits. Citeseer.

Lariviere, M.A., E.L. Porteus. 1999. Stalking Information: Bayesian Inventory Management with Unobserved Lost Sales. *Management Science* **45**(3) 346–363.

Le, Q.V., T. Mikolov. 2014. Distributed representations of sentences and documents. *arXiv preprint arXiv:1405.4053* .

Letham, B., C. Rudin, D. Madigan. 2013. Sequential event prediction. *Mach. Learn.* **93**(2-3) 357–380.

Manning, C.D., P. Raghavan, H. Schtze. 2008. *Introduction to Information Retrieval*. Cambridge University Press, Cambridge, MA, USA.

May, B.C., N. Korda, A. Lee, D.S. Leslie. 2012. Optimistic bayesian sampling in contextual-bandit problems. *Journal of Machine Learning Research* **13** 2069–2106.

Melville, P., R.J. Mooney, R. Nagarajan. 2002. Content-boosted collaborative filtering for improved recommendations. *AAAI/IAAI*. 187–192.

Nallapati, R., A. Ahmed, E.P. Xing, W.W. Cohen. 2008. Joint latent topic models for text and citations. *Proceedings of The Fourteen ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

Nino-Mora, J. 2001. Restless bandits, partial conservation laws and indexability. *Advances in Applied Probability* **33**(1) 76–98.

Paulson, E. 1964. A sequential procedure for selecting the population with the largest mean from k normal populations. *The Annals of Mathematical Statistics* **35**(1) 174–180.

Powell, W.B. 2007. *Approximate Dynamic Programming: Solving the Curses of Dimensionality*. Wiley-Interscience, Hoboken, NJ, USA.

Powell, W.B., A. Ruszczyn'ski, H. Topaloglu. 2004. Learning algorithms for separable approximations of discrete stochastic optimization problems. *Mathematics of Operations Research* **29**(4) 814–836.

Radlinski, F., R. Kleinberg, T. Joachims. 2008. Learning diverse rankings with multi-armed bandits. *Proceedings of the 25th international conference on Machine learning*. ICML '08, ACM, New York, NY, USA, 784–791.

Resnick, S. 2005. *A Probability Path*. Birkhäuser Boston, 5th printing.

Rinott, Y. 1978. On two-stage selection procedures and related probability-inequalities. *Communications in Statistics-Theory and Methods* **7**(8) 799–811.

Rubens, N., D. Kaplan, M. Sugiyama. 2011. Active learning in recommender

systems. P.B. Kantor, F. Ricci, L. Rokach, B. Shapira, eds., *Recommender Systems Handbook*, chap. 23. Springer US, 735–767.

Russo, D., B. Van Roy. 2014. Learning to optimize via posterior sampling. *Mathematics of Operations Research* .

Salter, J., N. Antonopoulos. 2006. Cinemascreen recommender agent: combining collaborative and content-based filtering. *Intelligent Systems, IEEE* **21**(1) 35–41.

Schein, A.I., A. Popescul, R. Popescul, L.H. Ungar, D.M. Pennock. 2002. Methods and metrics for cold-start recommendations. *Proceedings of the 25th Anual International ACM SIGIR on Research and Development in Information Retrieval*. SIGIR '02, ACM, New York, NY, USA, 253–260.

Shaked, M., J.G. Shanthikumar. 2007. *Stochastic Orders*. Springer-Verlag New York.

Shani, G., D. Heckerman, R.I. Brafman. 2005. An mdp-based recommender system. *J. Mach. Learn. Res.* **6** 1265–1295.

Shivaswamy, P., T. Joachims. 2012. Multi-armed bandit problems with history. *Conference on Artificial Intelligence and Statistics (AISTATS)*. 1046–1054.

Sonin, I.M. 2008. A generalized gittins index for a markov chain and its recursive calculation. *Statistics & Probability Letters* **78**(12) 1526 – 1533.

Sutton, R.S., A.G. Barto. 1998. *Introduction to Reinforcement Learning*. MIT Press, Cambridge, MA, USA.

Swisher, J.R., S.H. Jacobson, E. Yücesan. 2003. Discrete-event simulation optimization using ranking, selection, and multiple comparison procedures: A

survey. *ACM Transactions on Modeling and Computer Simulation (TOMACS)* **13**(2) 134–154.

Thompson, W. 1933. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika* **25**(3-4) 285–294.

Van Noorden, R. 2014. The arxiv preprints server htis 1 million articles. *Nature* doi:doi:10.1038/nature.2014.16643.

Wainwright, M.J., M.I. Jordan. 2008. *Graphical Models, Exponential Families, and Variational Inference*, vol. 1. Foundations and Trends in Machine Learning.

Wang, C., D. Blei. 2011. Collaborative topic modeling for recommending scientific articles. *KDD*.

Whittle, P. 1980. Multi-armed bandits and the Gittins index. *Journal of the Royal Statistical Society. Series B (Methodological)* **42**(2) 143–149.

Whittle, P. 1988. Restless bandits: Activity allocation in a changing world. *Journal of applied probability* (special vol. 25A) 287–298.

Xie, J., P.I. Frazier. 2013a. Sequential bayes-optimal policies for multiple comparisons with a known standard. *Operations Research* **61**(5) 1174–1189.

Xie, J., P.I. Frazier. 2013b. Upper bounds on the bayes-optimal procedure for ranking & selection with independent normal priors. *Proceedings of the 2013 Winter Simulation Conference*. Institute of Electrical and Electronics Engineers, Inc., Piscataway, New Jersey, 877–887.

Xu, Z., R. Akella. 2008. A new probabilistic retrieval model based on the dirichlet compound multinomial distribution. *Proceedings of the 31st annual interna-*

tional ACM SIGIR conference on Research and development in information retrieval. SIGIR '08, ACM, New York, NY, USA, 427–434.

Yang, T., R. Jin, Y. Chi, S. Zhu. 2009. Combining link and content for community detection: A discriminative approach. *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM.

Yue, Y., J. Broder, R. Kleinberg, T. Joachims. 2009. The K-armed Dueling Bandits Problem. *Conference on Learning Theory (COLT)*.

Zhang, Y., J. Callan. 2001. Maximum likelihood estimation for filtering thresholds. *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*. SIGIR '01, ACM, New York, NY, USA, 294–302.

Zhang, Y., W. Xu, J. Callan. 2003. Exploration and exploitation in adaptive filtering based on bayesian active learning. *Proceedings of the 20th International Conference*. ICML '03, ACM, Washington, DC, USA, 896–903.

Zhao, X., P. I. Frazier. 2014a. Exploration vs. exploitation in the information filtering problem. *Arxiv preprint arXiv:1407.8186* .

Zhao, X., P.I. Frazier. 2014b. A markov decision process analysis of the cold start problem in bayesian information filtering. *Arxiv preprint arXiv:1410.7852* .

Zhao, X., P.I. Frazier. 2015. A markov decision process analysis of the cold start problem in bayesian information filtering .

Zhou, Y., H. Cheng, J.X. Yu. 2009. Graph clustering based on structural/attribute similarities. *Thirty-fifth International Conference on Very Large Databases - VLDB '09*. ACM.