ANALYZING INTENTIONS FROM BIG DATA TRACES OF HUMAN ACTIVITIES

BY

ASTON ZHANG

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Computer Science
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2017

Urbana, Illinois

Doctoral Committee:

Professor Carl A. Gunter, Chair
Professor Jiawei Han, Chair
Professor ChengXiang Zhai
Professor Ricardo Baeza-Yates, NTENT

Abstract

The rapid growth of big data formed by human activities makes research on intention analysis both challenging and rewarding. We study multifaceted problems in analyzing intentions from big data traces of human activities, and such problems span a range of machine learning, optimization, and security and privacy.

We show that analyzing intentions from industry-scale human activity big data can effectively improve the accuracy of computational models. Specifically, we take query auto-completion as a case study. We identify two hitherto-undiscovered problems: adaptive query auto-completion and mobile query auto-completion. We develop two computational models by analyzing intentions from big data traces of human activities on search interface interactions and on mobile application usage respectively.

Solving the large-scale optimization problems in the proposed query auto-completion models drives deeper studies of the solvers. Hence, we consider the generalized machine learning problem settings and focus on developing lightweight stochastic algorithms as solvers to the large-scale convex optimization problems with theoretical guarantees. For optimizing strongly convex objectives, we design an accelerated stochastic block coordinate descent method with optimal sampling; for optimizing non-strongly convex objectives, we design a stochastic variance reduced alternating direction method of multipliers with the doubling-trick.

Inevitably, human activities are human-centric, thus its research can inform security and privacy. On one hand, intention analysis research from human activities can be motivated from the security perspective. For instance, to reduce false alarms of medical service providers' suspicious accesses to electronic health records, we discover potential *de facto* diagnosis specialties that reflect such providers' genuine and permissible intentions of accessing records with certain diagnoses. On the other hand, we examine the privacy risk in anonymized heterogeneous information networks representing large-scale human activities, such as in social networking. Such data are released for external researchers to improve the prediction accuracy for users' online social networking intentions on the publishers' microblogging site. We show a negative result that makes a compelling argument: privacy must be a central goal for sensitive human activity data publishers.

To My Wife.

Acknowledgments

I would like to thank my advisors Professor Carl Gunter and Professor Jiawei Han for their insightful advice and constant support. For the past five years during my Ph.D. study at UIUC, Professor Gunter and Professor Han have inspired me by their respectful and humble personalities, their enthusiasm and vision in research, and their wisdom and dedication. I am always feeling extremely fortunate and thankful as being their student. There are numerous times when they helped me out of my difficult times. What they have inspired me in the past will keep motivating me, forever.

Next, I would like to thank Professor ChengXiang Zhai and Ricardo Baeza-Yates for providing comments and guidance to this dissertation. Beyond the dissertation, Professor Zhai taught me two courses on information retrieval at UIUC, and Ricardo guided me during my research internship at Yahoo Labs. Besides their solid expertise, their infective energy and attention to details are enormous inspirations.

I have spent wonderful times at Yahoo Labs, Microsoft Research, and Google Research. I am grateful for all my mentors. Working with Amit Goyal, Yi Chang, and Ricardo Baeza-Yates at Yahoo Labs developed my essential engineering and research skills. Xing Xie let me know how to solve new and hard problems under a tight schedule when I visited Microsoft Research. The summer at Google Research enabled me to gain hands-on experiences of deep learning and I wish to thank the team: Luis Garcia-Pueyo, James Wendt, Marc Najork, and Andrei Broder.

I also owe sincere thanks to all my coauthors and collaborators in the past years. I would like to thank all my friends at UIUC and those who graduated from UIUC for making my Ph.D. study enjoyable. I acknowledge UIUC Computer Science Department for making it such a stimulating environment.

Last but not least, I would like to thank my family for always being with me.

Without any of them, this dissertation would be impossible.

Thank you all.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

> The more aware of your intentions and your experiences you become, the more you will be able to connect the two, and the more you will be able to create the experiences of your life consciously. This is the development of mastery. It is the creation of authentic power.
>
> *Gary Zukav*

Human activities are being traced and logged, forming ever-growing big data. Human activity data are valuable for scientists, such as in analyzing human intentions, which may lead to a better understanding of human behavior. The discovered knowledge from such big data traces of human activities enables engineers to develop advanced computational techniques that form the modern life.

How good or bad could a modern life be? Let us start by taking a look at a short story:

> *In 2017, Bob bought a mobile phone of the latest generation for his beloved wife Ada. Ada loved it and installed her favorite applications (apps) on this little device. One day she opened a music app and listened to her favorite band* Maroon 5. *With the beautiful rhythm she sang softly: "Your sugar..." "Yes, please..." Followed by Bob. Ada loved this song and searched for the lyrics. On her device, Ada typed "sugar" on the search bar and a list of suggested query auto-completions (QAC) instantly popped up, such as "sugar cookie recipes" and "sugar glider". She had to compose the full query "sugar lyrics maroon 5". Bob saw that his wife stopped singing but struggled in typing on the small screen. As a manager in a mobile search team, Bob made the anonymized user activity data accessible to external scientists for designing QAC models to better understand users. This is because his team did not possess the expertise in big data analysis. Eventually, effective models were designed with efficient optimization algorithms for analyzing intentions from such big data traces of user activities. Ada was satisfied because the newly-deployed QAC saved more of her keystrokes. However, she was scammed by a phishing URL camouflaged by the*

online-banking interface of her account. In fact, the adversary de-anonymized the released data about Ada and inferred her sensitive information.

Indeed, this is the modern life. On one hand, recent users of commercial search engines on mobile devices saved more than 60% of the keystrokes when submitting English queries by selecting query auto-completion (QAC) suggestions [183]. On the other hand, 8% of some sampled 25 million URLs posted to microblogging sites point to phishing, malware, and scams [54].

1.1 Background and Problems

Nowadays, various forms of human activities take place in digital systems. For instance, when people search on the Web, they can type their queries on the search bar of search engines; microblogging users may follow and mention their interested users, and retweet and comment their interested tweets; medical service providers can access patients' diagnosis information via electronic health record systems. Human activities are often traced and logged, thus human activity data are being generated at a large scale. Taking mobile app usage as an example, on average, every user installs 95 apps on the mobile device and opens 35 unique apps 100 times per day [183].

The big data traces of human activities present both opportunities and challenges, such as in analyzing human intentions. People often intend to do, complete, or achieve something during their interactions with digital devices, or more broadly, with machines. For instance, people may complete queries on the search bar, may access electronic health records of certain categories of diagnoses, or may follow several microblogging users that may be of interest later. If such intentions can be recognized or predicted, human efforts in such interactions with machines may be reduced. For instance, after people just type one or two prefix characters, the search engine may auto-complete the much longer desired queries; physicians may access electronic health records with certain categories of diagnoses without the need for resolving potential false alarms if their actual related specialties can be recognized and assigned to them, which reflect their genuine intentions in accessing records of certain diagnoses; microblogging users may leverage the recommendation systems that predict their networking intentions to expand their online social networks with ease.

Therefore, what could be the possible ways of analyzing intentions from human activity big data? What could be the algorithmic challenges? How could such human-centric research inform security and privacy? To illustrate, below we list and describe important research problems of analyzing intentions from big data traces of human activities:

- From a practical perspective, it is crucial to study big data traces of human activities to have a better understanding of human intentions. For instance, a better understanding of human intentions can lead to more effective computational models that may save humans' efforts in their interactions with machines. Specifically, as illustrated in the aforementioned short story in 2017, a more accurate QAC model can reduce users' typing efforts, especially on relatively small screens of many mobile devices. In fact, there are various related problems in practice, such as discovering and selecting helpful signals from human activities, modeling such signals for classification, prediction, and ranking tasks, and analyzing the experimental results to gain further insights on user intention understanding.

- From a methodological perspective, it is challenging to discover knowledge such as human intentions from big data. Many computational models, such as various machine learning models for empirical risk minimization, attempt to optimize pre-defined objectives with respect to all the available data. Although it is possible to leverage distributed computing when multiple computing nodes are accessible, such computing resources may not always be available. This drives the development of efficient stochastic optimization algorithms to be employed by computational models on a single machine, or, on individual computing nodes of a grid. In contrast to batch (determined) optimization algorithms, a stochastic optimization algorithm enjoys a much lighter per-iteration computational cost because the per-iteration computation is based on a randomly sampled data subset.

- As another important practical point of view, due to the human-centric nature, intention analysis from human activities can be closely related to security and privacy. On one hand, intention analysis from human activities can be driven by the growing demand of security, such as the need for more accurate identification of suspicious access to sensitive data like electronic health records. On the other hand, human activity data must always be handled with care because of the privacy implications of this type of research. For instance, in the $1 million Netflix Prize, the rental firm published the data as a challenge to the world's researchers to improve its movie-recommendation algorithms. However, individual users in the anonymized data set were re-identified with the matching of film ratings on the Internet Movie Database (IMDb) [115].

1.2 Dissertation Statement

We study multifaceted problems of analyzing intentions from big data traces of human activities. Specifically, this dissertation offers evidence for the following statement:

Dissertation Statement: Analyzing intentions from big data traces of human activities can improve the accuracy of computational models, such as for query auto-completion; can be faster with an appropriate algorithm design, such as with variance reduction techniques; and can inform security and privacy, such as in the heterogeneous information network setting.

1.3 Dissertation Contributions

To support the dissertation statement in Section 1.2, we study the problems listed and discussed in Section 1.1 and make the following contributions.

- We show that analyzing intentions from industry-scale human activity data can effectively improve the accuracy of computational models. Specifically, we take QAC as a data mining case study. After identifying two hitherto-undiscovered problems adaptive QAC (Chapter 2) and mobile QAC (Chapter 3), we develop two computational QAC models by analyzing query intentions from large-scale human activities on search interface interactions and on mobile app usage respectively. Both newly proposed models consistently and significantly outperform the baseline models in various accuracy measures.

- The optimization problems in both of these proposed QAC models share a common property: they both minimize a finite sum of convex functions. Hence, we generalize two problem settings: strongly convex objectives and non-strongly convex objectives. Considering the big data challenge, we focus on developing lightweight stochastic algorithms as solvers to the large-scale convex optimization problems with theoretical guarantees. For optimizing strongly convex objectives, we propose ASBCD, an accelerated stochastic block coordinate descent method with optimal sampling (Chapter 4); for optimizing non-strongly convex objectives, we propose SVR-ADMM-D, a stochastic variance reduced alternating direction method of multipliers with the doubling-trick (Chapter 5). Both proposed algorithms are based on variance reduction techniques.

- We highlight that human activities are human-centric. On one hand, such data mining research can be motivated from the security perspective. To reduce false alarms of suspicious electronic health record access activity detection, we invoke machine learning to discover potential *de facto* diagnosis specialties that exist in practice regardless of the medical specialty codes to reflect health service providers' genuine and permissible intentions in accessing records with certain diagnoses (Chapter 6). Specifically, the setting of heterogeneous information networks is considered for a proposed discovery method. On the other hand, we examine the privacy risk in anonymized heterogeneous information networks

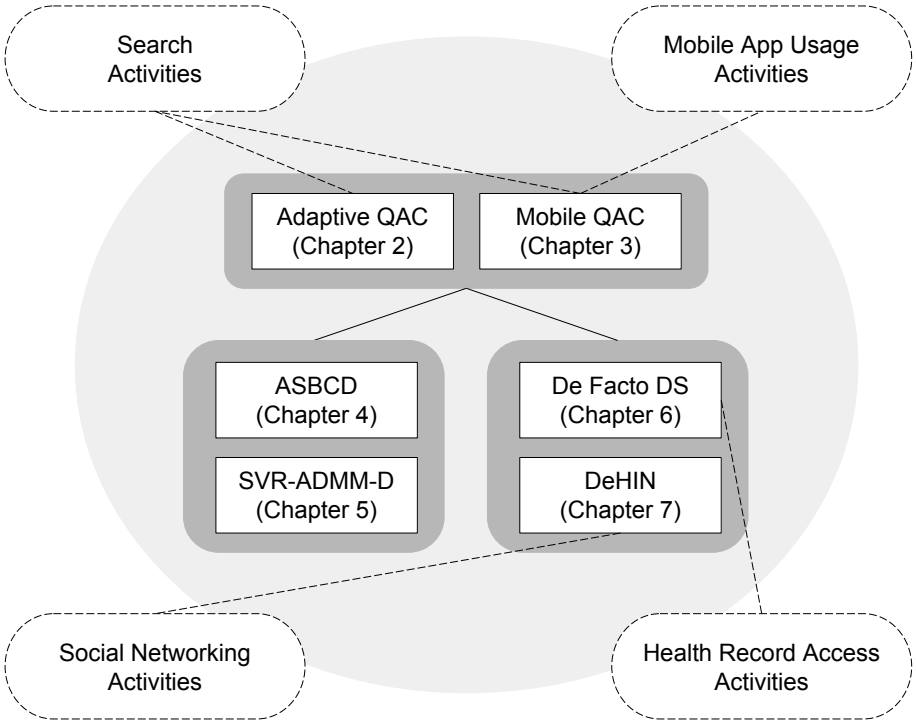


Figure 1.1: Connections among the dissertation contributions and chapters with the studied types of human activity data.

representing large-scale human activities, such as in social networking. We show a de-anonymization algorithm DeHIN over anonymized data that are released for online networking intention prediction so as to improve the data publishers' in-house recommendation system. This research makes a compelling argument: privacy must be a central goal for sensitive human activity data publishers, especially in the heterogeneous information network setting (Chapter 7).

1.4 Organization of the Dissertation

Chapters in this dissertation span a range of machine learning, optimization, and security and privacy, describing a variety of problems and methods in analyzing intentions from big data traces of human activities, such as activities in search, mobile app usage, health record access, and social networking.

Figure 1.1 elucidates the connections among each of the dissertation contributions and chapters that are described in Section 1.3 with the studied types of human activity data. Note that big data traces of human activities are often in the form of logs. Adaptive QAC (Chapter 2) and mobile QAC (Chapter 3) leverage query intention analysis from large-scale search and mobile app usage activity logs to improve the accuracy of computational models for the QAC problem. Such models fit in more generalized machine learning prob-

lem settings where two fast solvers using variance reduction techniques, stochastic optimization algorithms ASBCD (Chapter 4) and SVR-ADMM-D (Chapter 5), are proposed with theoretical justifications. In the context of the other types of traced human activities that can be represented by heterogenous information networks, such as health record access logs and social networking activity logs, *de facto* diagnosis specialty (Chapter 6) and de-anonymization algorithm DeHIN (Chapter 7) are studied to reiterate the human-centric nature of such human intention and activity research from the security and privacy perspective. Specifically, research on intentions from human activities, such as those in Chapters 2 and 3, can be motivated from the security perspective, but must be conducted with care due to potential privacy leakage in the released data even after anonymization.

Below we provide an overview of the subsequent major chapters to further illustrate the dissertation contributions as described in Section 1.3 and elucidated in Figure 1.1.

Chapter 2 (Intention Analysis from Human Activities on Search Interface Interactions) studies a specific problem of intention analysis from big data traces of human activities: the QAC problem with query intention prediction. QAC facilitates user query composition by suggesting queries given query prefix inputs. In fact, users' preference of queries can be inferred during user-QAC interactions, such as dwelling on suggestion lists for a long time without selecting query suggestions ranked at the top. However, the wealth of such implicit negative feedback has not been exploited for designing QAC models. Most existing QAC models rank suggested queries for given prefixes based on certain relevance scores. We take the initiative towards studying implicit negative feedback during user-QAC interactions. This motivates re-designing QAC in the more general "(static) relevance–(adaptive) implicit negative feedback" framework. We propose a novel adaptive model adaQAC that adapts query auto-completion to users' implicit negative feedback towards unselected query suggestions. We collect user-QAC interaction data and perform large-scale experiments. Empirical results show that implicit negative feedback significantly and consistently boosts the accuracy of the investigated static QAC models that only rely on relevance scores. Our work compellingly makes a key point: QAC should be designed in a more general framework for adapting to implicit negative feedback.

Chapter 3 (Intention Analysis from Human Activities on Mobile Application Usage) goes on to study QAC. Specifically, we study the new mobile QAC problem to exploit mobile devices' exclusive signals, such as those related to mobile apps. We propose AppAware, a novel QAC model using installed app and recently opened app signals to suggest queries for matching input prefixes on mobile devices. To overcome the challenge of noisy and voluminous signals, AppAware optimizes composite objectives with a lighter processing cost at a linear rate of convergence. We conduct experiments on a large commercial data

set of mobile queries and apps. Installed app and recently opened app signals consistently and significantly boost the accuracy of various baseline QAC models on mobile devices.

Chapter 4 (Stochastic Optimization for Big Data Analysis: Strongly Convex Objectives) considers the solver to the optimization problems in Chapters 2 and 3 where the objectives are strongly convex objectives. To be precise, we study the composite minimization problem where the objective function is the sum of two convex functions: one is the sum of a finite number of strongly convex and smooth functions, and the other is a general convex function that is non-differentiable. Specifically, we consider the case where the non-differentiable function is block separable and admits a simple proximal mapping for each block. This type of composite optimization is common in many data mining and machine learning problems, and can be solved by block coordinate descent algorithms. We propose an accelerated stochastic block coordinate descent (ASBCD) algorithm, which incorporates the incrementally averaged partial derivative into the stochastic partial derivative (variance reduction technique) and exploits optimal sampling. We prove that ASBCD attains a linear rate of convergence. In contrast to uniform sampling, we reveal that the optimal non-uniform sampling can be employed to achieve a lower iteration complexity. Experimental results on different large-scale real data sets support our theory.

Chapter 5 (Stochastic Optimization for Big Data Analysis: Non-Strongly Convex Objectives) further studies the same problem setting as depicted in Chapter 4 except for the fact that the smooth functions can be non-strongly convex, which is a more relaxed constraint than strong convexity. We propose a stochastic variance reduced alternating direction method of multipliers with the doubling-trick: SVR-ADMM-D. SVR-ADMM-D is a more efficient variant of the ADMM algorithm, which is scalable when multiple computational nodes are available to tackle the big data challenge [5]. The proposed algorithm leverages past variable values to progressively reduce the variance of the gradient estimator. The algorithm also incorporates the doubling-trick to enable itself to be a theoretically-sound anytime algorithm: it can be interrupted anytime while the training error converges to zero with increasing iterations. Experimental results on different real data sets demonstrate that SVR-ADMM-D converges faster than several baseline stochastic alternating direction methods of multipliers.

Chapter 6 (Intention Analysis from Human Activities as Motivated by Security) studies an intention analysis problem from medical service providers' electronic health record access activities as motivated by the security perspective. In health care institutions, medical specialty information may be lacking or inaccurate. As a result, false alarms of suspicious accesses to electronic health records might be raised. We think that medical service providers can save their efforts in resolving such false alarms if their actual related specialties can be recognized and assigned to them. In fact, diagnosis histories offer

information on which medical specialties may exist in practice, regardless of whether they have official codes. We refer to such specialties that are predicted with high certainty by diagnosis histories as *de facto* diagnosis specialties. Since the false alarms of suspicious accesses to electronic health records may be due to the lacking or inaccurate medical specialty information, we aim to discover *de facto* diagnosis specialties, which reflect medical service providers' genuine and permissible intentions in accessing electronic health records with certain diagnoses. The problem is studied under a general discovery–evaluation framework. Specifically, we employ a semi-supervised learning model analyzing heterogeneous information networks and an unsupervised learning method for discovery. We further employ four supervised learning models for evaluation. We use one year of diagnosis histories from a major medical center, which consists of two data sets: one is fine-grained and the other is general. The semi-supervised learning model discovers a specialty for *Breast Cancer* on the fine-grained data set; while the unsupervised learning method confirms this discovery and suggests another specialty for *Obesity* on the larger general data set. The evaluation results reinforce that these two specialties can be recognized accurately by supervised learning models in comparison with 12 common diagnosis specialties defined by the Health Care Provider Taxonomy Code Set.

Chapter 7 (Privacy Risk in Anonymized Big Data Traces of Human Activities) studies the privacy risk in anonymized big data traces of human activities that are released for external intention analysis research. As an example, *t.qq.com* released its anonymized users' profile, social interaction, and recommendation log data in KDD Cup 2012 to call for recommendation algorithms. The goal is to improve the prediction accuracy for users' online networking intentions on *t.qq.com*. Specifically, the online networking intention prediction task involves predicting whether or not a user will follow an item (person, organization, or group) that has been recommended to the user. Since the entities (users and so on) and edges (links among entities) are of multiple types, the released social network is a *heterogeneous information network*. Prior work has shown how privacy can be compromised in homogeneous information networks by the use of specific types of graph patterns. We show how the extra information derived from heterogeneity can be used to relax these assumptions. To characterize and demonstrate this added threat, we formally define privacy risk in an anonymized heterogeneous information network to identify the vulnerability in the possible way such data are released, and further present a new de-anonymization attack that exploits the vulnerability. Our attack successfully de-anonymized most individuals involved in the data—for an anonymized 1,000-user *t.qq.com* network of density 0.01, the attack precision is over 90% with a 2.3-million-user auxiliary network.

Next, we present each of the main contributions of the dissertation.

Chapter 2

Intention Analysis from Human Activities on Search Interface Interactions

In this chapter, we study a specific problem of intention analysis from big data traces of human activities: the query auto-completion (QAC) problem with query intention prediction. QAC facilitates user query composition by suggesting queries given query prefix inputs. In fact, users' preference of queries can be inferred during user-QAC interactions, such as dwelling on suggestion lists for a long time without selecting query suggestions ranked at the top. However, the wealth of such implicit negative feedback has not been exploited for designing QAC models. Most existing QAC models rank suggested queries for given prefixes based on certain relevance scores. We take the initiative towards studying implicit negative feedback during user-QAC interactions. This motivates re-designing QAC in the more general "(static) relevance–(adaptive) implicit negative feedback" framework. We propose a novel adaptive model adaQAC that adapts query auto-completion to users' implicit negative feedback towards unselected query suggestions. We collect user-QAC interaction data and perform large-scale experiments. Empirical results show that implicit negative feedback significantly and consistently boosts the accuracy of the investigated static QAC models that only rely on relevance scores. Our work compellingly makes a key point: QAC should be designed in a more general framework for adapting to implicit negative feedback.

2.1 Introduction

Query auto-completion (QAC) helps user query composition by suggesting queries given prefixes. As illustrated in Figure 2.1, upon a user's keystroke, QAC displays a *suggestion list* (or *list*) below the current *prefix*. We refer to queries in a suggestion list as *suggested queries* or *query suggestions*. A user can select to submit a suggested query; a user can also submit a query without selecting query suggestions. In 2014, global users of Yahoo! Search saved more than 50% keystrokes when submitting English queries by selecting suggestions of QAC.

Typically, a user *favors* and submits a query if it reflects the user's query intent in a query composition. However, predicting query intent is challenging. Many of the recently proposed QAC models rank a list of

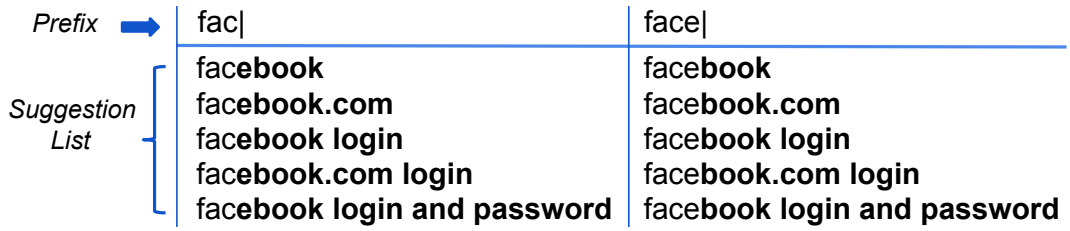


Figure 2.1: A commercial search engine QAC. Given prefixes "fac" and "face", popular "facebook"-related queries are suggested to users after being ranked by certain relevance scores.

suggested queries for each prefix based on different relevance scores, such as popularity-based QAC (using historical query frequency counts) [10], time-based QAC (using time information) [142, 168], context-based QAC (using previous query information of users) [10], personalized QAC (using user profile information) [141], time-and-context-based QAC (using both time and previous query information of users) [20].

The aforementioned models use different relevance features but do not fully exploit user-QAC interactions, such as users' dwell time on suggestion lists and ranked positions of suggested queries by QAC. When users do not select query suggestions at keystrokes of compositions, users implicitly express negative feedback to these queries. Hence at such keystrokes, the user-QAC interaction information is users' *implicit negative feedback* to unselected queries. We aim at complementing relevance features with implicit negative feedback to improve the existing QAC models.

We start with a motivating example.

Motivating Example: Consider a user who wants to query Apple Inc.'s "facetime" with a popularity-based QAC [10]. When the user types "fac", "facebook" is ranked at the top in the suggestion list because it is most popular in historical query logs. The user dwells for a long time to examine the suggested query "facebook" but does not select it because it is not "facetime". However, in the next keystroke "e", popularity-based QAC still makes "facebook" top in the list because it is still the most popular query that matches the prefix "face". Figure 2.1 depicts our interactions with a commercial search engine QAC known to depend on relevance scores only.

Here the user implicitly expresses negative feedback to "facebook": "facebook" is the top query suggestion, and the user dwells on the suggestion list for a long time without selecting this query. Hence, based on such implicit negative feedback, the user may not favor this unselected query. Can QAC be more accurate and demote "facebook" properly given the prefix "face"?

Our Approach: To the best of our knowledge, no existing QAC adapts its ranking of query suggestions to implicit negative feedback. We refer to a QAC model as **static QAC**, if its ranking of suggested queries

does not adapt to implicit negative feedback in a query composition. Examples include popularity-based QAC, time-based QAC, and context-based QAC.

We go beyond static QAC by designing QAC in the new and more general "(static) relevance–(adaptive) implicit negative feedback" framework. In this framework, we propose a novel adaQAC model that adapts QAC to implicit negative feedback. adaQAC reuses the relevance scores of queries from static QAC to pre-index top-N queries. In a single query composition, adaQAC re-ranks these N queries at every keystroke based on users' implicit negative feedback. Personalized learning for every different user with batch inference is employed by adaQAC, and adaQAC can be extended by un-personalized learning and online inference.

Our Contributions: This work has many distinctions from related research in QAC, negative feedback, and dynamic information retrieval; we present detailed discussions on such distinctions in Section 2.5. Our contributions are summarized as follows.

• To the best of our knowledge, this is the first study on implicit negative feedback in user-QAC interactions. We find that the strength of implicit negative feedback to unselected query suggestions can be inferred, and a simple model fails (Section 2.2).

• We go beyond static QAC under a general "(static) relevance–(adaptive) implicit negative feedback" framework: we propose a novel adaQAC model that adapts QAC to implicit negative feedback using personalized learning with batch inference, including un-personalized learning and online inference extensions (Section 2.3).

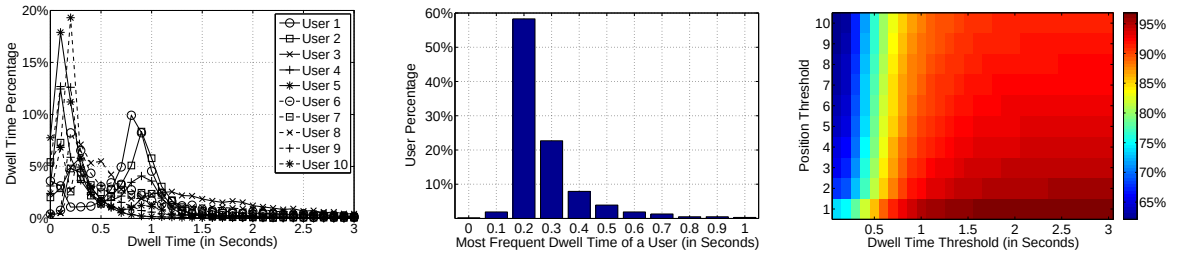
• We collect user-QAC interaction data from a commercial search engine and perform large-scale experiments. We show that implicit negative feedback significantly and consistently boosts the accuracy of the investigated static QAC models (Section 2.4).

2.2 Implicit Negative Feedback in User-QAC Interactions

We study QAC log data from a commercial search engine and discuss several motivational observations on implicit negative feedback in user-QAC interactions.

Terminology: In general, on search engines, queries are *submitted* upon users' *selection* from suggestion lists. Below are the other used terms.

Query composition (Composition): The duration of composing and submitting a single query. It starts from the keystroke of a new query's first character, or from the keystroke starting to edit a previous query. It ends when a query is submitted.



(a) Distributions of dwell time from 10 randomly sampled users from QAC log data.

(b) Histogram of the most frequent dwell time of a user. Bins with a user percentage below 0.25% are omitted.

(c) Percentage of the unselected query suggestions that are not submitted by users.

Figure 2.2: Dwell time and position study. In (a) and (b), Value t at the horizontal axis corresponds to the dwell time bin $[t, t + 0.1)$.
(a) The two peak clusters imply two broad groups of users in the figure: User 1 and 2 generally type slower than the rest;
(b) The distribution shows that different users may have different typing speed;
(c) The percentage varies with different combinations of dwell time and position thresholds. Red color (wide on the right) corresponds to a higher percentage while blue color (narrow on the left) corresponds to a lower percentage. With a longer dwell time and a higher position, the likelihood that an unselected query suggestion will not be submitted by users at the end of query compositions is higher.

Dwell time: The time dwelled on a suggestion list. It is the time gap between two immediate keystrokes in a query composition.

Position: The ranked position of a query in a suggestion list by QAC. Position 1 means being ranked highest or at the top, while 10 corresponds to being ranked lowest or at the bottom.

QAC log data: Our collected user-QAC interaction data from Yahoo! Search. There are 2,932,035 query compositions via desktops and they are sampled over five months in 2014. A composition has the prefixes, timestamps and suggested queries of every keystroke, and the submitted query. More details of the data are in Section 2.4.1. Due to the proprietary nature of the data, some details are omitted in data descriptions and figures.

2.2.1 Implicit Negative Feedback

Typically, a user favors and submit a query that reflects the user's query intent in a query composition. We make the following assumption.

Assumption 2.2.1 *In a query composition, a user submits a query if and only if the user favors it.*

When a suggestion list is displayed, a user may examine or ignore a suggested query [89]. If a user ignores and does not select a suggested query, whether the user favors this query is unknown. If a user examines a suggested query but does not select it, there are two usual cases: (1) the user does not favor it; (2) the

user still favors the suggestion but the user thinks selecting it from the list is less convenient than typing. In spite of possibly complicated cases, under Assumption 2.2.1 we make the following assumption.

Assumption 2.2.2 *In a query composition, suppose a user favors a suggested query. For the user, the likelihood of selecting this query is proportional to the likelihood of examining the query.*

Suppose a user examines a suggested query in a composition with a higher likelihood. From Assumption 2.2.2, if the user favors the query, the user selects it with a higher likelihood. Otherwise, if the user does not select a suggested query after examining the query, it hints that the user may not favor this query. Under Assumption 2.2.1, this user may not submit this unfavored query in the composition. Hence, the examined but unselected query may be demoted at the subsequent keystrokes in the same composition; it allows the user's favored query to rank higher in the composition.

Therefore, in a composition when a user does not select a suggested query, it may be helpful to know whether the user examines the unselected query. In other words, if the user examines an unselected query with a higher likelihood, this query may be demoted more heavily at the subsequent keystrokes of the composition.

For an unselected query suggestion, although whether a user examines it, is not observed, user-QAC interactions can be observed. Such interaction information includes user behavior (dwell time) and settings (position) that are observed during the interactions.

Implicit negative feedback from a user to an unselected query suggestion is observed user-QAC interaction information, when the query is suggested to the user upon keystrokes of a composition. In other words, a user can implicitly express negative feedback to an unselected query "facebook": "facebook" is the top query suggestion, and the user dwells on the list for long without selecting it.

We claim that implicit negative feedback can be strong or weak, and its *strength* cannot be directly observed thus has to be inferred. The properly inferred implicit negative feedback strength may be used to properly demote unselected query suggestions. Recall the discussion that "if the user examines an unselected query with a higher likelihood, this query may be demoted more heavily". Some implicit negative feedback may indicate the likelihood of a user's examination of an unselected query suggestion. Hence, such feedback is of interest. Important examples are dwell time and position.

2.2.2 Dwell Time

If a user dwells on a suggestion list for a longer time, the user may have more time to carefully examine the suggested queries.

On the other hand, if a user dwells for a shorter time, more likely the suggested queries are ignored; thus, even if these queries are unselected, whether the user favors them is unknown.

Figure 2.2(a) elucidates the distributions of the 0.1-second dwell time bin between 0 and 3.1 seconds of 10 randomly sampled users from QAC log data[1]. Dwell time t (in seconds) falls in the bin $[t, t + 0.1)$. As the peak shows the most frequent dwell time bin of a user, it may suggest the user's comfortable typing speed: if the peak falls in the bin of a longer dwell time, the user's general typing speed is slower. The observed heavy-tails of the distributions manifest that longer dwell time is generally rarer, and the peak can characterize the user's typing speed. Thus, in Figure 2.2(a), the two peak clusters may imply two broad groups of users: User 1 and 2 generally type slower than the rest.

Figure 2.2(b) zooms out from 10 users' dwell time distributions to all the users' implied comfortable typing speed with a distribution for dwell time of the peaks in Figure 2.2(a). It demonstrates that different users may have different typing speed. Hence, inference of implicit negative feedback strength by dwell time should be personalized.

2.2.3 Dwell Time and Position as Implicit Negative Feedback

We study dwell time and position of unselected query suggestions that are not submitted by users.

The suggested queries at all the keystrokes in query compositions are collected. Then, suggested queries at the final keystrokes in query compositions are excluded because users may select a suggested query at the final keystroke: only the percentage of unselected queries that are not submitted by users is of interest.

Suppose a dwell time threshold T_{DT} and a position threshold T_P are set up. Consider all the suggested queries $\mathbf{Q}(T_{DT}, T_P)$ that are, both in the list that is dwelled for no shorter than T_{DT}, and, ranked at positions no lower than T_P (dwell time $\geq T_{DT}$ and position $\leq T_P$). Given T_{DT} and T_P, $\forall q \in \mathbf{Q}(T_{DT}, T_P)$, the percentage of occurrences of q that are not submitted by users at the end of query compositions is recorded. The recorded results are illustrated in Figure 2.2(c), with 300 different combinations of T_{DT} and T_P values, where $T_{DT} \in \{0.1, 0.2, \ldots, 3.0\}$ and $T_P \in \{1, 2, \ldots, 10\}$.

Recall Assumption 2.2.1 that a user submits the favored query in a composition. The percentage of users' unselected query suggestions that are not favored by them, can be interpreted by the corresponding color in Figure 2.2(c). As discussed in Section 2.2.1, implicit negative feedback strength may indicate how to demote unselected queries. For a more accurate QAC, the demotion should properly reflect the likelihood of not favoring or submitting an unselected query: such likelihood is higher with a longer dwell time and a

[1]Binning masks details of the data for its proprietary nature.

higher position, as shown in Figure 2.2(c). Thus, the results in Figure 2.2(c) support the hypothesis that dwell time and position are important to infer the strength of implicit negative feedback.

From Figure 2.2(c), when a position threshold T_P is fixed, a dwell time threshold T_{DT} better differentiates the likelihood of not favoring or submitting an unselected query, when $0 < T_{DT} < 1$. This agrees with the results in Figure 2.2(a)—2.2(b) that, longer dwell time is generally rarer.

2.2.4 Filtering Queries by Thresholds Fails

Following the findings in Figure 2.2(c), it is tempting to extend an existing QAC model by filtering out all the suggested queries based on dwell time and position thresholds. Thus, we set up a baseline model **Filtering QAC** to filter out all the suggested queries by using fixed dwell time and position thresholds in the subsequent keystrokes of a query composition. For instance, for $T_{DT} = 2$ and $T_P = 3$, any previously suggested queries with positions higher than or equal to 2 and dwell time longer than or equal to 3 seconds are not suggested anymore in the subsequent keystrokes of the same query composition. To ensure a higher ranking accuracy, the results of Filtering QAC are tuned among $T_{DT} \in \{0.1, 0.2, \ldots, 3.0\}$ and $T_P \in \{1, 2, \ldots, 10\}$.

However, experimental results (Section 2.4.2) show this simple model fails to significantly boost the static QAC models.

2.3 Adaptive Query Auto-Completion

Motivated by the findings from large commercial search engine QAC log data in Section 2.2, we propose a novel **adaQAC** model that adapts query auto-completion to implicit negative feedback.

2.3.1 Method Overview

We describe the system design of adaQAC to rank the suggested queries for a given prefix. A toy example with two queries "facebook" and "facetime" that match prefixes "fac" and "face" at top positions is used to illustrate the idea. Figure 2.3 explains the system design and data flow of adaQAC: it has two stages.

Stage 1 (Pre-indexing): For a given prefix, top-N query suggestions with the highest relevance scores of static QAC are pre-indexed: the higher score, the higher position. In Figure 2.3, for the prefix "face", the top-2 ($N = 2$) queries "facebook" and "facetime" are pre-indexed by static QAC based on the historical query frequency counts.

Stage 2 (Re-ranking): adaQAC re-ranks these top-N queries based on the implicit negative feedback strength inferred from user-QAC interaction information in the same composition. To illustrate Stage 2,

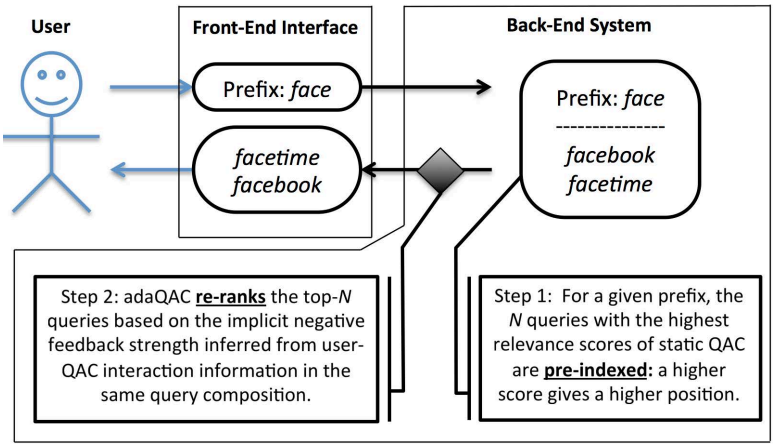


Figure 2.3: The system design and data flow of adaQAC

upon a keystroke "e" following the prefix "fac" from a user, the front-end interface takes the current prefix "face" as an input and immediately fetches the pre-indexed queries "facebook" and "facetime". Suppose when "facebook" was ranked highest in the suggestion list at the prefix "fac", the user dwells for a long time but does not select it. With this observation, suppose adaQAC is able to infer the user's implicit negative feedback strength. Thus, adaQAC updates the ranking score of "facebook" and re-ranks the top 2 ($N = 2$) queries "facebook" and "facetime". With re-ranking, "facetime" is now at Position 1, after "facebook" is demoted to Position 2.

The number of the pre-indexed top queries N can be set to a small positive integer in a production, such as 10 in our experiments. With a small constant value N, sorting N queries based on the updated ranking scores can be achieved in constant time [30].

2.3.2 "(Static) Relevance–(Adaptive) Implicit Negative Feedback" Framework

We highlight that, adaQAC is designed in a more general **"(static) relevance–(adaptive) implicit negative feedback" framework**. The "relevance" component is the relevance score of a query for a user that can be obtained from an existing static QAC model, such as popularity-based QAC; the other "implicit negative feedback" component adapts QAC to implicit negative feedback.

The "(static) relevance–(adaptive) implicit negative feedback" framework is more general for both reusing existing static QAC research and adapting QAC to the newly discovered implicit negative feedback. In this framework, adaQAC is not constrained to employ a certain relevance score: in Section 2.4 we investigate several different relevance scores with different parameter values in these scores.

Table 2.1: Main Notations

Symbol	Description
$\mathbf{U}$	User set.
u	User.
$\mathbf{C}(u)$	Query composition set of a user u.
c	Query composition.
$K(c)$	Number of keystrokes in a query composition c.
k	Keystroke index: $k \in \{1, 2, \ldots, K(c)\}$.
$\mathbf{Q}$	Query set.
q, q'	Query.
$q^*(c)$	Submitted query in a query composition c.
$r^{(k)}(u, q, c)$	Relevance score for a user u of a query q that matches the prefix at a keystroke k in a query composition c.
$\mathbf{Q}^{(k)}(r, u, c, N)$	Set of top N queries ranked by $r^{(k)}(u, q, c)$.
$\mathbf{x}^{(k)}_{l\times 1}(u, q, c)$	Implicit negative feedback feature vector from a user u to a query q at a keystroke k in a query composition c.
$\mathbf{\Phi}_{l\times m}(\mathbf{U})$	Implicit negative feedback feature weight matrix for a user set $\mathbf{U}$.
$\boldsymbol{\phi}_{l\times 1}(u)$	Implicit negative feedback feature weight vector for a user u.
$p^{(k)}(u, q, c)$	Preference for a query q of a user u at a keystroke k in a query composition c.
λ	Regularizer weight parameter.

2.3.3 Problem Formulation

Consider a user $u \in \mathbf{U}$, where $\mathbf{U}$ is the set of all adaQAC users, at the k-th keystroke in a query composition $c \in \mathbf{C}(u)$, where $\mathbf{C}(u)$ is the query composition set of u. adaQAC suggests a ranked list of queries in $\mathbf{Q}$ according to the ranking scores determined by a probabilistic model. The probabilistic model is based on a combination of the relevance score and the inferred strength of implicit negative feedback. For a query q that matches the prefix at the keystroke k in the query composition c, the relevance score of q for the user u is denoted as $r^{(k)}(u, q, c)$.

Implicit negative feedback from the user u to the query q at the k-th keystroke in the query composition c is represented by a feature vector $\mathbf{x}^{(k)}_{l\times 1}(u, q, c)$, where l is the number of features. The strength of implicit negative feedback is based on $\mathbf{x}^{(k)}_{l\times 1}(u, q, c)$ and its associated implicit negative feedback feature weight vector $\boldsymbol{\phi}_{l\times 1}(u)$ for u. $\boldsymbol{\phi}_{l\times 1}(u)$ is a column vector indexed by u from the implicit negative feedback feature weight matrix $\mathbf{\Phi}_{l\times m}(\mathbf{U})$ for all the users in $\mathbf{U}$. Here m is the number of users in $\mathbf{U}$.

In a query composition c, prefixes with the corresponding suggestion lists are referred to by sequential keystroke indices $k \in \{1, 2, \ldots, K(c)\}$, where $K(c)$ is the number of keystrokes in a query composition c. For instance, for a query composition c starting from an empty string with three keystrokes "fac" ($K(c) = 3$), the prefix "fac" with the suggestion list in the left of Figure 2.1 can be referred to by $k = 3$ in c or simply $K(c)$ in c. Table 3.4 briefly summarizes the main notations.

Table 2.2: Feature descriptions of the adaQAC model. The implicit negative feedback feature vector $\mathbf{x}^{(k)}(u, q, c)$, from a user u to a query q at a keystroke k in a query composition c, contains the following information collected from the beginning of c to the $(k-1)$-th keystroke in c.

Feature	Description
DwellT-M	The maximum dwell time when q is suggested.
DwellT	Total dwell time where q is suggested.
WordBound	No. of the keystrokes at word boundaries when q is suggested.
SpaceChar	No. of the keystrokes at space characters when q is suggested.
OtherChar	No. of the keystrokes at non-alphanum. char. when q is suggested.
IsPrevQuery	1 if q is the immediately previous query; 0 otherwise.
Pos@i	No. of the keystrokes when q is at Position i of a suggestion list ($i = 1, 2, \ldots, 10$).

*Dwell time greater than 3 seconds at one suggestion list is set to 3 seconds.

2.3.4 Personalized Learning

Table 2.2 lists the features used by adaQAC to fit in the "implicit negative feedback" component. Dwell time and positions are studied in Section 2.2.3. Likewise, the other features also indicate how likely users examine query suggestions.

Based on Section 2.2.2, such as the observation that different users may have different typing speed, personalized learning is used: $\boldsymbol{\phi}(u)$ is to be learned separately for each $u \in \mathbf{U}$ to form $\boldsymbol{\Phi}(\mathbf{U})$.

Probabilistic Model

We model preference $p^{(k)}(u, q, c)$ for a query q of a user u at a keystroke k in a query composition c, by a generalized additive model [60]:

$$p^{(k)}(u, q, c) = r^{(k)}(u, q, c) + \boldsymbol{\phi}^{\top}(u)\mathbf{x}^{(k)}(u, q, c). \tag{2.3.1}$$

In (3.3.1), the preference model $p^{(k)}(u, q, c)$ is able to reflect a user u's preference for a query q after the implicit negative feedback $\mathbf{x}^{(k)}(u, q, c)$ is expressed to q before the k-th keystroke in a query composition c. With the associated feature weights $\boldsymbol{\phi}(u)$ personalized for u, $\boldsymbol{\phi}^{\top}(u)\mathbf{x}^{(k)}(u, q, c)$ encodes the strength of implicit negative feedback to q from u with personalization.

When a user u submits a query $q^*(c)$ at the final keystroke $K(c)$ in a query composition c, c ends. The likelihood of the observations on the submitted query in a query composition together with implicit negative feedback in Table 2.2 is to be maximized. Hence, we define a probabilistic model for a submitted query $q^*(c)$ by u at $K(c)$ in c with a softmax function that represents a smoothed version of the "max" function

[13, 179]:

$$\mathbb{P}\Big(Q = q^*(c) \mid u, c, K(c)\Big) = \frac{\exp\Big[p^{\big(K(c)\big)}\big(u, q^*(c), c\big)\Big]}{\sum\limits_{q \in \mathbf{Q}^{(k)}(r,u,c,N) \bigcup \{q^*(c)\}} \exp\Big[p^{\big(K(c)\big)}(u, q, c)\Big]}, \tag{2.3.2}$$

where $\mathbf{Q}^{(k)}(r, u, c, N)$ represents the set of top N queries ranked by $r^{(k)}(u, q, c)$. Its union with $\{q^*(c)\}$ ensures proper normalization. Likewise, adaQAC predicts the likelihood that a query $q' \in \mathbf{Q}^{(k)}(r, u, c, N)$ to be submitted by a user u at any k in c by

$$\mathbb{P}\Big(Q = q' \mid u, c, k\Big) = \frac{\exp\big[p^{(k)}(u, q', c)\big]}{\sum\limits_{q \in \mathbf{Q}^{(k)}(r,u,c,N)} \exp\big[p^{(k)}(u, q, c)\big]} \propto p^{(k)}(u, q', c). \tag{2.3.3}$$

In practice, the simpler form $p^{(k)}(u, q', c)$ in (3.3.3) is used for re-ranking in Stage 2 of adaQAC (Section 2.3.1) after $\boldsymbol{\phi}(u)$ in (3.3.1) is inferred. If a query q never appears in any suggestion list before a keystroke k in a query composition c, $\mathbf{x}^{(k)}(u, q, c)$ is a zero vector and the user u's preference for q is the same as the relevance score $r^{(k)}(u, q, c)$. Here k, c are used to refer to the prefix at k in c and suggested queries must match the prefix. However, if u expresses possibly stronger implicit negative feedback to q before k in c, say q is dwelled longer and at a higher position for several times, then the corresponding weights in $\boldsymbol{\phi}(u)$ updates preference for q of u at k in c with a lower $p^{(k)}(u, q, c)$ value; while possibly weaker implicit negative feedback may correspond to shorter dwell time and a lower position. The strength of the expressed implicit negative feedback determines the level of penalizing u's preference for q in $p^{(k)}(u, q, c)$, which affects how to re-rank in Stage 2 of adaQAC. This agrees with the earlier discussions on using proper implicit negative feedback strength to properly demote an unselected query suggestion (Section 2.2).

We highlight that, the preference model $p^{(k)}(u, q, c)$ in (3.3.1) is designed in the more general framework as discussed in Section 2.3.2. The "(static) relevance" component is $r^{(k)}(u, q, c)$, and $\boldsymbol{\phi}^\top(u)\mathbf{x}^{(k)}(u, q, c)$ acts as "(adaptive) implicit negative feedback".

Batch Inference

In (3.3.1) $\boldsymbol{\phi}(u)$ is inferred with batch inference. The likelihood for all compositions $C(u)$ of a user u should be maximized.

$$\underset{\boldsymbol{\phi}(u)}{\text{maximize}} \quad \prod_{c \in \mathbf{C}(u)} \mathbb{P}\Big(Q = q^*(c) \mid u, c, K(c)\Big). \tag{2.3.4}$$

By (3.3.2) and 2.3.4, a constrained optimization problem out of minimizing negative log-likelihood with $L2$ regularization (to avoid overfitting) is obtained as

$$\begin{aligned} \underset{\boldsymbol{\phi}(u)}{\text{minimize}} \quad & \sum_{c \in \mathbf{C}(u)} \log \sum_{q \in \mathbf{Q}^{(k)}(r,u,c,N) \bigcup \{q^*(c)\}} \exp\left[p^{\left(K(c)\right)}(u,q,c)\right] - p^{\left(K(c)\right)}\left(u, q^*(c), c\right) \\ \text{subject to} \quad & \|\boldsymbol{\phi}(u)\|_2^2 \leq v,\ v \in \mathbb{R}^+. \end{aligned} \tag{2.3.5}$$

There is a one-to-one correspondence between the parameters v in (2.3.5) and $\lambda \in \mathbb{R}^+$, and the corresponding un-constrained optimization problem is:

$$\begin{aligned} \underset{\boldsymbol{\phi}(u)}{\text{minimize}} \quad & \sum_{c \in \mathbf{C}(u)} \log \sum_{q \in \mathbf{Q}^{(k)}(r,u,c,N) \bigcup \{q^*(c)\}} \exp\left[p^{\left(K(c)\right)}(u,q,c)\right] \\ & - p^{\left(K(c)\right)}\left(u, q^*(c), c\right) + \frac{\lambda}{2}\|\boldsymbol{\phi}(u)\|_2^2, \end{aligned} \tag{2.3.6}$$

where λ is the regularizer weight parameter. As there is no closed-form solution for the optimization problem in (2.3.6) due to non-linearity of the softmax function [13], iterative batch inference by gradient descent is used. We refer to an adaQAC model using personalized learning with batch inference as **adaQAC-Batch**. Details for inferring $\boldsymbol{\phi}(u)$ are in Section 2.7.1.

Optimum and Convergence

The objective function of negative log-likelihood for softmax functions with $L2$ regularization in (2.3.6) is strongly convex [113]. Hence, the inference is guaranteed to converge to the global optimum [117]: adaQAC-Batch can be inferred precisely. As we know, for a strongly convex objective function $f(x)$ whose optimal value is achieved with $x = x^*$, the number of iterations to get to accuracy $|f(x^*) - f(x)| \leq \epsilon$ takes a $\mathcal{O}(\ln(\frac{1}{\epsilon}))$ time [18]. Our experiments in Section 2.4.3 reinforce that, adaQAC-Batch converges quickly and reaches the global optimum within a constant number of iterations.

Computational Complexity

Suppose the relevance scores of queries for users, which depend on static QAC, are available. During the training phase for a user u, $\boldsymbol{\phi}(u)$ is inferred with the constructed feature vectors. Assuming the number of queries in a suggestion list and the number of top queries for re-ranking (N in Section 2.3.1) are fixed small constants, the feature construction has a time complexity of $\mathcal{O}(lK(c))$, where l is the feature vector size and $K(c)$ is the number of keystrokes in a query composition c. Since the inference algorithm in Section 2.7.1 converges within a constant number of steps (Section 2.3.4), it takes a $\mathcal{O}(l^2\,|\mathbf{C}(u)|)$ time with a constant

factor corresponding to the number of convergence steps or a predefined value. Here $|\mathbf{C}(u)|$ is the number of query compositions for a user u. Note that the features in Table 2.2 are all distributive functions: the result derived by applying the function to aggregate values is the same as that derived by applying the function on all the data without partitioning. To explain, let $\mathbf{x}_i^{(k)}(u, q, c)$ be $DwellT\text{-}M^{(k)}(u, q, c)$; $DwellT\text{-}M^{(k+1)}(u, q, c)$ can be updated by simply taking the larger value of $DwellT\text{-}M^{(k)}(u, q, c)$ and the dwell time at $k+1$ in c, if q appears in the suggestion list. With a fixed small constant value N (Section 2.3.1), the suggestion at each keystroke takes a $\mathcal{O}(l)$ time.

Scalability on Hadoop MapReduce

A nice property of personalized learning is scalability. As adaQAC-Batch infers $\boldsymbol{\phi}(u)$ for each individual user u, the inference is parallel for different users on big query log data.

In particular, in the Hadoop MapReduce framework, the $\boldsymbol{\Phi}(\mathbf{U})$ inference phase of our experiments is conducted in parallel for different users by different Reducer nodes.

2.3.5 Extensions

For a user u, adaQAC-Batch requires training data related to u to infer the feature weight $\boldsymbol{\phi}(u)$. Now we consider a more challenging cold-start scenario where u is a new user without related data for training. Two ways of extensions can address the challenge.

Un-Personalized Learning

The first way is to infer the feature weights from all the existing users excluding the new user. To maintain scalability on Hadoop MapReduce, a gradient descent variant with averaging is used [179]. This un-personalized approach does not differentiate one user from another, and is referred to as **adaQAC-UnP**.

Because only one feature weight vector is stored and shared by all the users, adaQAC-UnP is cheap in storage.

Online Inference

adaQAC-Batch can be extended to an online inference style. For a new user, first, assign the un-personalized learning output to initialize the feature weights; then, keep update the feature weights with more observations of the user's interactions with QAC.

We call this personalized online learning style extension **adaQAC-Online**. Stochastic gradient descent is used for the online inference. It is similar to batch inference with the constrained optimization problem

out of minimizing negative log-likelihood with $L2$ regularization in (2.3.5) replaced by

$$\begin{aligned} &\underset{\boldsymbol{\phi}(u)}{\text{minimize}} \quad \log \sum_{q \in \mathbf{Q}^{(k)}(r,u,c,N) \bigcup \{q^*(c)\}} \exp\left[p^{\left(K(c)\right)}(u,q,c)\right] - p^{\left(K(c)\right)}\left(u,q^*(c),c\right) \\ &\text{subject to} \quad \|\boldsymbol{\phi}(u)\|_2^2 \leq v,\ v \in \mathbb{R}^+. \end{aligned}$$

Details for inferring $\boldsymbol{\phi}(u)$ are in Section 2.7.2.

Cost Analysis: adaQAC-Online costs more storage than adaQAC-UnP due to maintaining different weights for all the users. As shown in Section 2.4.4, adaQAC-Online trades its storage cost for slightly higher accuracy than adaQAC-UnP. Compared with adaQAC-Batch, the inference of adaQAC-Online takes a $\mathcal{O}(tl^2)$ time, where t is the number of observations and l is the feature vector size. Generally adaQAC-Online takes less time than adaQAC-Batch in inference and has the same storage requirement for maintaining different feature weights for all the users. Comparing with adaQAC-Batch, adaQAC-UnP takes the same order of time with less storage requirement as it maintains only one feature weight vector that is shared by all the users.

2.4 Evaluation

We evaluate the proposed adaQAC-Batch and its two extensions adaQAC-UnP and adaQAC-Online on QAC log data.

2.4.1 Data and Evaluation Measures

Data: We describe important details of our collected QAC log data. Due to the proprietary nature of the data, some details are omitted. The QAC log data are collected from Feb 28 to Jul 28, 2014 and all the queries are submitted via desktops. If a query is submitted by more than two different users, its corresponding query composition is used for evaluation. As adaQAC-Batch requires training data for the feature weight inference, all the users with fewer than 100 query compositions during the given five-month range are filtered out. After the filtering, users are randomly sampled and their 2,932,035 query compositions constitute the evaluation data. There are in total 481,417 unique submitted queries. All the query compositions have their anonymized user IDs and the submitted queries. In one composition, the prefixes, timestamps and suggested queries of every keystroke are collected.

The training, validation and testing data are split with a ratio of 50%/25%/25% in an ascending time order: the first half of a user's query compositions are used for training; the second and third quar-

ters are for validation and testing respectively. The validation data are only used for parameter tuning. As adaQAC infers implicit negative feedback from user-QAC interactions in query compositions, in Section 2.4.2—Section 2.4.5 we experiment on the prefixes at the last keystroke of query compositions to use more interaction information. The average length of query prefixes is 8.53 characters.

The data standardization procedure is transforming data to zero mean and unit variance. All the feature values in Table 2.2 and the relevance scores are standardized.

Measures for Accuracy: Mean reciprocal rank (MRR) is the average reciprocal of the submitted query's ranking in a suggestion list. It is a widely-adopted measure to evaluate the ranking accuracy of QAC [10, 89, 69, 141]. Success Rate@top-k (SR@k) denotes the average percentage of the submitted queries that can be found in the top-k suggested queries on the testing data, and was also used to evaluate the QAC ranking accuracy [69]. In general, a higher MRR or SR@k indicates a higher ranking accuracy of QAC [10, 89, 69, 141, 20]. Paired-t test is used to validate the statistical significance of the accuracy improvement ($p < 0.05$).

2.4.2 Boosting the Accuracy of Static QAC with Implicit Negative Feedback

Following the "(static) relevance–(adaptive) implicit negative feedback" framework (Section 2.3.2), we investigate relevance scores from popular static QAC with different parameter settings to compare the accuracy of adaQAC-Batch, Filtering QAC, and static QAC.

The relevance scores reuse the existing research: MPC [10, 69, 89, 141], Personal(-S) [10, 20, 141], and TimeSense(-S) [20, 142, 168, 112].

• **MPC:** Most Popular Completion (MPC) ranks suggested que-ries for a prefix based on the historical popularity of a query. A more popular query gets a higher rank. Despite its simplicity, it was found competitive by various studies [10, 69, 89, 141].

• **Personal:** Personal QAC for distinguishing different users can achieve better accuracy [10, 20, 141]. Although personal information may take many different forms, the Personal relevance score in this work is an equal-weighted linear combination of the MPC score and the standardized personal historical query frequency counts.

• **Personal-S:** It is the Personal relevance score with an optimal combination with different weights of the MPC score and the standardized personal query frequency counts. The optimal weights achieving the highest accuracy are tuned on validation data. Tuning to the optimal weights makes Personal-S more competitive.

Table 2.3: Accuracy comparison of static QAC, Filtering QAC, and adaQAC-Batch (in percentage). Boldfaced results denote that the accuracy improvement over static QAC is statistically significant ($p < 0.05$) for the same relevance score. adaQAC-Batch significantly and consistently boosts the accuracy of static QAC for each relevance score. For instance, adaQAC-Batch (MPC) significantly boosts static QAC (MPC) by 21.2% in MRR.

Relevance	MRR			SR@1		
	Static	Filter	adaQAC-Batch	Static	Filter	adaQAC-Batch
MPC	50.62	51.83	**61.33 (+21.2%)**	40.74	42.27	**55.86 (+37.1%)**
Personal	61.85	62.68	**70.97 (+14.8%)**	51.31	52.45	**64.27 (+25.3%)**
Personal-S	66.02	66.52	**74.43 (+12.7%)**	55.30	56.24	**67.09 (+21.3%)**
TimeSense	64.32	65.14	**73.70 (+14.6%)**	53.77	54.92	**66.82 (+24.3%)**
TimeSense-S	65.56	66.19	**74.69 (+13.9%)**	55.02	56.11	**67.57 (+22.8%)**
Relevance	SR@2			SR@3		
	Static	Filter	adaQAC-Batch	Static	Filter	adaQAC-Batch
MPC	52.03	53.19	**63.17 (+21.4%)**	58.09	59.21	**66.09 (+13.8%)**
Personal	64.02	64.78	**73.71 (+15.1%)**	70.34	71.09	**76.94 (+9.4%)**
Personal-S	68.51	68.92	**77.73 (+13.5%)**	74.58	74.97	**80.97 (+8.6%)**
TimeSense	66.54	67.45	**76.41 (+14.8%)**	72.39	73.28	**79.81 (+10.3%)**
TimeSense-S	67.83	68.27	**77.76 (+14.6%)**	73.68	74.11	**80.97 (+9.9%)**

*Static: Static QAC; Filter: Filtering QAC

• **TimeSense:** Time is useful in QAC [20, 142, 168]. Hence, TimeSense is the same as Personal except that the personal historical query frequency counts is replaced by the all-user popularity counts of a query in the 28-day time window before a query composition.

• **TimeSense-S:** It is the same as Personal-S except that Personal is replaced by TimeSense.

For brevity, we denote "static QAC employing the MPC relevance score" as "Static (MPC)". Similar notations are used for QAC models employing any relevance score.

Parameters values are tuned to achieve the highest accuracy on validation data. Unless otherwise stated we set the number of iterations to 40 (adaQAC-Batch and adaQAC-UnP) and the regularizer weight to 0.01. Personal-S and TimeSense-S both combine a MPC score with the optimal weight α and the other score with the weight $1 - \alpha$. The optimal weights in Personal-S ($\alpha = 0.34$) and TimeSense-S ($\alpha = 0.42$) achieve the highest MRR for static QAC.

In Section 2.2.4 we set up Filtering QAC with relevance scores, by additionally filtering out all the suggested queries with certain dwell time thresholds (T_{DT}) and position thresholds (T_P) in the subsequent keystrokes in a composition. To ensure higher competitiveness, the model is tuned among the 300 threshold value combinations in Section 2.2.4. We set $T_{DT} = 0.9$ and $T_P = 1$.

Table 2.3 presents the accuracy comparison of static QAC, Filtering QAC, and adaQAC-Batch. The simple Filtering QAC model fails to outperform the corresponding static QAC with the same relevance scores significantly. For each same relevance score, adaQAC-Batch exploiting the added implicit negative feedback information significantly and consistently boosts the accuracy of these static QAC models that only

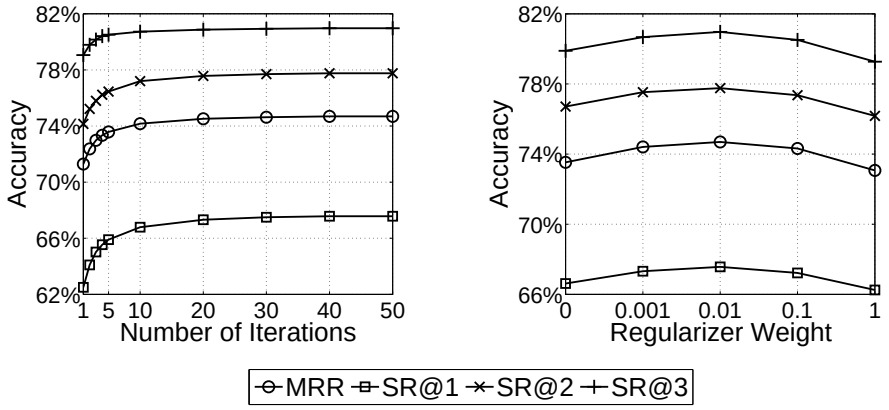


Figure 2.4: Convergence (left) and regularizer weight (right) study for adaQAC-Batch (TimeSense-S). Plots are similar for the other relevance scores. adaQAC-Batch converges quickly and is not sensitive to the chosen regularizer weight near its optimum.

use relevance scores. With more accurate relevance scores such as Personal and TimeSense, adaQAC-Batch is more accurate. Given the relevance scores with different parameter settings (Personal *vs.* Personal-S and TimeSense *vs.* TimeSense-S), the accuracy of adaQAC-Batch slightly varies depending on the accuracy of the relevance scores for the chosen parameter values.

The newly-discovered implicit negative feedback is promising in boosting the accuracy of the existing static QAC models.

2.4.3 Parameter Study

Here we set the number of iterations and regularizer weight to different values for the parameter study on the validation data. adaQAC-Batch (TimeSense-S) is tested. The results for the other relevance scores are similar.

Convergence: Figure 2.4 (left) shows the evaluation measures against the number of iterations. The results reinforce the fact that, adaQAC-Batch converges quickly and the precise global optimum can be reached within a constant number of iterations (Section 2.3.4).

Regularizer Weight: Figure 2.4 (right) plots the evaluation measures of adaQAC-Batch (TimeSense-S) with regularizer weights that are varied around the optimum 0.01. adaQAC-Batch is not sensitive to different regularizer weights near the optimum. This property shows that the accuracy of adaQAC-Batch has less dependence on the chosen regularizer weight value.

Table 2.4: Accuracy of adaQAC-UnP and adaQAC-Online in comparison with static QAC (in percentage). Boldfaced results denote that the accuracy improvement over static QAC is statistically significant ($p < 0.05$) for the same relevance score. Both of the adaQAC extension models significantly and consistently boost the accuracy of static QAC for each relevance score. For instance, adaQAC-Online (MPC) significantly boosts static QAC (MPC) by 20.3% in MRR.

Relevance	MRR		SR@1	
	adaQAC-UnP	adaQAC-Online	adaQAC-UnP	adaQAC-Online
MPC	**60.60 (+19.7%)**	**60.92 (+20.3%)**	**54.54 (+33.9%)**	**55.06 (+35.1%)**
Personal	**69.80 (+12.9%)**	**70.22 (+13.5%)**	**62.27 (+21.4%)**	**62.98 (+22.7%)**
Personal-S	**73.16 (+10.8%)**	**73.59 (+11.5%)**	**64.87 (+17.3%)**	**65.59 (+18.6%)**
TimeSense	**72.69 (+13.0%)**	**73.05 (+13.6%)**	**65.00 (+20.9%)**	**65.61 (+22.0%)**
TimeSense-S	**73.57 (+12.2%)**	**73.96 (+12.8%)**	**65.65 (+19.3%)**	**66.26 (+20.4%)**
Relevance	SR@2		SR@3	
	adaQAC-UnP	adaQAC-Online	adaQAC-UnP	adaQAC-Online
MPC	**62.75 (+20.6%)**	**62.99 (+21.1%)**	**66.01 (+13.6%)**	**66.11 (+13.8%)**
Personal	**72.76 (+13.7%)**	**73.06 (+14.1%)**	**76.64 (+9.0%)**	**76.77(+9.1%)**
Personal-S	**76.83 (+12.1%)**	**77.11 (+12.6%)**	**80.65 (+8.1%)**	**80.81 (+8.4%)**
TimeSense	**75.69 (+13.8%)**	**75.97 (+14.2%)**	**79.64 (+10.0%)**	**79.74 (+10.2%)**
TimeSense-S	**76.78 (+13.2%)**	**77.13 (+13.7%)**	**80.74 (+9.6%)**	**80.91 (+9.8%)**

2.4.4 Un-Personalized Learning and Online Inference

Motivated by the more challenging cold-start scenario where there is a lack of training data for new users, we evaluate the two adaQAC extensions adaQAC-UnP (Section 2.3.5) and adaQAC-Online (Section 2.3.5).

For a user u, the un-personalized learning is performed by learning from training data related to all the users excluding u, and the learned feature weights are fed into adaQAC-Online for u as the initial feature weights. Neither adaQAC-UnP nor adaQAC-Online uses the training data related to u.

Table 2.4 shows that, both adaQAC-UnP and adaQAC-Online significantly and consistently boost the accuracy of static QAC for each relevance score. The mean measure values of adaQAC-UnP and adaQAC-Online are slightly lower than those of adaQAC-Batch for the same relevance score. This slight difference can be justified by the added benefits of the more expensive personalized learning with batch inference of adaQAC-Batch.

It was pointed out that (Section 2.3.5), adaQAC-Online costs more storage than adaQAC-UnP due to maintaining different weights for all the users. The slight difference between the mean of the measure values of adaQAC-Online and adaQAC-UnP in Table 2.4 shows that, adaQAC-Online trades its storage cost for slightly higher accuracy than adaQAC-UnP. In addition to the benefits for addressing the cold-start challenge, according to the cost analysis in Section 2.3.5, an important practical implication from the results of Table 2.4 is, adaQAC-UnP and adaQAC-Online can be good substitutes for the more expensive adaQAC-Batch if time and storage budgets are limited in the real-world productions.

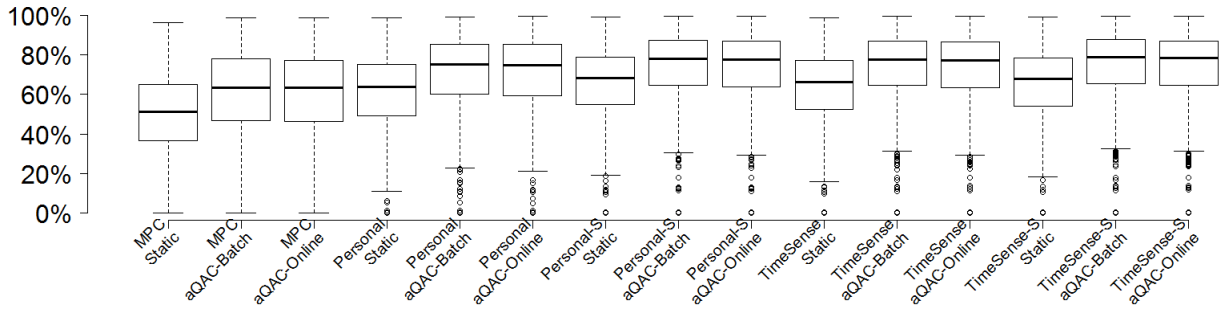


Figure 2.5: Box-and-Whisker plots of individual users' MRR for static QAC, adaQAC-Batch, and adaQAC-Online with five relevance scores. Each data instance is the corresponding MRR on one user. The minimum (bottom bar), quartiles (box edges), median (middle of the box), and maximum (top bar) after removal of the detected outliers (empty circles) are depicted. adaQAC with more accurate relevance scores are able to detect more outliers with the raised minimum bars.

2.4.5 Model Accuracy on Different Users

We study the model accuracy on different users using the Box-and-Whisker plots. With each data instance being the MRR on one user, Figure 2.5 shows the minimum (bottom bar), quartiles (box edges), median (middle of the box), maximum (top bar) after removal of the detected outlier users (empty circles).

In general, model comparison using medians and quartiles of MRR agrees with the results in Table 2.3—2.4 and reaffirms the boosted accuracy by the added implicit negative feedback.

Note that, all the models perform poorly on a few users. Models with the MPC relevance score fail to detect any outlier, and have minimum bars close to 0. The other models still perform poorly on certain users with MRR close to 0. These users are detected as the outliers. The outlier users may behave inconsistently, submit rare queries, or the collected data related to them are noisy or incomplete due to unknown reasons.

To explain, models with the MPC relevance have a larger MRR variance (implied by a longer box in Figure 2.5) so outlier users cannot be easily detected. It is easier to see when comparing adaQAC-Online (MPC) with Static (Personal): they have close medians but the lower-variance Static (Personal) is able to detect a few outliers and raise its minimum bar after their removal. When the relevance score is more accurate with a lower variance, adaQAC is able to detect more outliers thus raises the minimum bar by further improving the MRR on the majority of the users.

Hence, even though the implicit negative feedback research is promising, further research on more accurate relevance scores is still required.

Table 2.5: MRR of static QAC, adaQAC-Batch, and adaQAC-Online under prefixes with varying lengths at every keystroke in query compositions (in percentage). Boldfaced results denote that the accuracy improvement over static QAC is statistically significant ($p < 0.05$) for the same relevance score and prefix length range. Both adaQAC-Batch and adaQAC-Online significantly and consistently boost the accuracy of static QAC under all prefix lengths for each relevance score. For instance, adaQAC-Batch (MPC) significantly boosts static QAC (MPC) by 17.1% in MRR under all prefix lengths.

Relevance	Static	adaQAC-Batch	adaQAC-Online	Static	adaQAC-Batch	adaQAC-Online
	$1 \leq$ Prefix Length ≤ 3			$4 \leq$ Prefix Length ≤ 6		
MPC	21.76	22.66	22.52	33.64	**38.67 (+15.0%)**	**38.40 (+14.1%)**
Personal	29.34	30.31	30.10	45.41	**49.52 (+9.1%)**	**49.38 (+8.7%)**
Personal-S	31.60	32.59	32.36	50.14	**53.38 (+6.5%)**	**53.30 (+6.3%)**
TimeSense	29.98	**31.94 (+6.5%)**	**31.91 (+6.4%)**	47.75	**52.61 (+10.2%)**	**52.65 (+10.3%)**
TimeSense-S	30.93	32.69	32.59	49.27	**53.69 (+9.0%)**	**53.67 (+8.9%)**
Relevance	Static	adaQAC-Batch	adaQAC-Online	Static	adaQAC-Batch	adaQAC-Online
	$7 \leq$ Prefix Length ≤ 9			$10 \leq$ Prefix Length ≤ 12		
MPC	41.60	**52.21 (+25.5%)**	**52.13 (+25.3%)**	47.28	**55.13 (+16.6%)**	**54.82 (+15.9%)**
Personal	49.81	**56.34 (+13.1%)**	**56.19 (+12.8%)**	52.16	**57.79 (+10.8%)**	**57.33 (+9.9%)**
Personal-S	53.94	**58.69 (+8.8%)**	**58.57 (+8.6%)**	55.21	**59.33 (+7.5%)**	**58.77 (+6.4%)**
TimeSense	52.48	**58.63 (+11.7%)**	**58.46 (+11.4%)**	54.91	**59.43 (+8.2%)**	**59.06 (+7.6%)**
TimeSense-S	53.65	**59.19 (+10.3%)**	**59.14 (+10.2%)**	55.73	**59.83 (+7.4%)**	**59.37 (+6.5%)**
	Prefix Length ≥ 13			All Prefix Lengths		
MPC	55.12	**59.28 (+7.5%)**	**58.94 (+6.9%)**	38.19	**44.72 (+17.1%)**	**44.43 (+16.3%)**
Personal	56.59	**59.93 (+5.9%)**	59.32	46.67	**51.75 (+10.9%)**	**51.20 (+9.7%)**
Personal-S	58.40	60.85	60.16	49.83	**54.30 (+9.0%)**	**53.66 (+7.7%)**
TimeSense	58.49	61.08	60.54	48.59	**53.77 (+10.7%)**	**54.01 (+11.2%)**
TimeSense-S	58.97	61.29	60.76	49.48	**54.47 (+10.1%)**	**53.95 (+9.0%)**

*Static: Static QAC

2.4.6 Varying-Length Prefix Study

Now we consider another challenging scenario where testing is based on all possible prefixes in query compositions. Table 2.5 reports MRR of static QAC, adaQAC-Static and adaQAC-Online for prefixes with varying lengths at every keystroke in query compositions. Both adaQAC-Batch and adaQAC-Online still significantly and consistently boost the accuracy of static QAC under all prefix lengths for each relevance score.

The MRR gap between adaQAC-Batch and adaQAC-Online is subtle and both are more accurate when prefixes are of "middle" lengths. That is, when the prefixes are short, the collected implicit negative feedback features probably contain little useful information to improve the re-ranking in Stage 2 of adaQAC (Section 2.3.1). When prefixes get longer, more user-QAC interaction information is obtained to make adaQAC more accurate in the adaptive re-ranking stage. However, when prefixes are longer, the QAC problem becomes less challenging due to a reduction of the matched queries: static QAC employing relevance scores are more accurate and it is harder to further improve the accuracy, even though the implicit negative feedback information may be richer.

2.4.7 Case Study

adaQAC has advantages over static QAC. We describe the following cases of Yahoo! Search, and hope that this work can inspire ongoing studies in a broader research community.

Disambiguation: When users have clear query intent and prefer disambiguated queries, adaQAC generally outperforms static QAC. Typically, users may prefer queries of the form "entity name + attribute" to "entity name only". Suppose a user wants to know the showtime of lefont sandy springs. When the user composes the query during the keystrokes "lefon", the entity name "lefont sandy springs" is the top suggestion. The user does not select it because an entity name query may result in diverse search results. So, the query "lefont sandy springs" receives implicit negative feedback. When the prefix becomes "lefont", "lefont sandy springs" is demoted by adaQAC and "lefont sandy springs showtime" gets promoted.

Query Reformulation: When users prefer new queries when reformulating older queries, adaQAC generally outperforms static QAC. Suppose a user wants to query "detroit lions" after querying "detroit red wings". When the user reformulates the query from "detroit red wings" to "detroit r" by consecutively hitting Backspace, "detroit red wings" is ranked highest but the user does not select it. So, the query "detroit red wings" receives implicit negative feedback. Hence, when the prefix becomes "detroit" after the user hits two more Backspace, "detroit red wings" is demoted by adaQAC; some other queries, such as "detroit lions", are promoted accordingly.

Smoothing "Over-Sense": Certain relevance scores may be sensitive to specific signals: TimeSense is sensitive to time. Studies showed users may have query intent for new or ongoing events [4, 71, 83]. In Yahoo! Search, we investigate the QAC results responded by the time-sensitive component. When a user wants to query an earlier event "russia attack georgia", the time-sensitive QAC keeps ranking a more recent event "russia attack ukraine" highest during keystrokes "russia att". Instead, adaQAC receives users' implicit negative feedback to 'russia attack ukraine" hence demotes it, and raises "russia attack georgia" up to the top.

2.5 Related Work

Query Auto-Completion (QAC): Numerous QAC models have been developed in recent years, such as popularity-based QAC using historical frequency counts [10], time-based QAC using time information [142, 168], context-based QAC using previous query information of users [10], personalized QAC learning from user profile information [141]. The relevance scores investigated in our work make use of the existing research, such as MPC [10, 69, 89, 141], Personal(-S) [10, 20, 141], and TimeSense(-S) [20, 142, 168, 112].

More recent QAC methods also predicted the probability that a suggested query would be clicked by users based on user models [74, 89], determined suggestion rankings based on query reformulation patterns [69], or combined information such as time and previous queries from users [20]. Furthermore, user interactions with QAC just began to be explored. Mitra *et al.* discussed user-QAC interactions from perspectives such as word boundaries, fraction of query typed, and keyboard distance [111]. Hofmann *et al.* identified common behavior patterns of user-QAC interactions [61].

Other aspects of QAC have also been studied, such as space efficient indexing [66] and spelling error toleration [26, 68, 41, 171].

However, none of the aforementioned work aimed at inferring implicit negative feedback from user-QAC interactions, or adapting QAC to such feedback. We take these initiatives and show that QAC can adapt to implicit negative feedback and be more accurate.

Negative Feedback: Relevance feedback is useful for improving information retrieval models, but further improving it using negative feedback was considered challenging [133, 107]. Recently, more efforts on negative feedback research was made in document retrieval tasks. Wang *et al.* found negative relevance feedback useful to improve vector-space models and language models [165]. Hong *et al.* proposed a hierarchical distance-based measure to differentiate the opposite intent from the true query intent [63]. Zhang and Wang studied language models with negative feedback through positive and negative document proportion on query classification [189]. New models using negative relevance feedback were also developed in TREC [91]. In particular, negative feedback was also found useful to retrieve documents for difficult queries [164, 73, 104].

However, these negative feedback studies focus only on document retrieval tasks. The richer interaction information, presented in the QAC settings, such as dwell time and positions, is not available in general document retrieval settings.

Dynamic IR: Recent work have gone beyond existing IR techniques to incorporate dynamics in session search [55, 103]. In this task, added or removed terms compared with the other queries of the same search session will update term weights to retrieve documents for the completed query [55, 103]. There are important differences between such research and ours. First, search and QAC are different problems. Second, adaQAC emphasizes adapting dynamics over a single query composition rather than multiple queries over a search session. Third, adaQAC does not assign weights to characters, prefixes or terms of a query. Other dynamic IR work was surveyed in a tutorial by Yang *et al.* [176].

2.6 Conclusion

We studied interactions between users and QAC where users implicitly express negative feedback to suggested queries. Under the more general "(static) relevance–(adaptive) implicit negative feedback" framework, our proposed adaQAC model can reuse the existing static QAC research and adapt QAC to implicit negative feedback using personalized learning with batch inference. Extensions with un-personalized learning and online inference were also presented. We collected user-QAC interaction data from a commercial search engine. Large-scale empirical results showed that implicit negative feedback significantly and consistently boosts the accuracy of the investigated static QAC models.

2.7 Details of the Inference

We present details of the inference for both adaQAC-Batch and adaQAC-Online.

2.7.1 Inference for adaQAC-Batch

Let $f\left[\boldsymbol{\phi}^{(t)}(u)\right]$ be the objective function in (2.3.6), where $\boldsymbol{\phi}^{(t)}(u)$ is the value of $\boldsymbol{\phi}(u)$ at the t-th iteration,

$$\boldsymbol{\phi}^{(t+1)}(u) = \boldsymbol{\phi}^{(t)}(u) - \eta \nabla f\left[\boldsymbol{\phi}^{(t)}(u)\right], \tag{2.7.1}$$

where

$$\nabla f\left[\boldsymbol{\phi}(u)\right] = \left[\frac{\partial f\left[\boldsymbol{\phi}(u)\right]}{\partial \boldsymbol{\phi}_1(u)}, \frac{\partial f\left[\boldsymbol{\phi}(u)\right]}{\partial \boldsymbol{\phi}_2(u)}, \ldots, \frac{\partial f\left[\boldsymbol{\phi}(u)\right]}{\partial \boldsymbol{\phi}_l(u)}\right]^\top, \tag{2.7.2}$$

and $\forall i = 1, 2, \ldots, l$,

$$\frac{\partial f\left[\boldsymbol{\phi}(u)\right]}{\partial \boldsymbol{\phi}_i(u)} = \sum_{c \in \mathbf{C}(u)} \frac{S_1}{S_2} - \mathbf{x}_i^{\left(K(c)\right)}\left(u, q^*(c), c\right) + \lambda \boldsymbol{\phi}_i(u), \tag{2.7.3}$$

where by denoting $\exp\left[r^{(k)}(u,q,c) + \boldsymbol{\phi}^\top(u)\mathbf{x}^{\left(K(c)\right)}(u,q,c)\right]$ as $E(q)$,

$$\begin{aligned} S_1 &= \sum_{q \in \mathbf{Q}^{(k)}(r,u,c,N) \bigcup \{q^*(c)\}} E(q)\mathbf{x}_i^{\left(K(c)\right)}(u,q,c), \\ S_2 &= \sum_{q \in \mathbf{Q}^{(k)}(r,u,c,N) \bigcup \{q^*(c)\}} E(q). \end{aligned} \tag{2.7.4}$$

In the experiments, $\boldsymbol{\phi}^{(0)}(u)$ in (2.7.1) is randomly sampled from:

$$\boldsymbol{\phi}^{(0)}(u) \sim \text{Uniform}(0, 0.01).$$

2.7.2 Inference for adaQAC-Online

The feature weight $\boldsymbol{\phi}^{(0)}(u)$ is initialized as the un-personalized learning weight (Section 2.3.5). After each query composition c, the feature weight is updated as in (2.7.1)—(2.7.4) with (2.7.3) replaced by

$$\frac{\partial f\left[\boldsymbol{\phi}(u)\right]}{\partial \boldsymbol{\phi}_i(u)} = \frac{S_1}{S_2} - \mathbf{x}_i^{\left(K(c)\right)}\left(u, q^*(c), c\right) + \lambda \boldsymbol{\phi}_i(u),$$

and η is discounted by a factor of 0.9 after each update as an annealing procedure [82].

Chapter 3

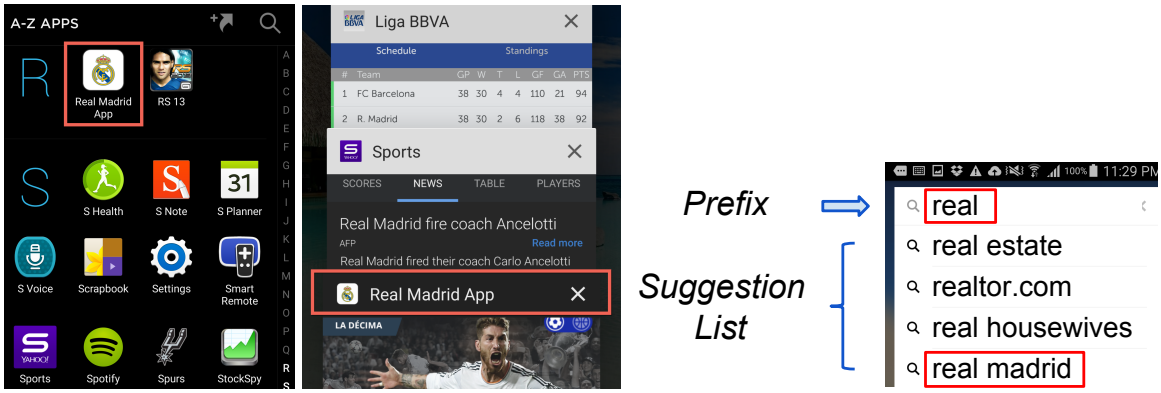
Intention Analysis from Human Activities on Mobile Application Usage

We continue to study QAC in this chapter. Specifically, we study the new mobile QAC problem to exploit mobile devices' exclusive signals, such as those related to mobile apps. We propose AppAware, a novel QAC model using installed app and recently opened app signals to suggest queries for matching input prefixes on mobile devices. To overcome the challenge of noisy and voluminous signals, AppAware optimizes composite objectives with a lighter processing cost at a linear rate of convergence. We conduct experiments on a large commercial data set of mobile queries and apps. Installed app and recently opened app signals consistently and significantly boost the accuracy of various baseline QAC models on mobile devices.

3.1 Introduction

Query auto-completion (QAC) facilitates user query compositions by suggesting queries given prefixes. Figure 3.1(c) depicts an example of QAC on mobile devices. Upon a user's keystroke, QAC displays a *suggestion list* (or *list*) below the current *prefix*. Queries in a suggestion list are called *suggested queries* or *query suggestions*. A user can select to submit a suggested query or type to submit a query without selecting any suggestion.

Baeza-Yates *et al.* found that Japan Yahoo Search users generally typed longer queries on mobile devices than desktops to avoid having to query again as mobile internet was slower in 2007 [8]. A report from Microsoft Bing also observed that English queries are generally longer from mobile users than desktop users, and believed that "query auto-suggestion plays an important role" [145]. We further discover that in 2014, global users of Yahoo Search on mobile devices saved more than 60% of the keystrokes when submitting English queries by selecting QAC suggestions. In comparison with such keystroke saving on desktops (around 50%) [184], users tend to rely on mobile QAC more heavily. It is probably due to the inconvenience of typing on mobile devices as revealed by Google Search [72]. In fact, users can type 21 words per minute on mobile devices but more than 60 words per minute on desktops [49]. Thus, QAC is even more important to mobile users than desktop users.



(a) Installed apps. (b) Recently opened apps. (c) Mobile query auto-completion.

Figure 3.1: A commercial mobile QAC. The *Real Madrid* app is installed and recently opened. Given prefix "real", popular queries on real estate ("real estate" and "realtor.com") are suggested at higher positions than query "real madrid".

Typically, a user favors and submits a query if it reflects the user's query intent in a query composition. Predicting query intents is nontrivial. Most of the recently proposed QAC models rank a list of suggested queries for each prefix according to relevance scores based on various signals, such as popularity-based QAC (historical query frequency count signals) [10], time-based QAC (time signals) [142, 168], context-based QAC (user previous query signals) [10], personalized QAC (user profile signals) [141], or time-and-context-based QAC (both time and user previous query signals) [20]. Note that the aforementioned signals are available on both desktops and mobile devices. Are there any useful signals exclusively exploitable on mobile devices for mobile QAC? Let us look at a few examples.

A mobile application is hereinafter referred to as a *mobile app* or simply as an *app*. Consider a fan of the Real Madrid Football Club who installs the *Real Madrid* app on the smart phone. The user opens this app and after a while wants to query "real madrid" to learn more of this club on the web with a popularity-based QAC [10]. When the user types "real", real estate-related queries, such as "real estate" and "realtor.com", are ranked at the top in the suggestion list because they are most popular in historical query logs. Figure 3.1 displays the user's installed apps, recently opened apps, and a commercial search engine QAC on the same mobile device. Here the user may have implicitly provided the query preference via the installed football club's app. Besides, the user's query intent may also be implied by the recently opened app if the subsequent query interest arises from the app opening. In a large commercial data set, we observe that on mobile devices and matching certain prefixes, users that install the *NBA* app may submit more queries related to basketball teams, and users may query lyrics more often after opening a music app (Section 3.2). Being aware of app

installation and opening on mobile devices, can QAC be more accurate on mobile devices? Our work answers this question affirmatively.

New Problem, New Challenge. To the best of our knowledge, no existing QAC employs mobile devices' exclusive signals. Hence, our goal is to study the new *mobile QAC* problem: QAC using mobile devices' exclusive signals. We refer to QAC that does not employ any signal exclusive to mobile devices as *Standard QAC*, such as QAC based on popularity and time. Mobile app-related signals are exclusive to mobile devices [9]. The sets of all available applications on desktops and mobile devices are different; even for desktop and mobile versions of the related applications, their contents or interfaces generally differ [67]. Although whether desktop applications can improve QAC is also an open question, we study mobile QAC by exploiting mobile devices' exclusive signals from installed mobile apps and recently opened mobile apps. This is motivated by the importance of mobile QAC.

We model the query–app relationships and the order of recently opened apps before query submissions. It is challenging because such signals are noisy and voluminous. In many cases, a certain installed app may not indicate a higher likelihood of a certain query submission. Besides, even though a certain app opening (*Real Madrid* app) may suggest a higher chance of a certain query ("real madrid"), when another app such as *Realtor.com* is opened more recently before a query, the less recently opened app (*Real Madrid* app) may be less relevant to the query intent. Moreover, even for 1,000 queries and 100 apps, potentially there can be voluminously 100,000 query–app relationship pairs to process.

Our Approach. We go beyond Standard QAC by exploiting signals exclusive to mobile devices. To solve the mobile QAC problem, we propose AppAware, a novel model to employ installed app and recently opened app signals. AppAware reuses the relevance scores of queries from Standard QAC to pre-index top queries. In a single query composition, AppAware re-ranks these top queries based on installed app and recently opened app signals. For these signals, AppAware captures relationships between different mobile queries and apps, and the order of recency for opened apps before query submissions.

To overcome the challenge of noisy and voluminous signals, AppAware optimizes a convex composite objective function by single-stage random coordinate descent with mini-batches. The composite objectives include filtering out noisy signals. When processing voluminous signals, the algorithm has a lighter processing cost at each iteration than either full proximal gradient descent or the gradient update with respect to all coordinates. Importantly, while enjoying a lighter processing cost for voluminous signals and capable of noisy signal filtering, our algorithm converges to the global optimum at a linear rate with a theoretical guarantee.

We make the following contributions:

- We jointly study mobile queries and apps from commercial products (Section 3.2). Specifically, we find that going beyond Standard QAC by exploiting installed app and recently opened app signals for mobile QAC is useful. For example, recently opened app signals abound on mobile devices before query submissions.
- We propose a novel AppAware model that exploits installed app and recently opened app signals to solve the mobile QAC problem (Section 3.3). To overcome the challenge of noisy and voluminous signals, AppAware optimizes composite objectives by an algorithm using single-stage random coordinate descent with mini-batches. We prove that our algorithm converges to the global optimum at a linear rate with a theoretical guarantee.
- We conduct comprehensive experiments (Section 3.4). Among many findings, we show that installed app and recently opened app signals consistently and significantly boost the accuracy of various investigated Standard QAC models on mobile devices.

3.2 Mobile Query and Application

We jointly study mobile query logs and mobile app logs from commercial products at a large scale and discuss our observations.

Terminology. In general, *mobile devices* (*devices*) are handheld computing devices with an operating system where various types of mobile apps can run. Below are other used terms.

Query composition (*Composition*): The duration of composing and submitting a single query. It starts from the keystroke of a new query's first character, or from the keystroke starting to edit a previous query. It ends when a query is submitted. A composition contains information on all keystrokes (with the timestamp of the first keystroke), submitted query, installed apps at the first keystroke time, and recently opened apps with timestamps.

Before query: Before the first keystroke of a query composition.

Mobile log data set: Our jointly collected data set of mobile query logs and mobile app logs from Yahoo. It contains 823,421 compositions sampled from 5 months in 2015. In one composition, all keystrokes (with the timestamp of the first keystroke), the submitted query, installed apps at the first keystroke time, and recently opened apps with timestamps are collected.

Example 1 (Mobile Query and Installed App). Users install apps on mobile devices. Some apps may reflect users' interests or preferences in sports, business, and other fields. Users' interests or preferences exhibited from their installed apps may be relevant to their query intents. Table 3.1 compares top queries (with percentage) prefixed by "chicago" from all users' mobile devices in the mobile log data set where the

Table 3.1: Top queries (with percentage) prefixed by "chicago" from all users' mobile devices where the *NBA* app is installed (left) or not (right).

chicago bulls	24%	chicago tribune	11%
chicago bears	12%	chicago weather	10%
chicago cubs	10%	chicago bears	9%
chicago blackhawks	9%	chicago craiglist	9%
chicago tribune	7%	chicago cubs	8%

Table 3.2: Top queries (with percentage) prefixed by "sugar" from all users' mobile devices where the *Spotify Music* app is opened within 30 minutes before queries (left) or not (right).

sugar maroon 5 lyrics	22%	sugar cookie recipe	13%
sugar lyrics maroon 5	18%	sugar glider	11%
sugar lyrics	14%	sugar bowl	10%
sugar maroon 5	13%	sugar maroon 5 lyrics	10%
sugar daddy	9%	sugar sugar	9%

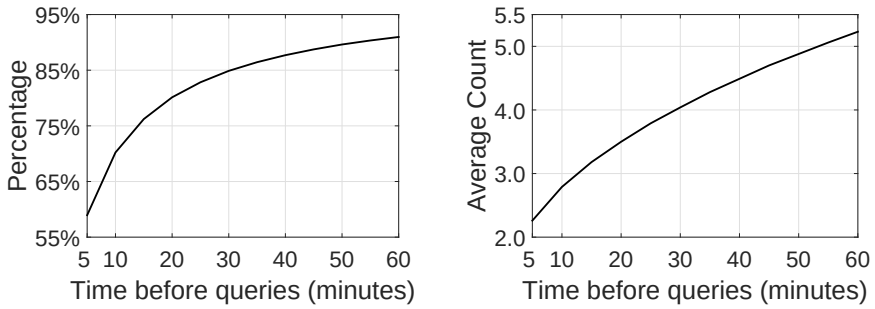
Table 3.3: Mobile app installation and opening statistics according to the Yahoo Aviate team.

Description	Average count
Installed apps per mobile device	95
App opening per day	100
Unique apps that are opened per day	35

NBA app is installed (left) or not (right). Among all the mobile queries prefixed by "chicago" submitted from devices installing the *NBA* app, 24% are "chicago bulls" followed by "chicago bears" with a sharp fall in its percentage. However, "chicago bulls" is not among the top 5 mobile queries prefixed by "chicago" on devices without installing the *NBA* app. So, installing the *NBA* app may exhibit users' interests in NBA basketball teams, such as Chicago Bulls (not Chicago Bears). Since the top 4 queries on the left column of Table 3.1 are sport teams, an NBA fan may generally submit more sports-related queries.

Example 2 (Mobile Query and Recently Opened App). Users open apps to perform activities, such as listening to music. After users open apps, the subsequent query intents may arise from the performed activities through those apps. Table 3.2 compares top queries (with percentage) prefixed by "sugar" from all users' mobile devices in the mobile log data set where the *Spotify Music* app is opened within 30 minutes before queries (left) or not (right). Four of five top queries on the left column of Table 3.2 are related to the song Sugar by the pop rock band Maroon 5. So, users may tend to search for music-related items, such as lyrics, after opening music apps on mobile devices.

Abundance of Signals. From the two examples above, signals of installed apps and recently opened apps may be useful for boosting the accuracy of mobile QAC. We proceed to study the existence of such app-related signals. The Yahoo Aviate team reported mobile app installation and opening statistics in Table 3.3. On average, there are 95 installed apps on each mobile device and they are opened 100 times every day. Some apps are opened more than once in a day and on average 35 unique apps are opened per day.



(a) Mobile queries (b) Recently opened apps

Figure 3.2: Recently opened app signals abound on mobile devices before queries. The left figure shows the percentage of mobile queries that have non-zero recently opened apps (at least one app is opened within a given time before queries). The right figure shows the average count of unique recently opened apps within a given time before queries (compositions that have no recently opened apps within the given time are not counted).

To further investigate opened app signals, there are two interesting open questions: do users open apps before query submissions within a short time? If so, how many unique apps do they open? To answer these questions, we jointly study mobile queries and apps. Figure 3.2(a) shows the percentage of mobile queries that have non-zero recently opened apps (at least one app is opened within a given time before queries). Specifically, 84.9% of mobile queries belong to the cases where at least one app is opened within 30 minutes before queries. Figure 3.2(b) shows the average count of unique recently opened apps within a given time before queries (compositions that have no recently opened apps within the time are excluded). Among those 84.9% queries, on average 4.0 unique apps are opened within 30 minutes before queries. Recently opened app signals abound on mobile devices before query submissions.

Recall Section 3.1 that mobile QAC is important. Given the observations that app-related signals may imply users' query intents and the abundance of such signals, it is appealing to exploit them for mobile QAC. We propose and discuss an app-aware approach to exploit such signals for mobile QAC in Section 3.3.

3.3 Application-Aware Approach

For mobile QAC, we propose the AppAware model to exploit installed app and recently opened app signals on mobile devices.

3.3.1 Design Overview

Before detailing the problem and method, we describe the high-level design of AppAware to rank suggested queries for a given prefix on mobile devices. AppAware has two stages: pre-indexing and re-ranking. A toy

example of two suggestions "real estate" and "real madrid" matching prefix "real" is used to describe the idea.

In the pre-indexing stage, given an input prefix, top N query suggestions with the highest relevance scores of Standard QAC are pre-indexed: a higher score gives a higher position. For prefix "real", the top 2 queries "real estate" and "real madrid" are pre-indexed by Standard QAC based on the historical query frequency counts. In the re-ranking stage, AppAware re-ranks these top N queries based on installed app and recently opened app signals in the same query composition. To illustrate, given prefix "real", the pre-indexed queries "real estate" and "real madrid" are instantly fetched. If a user's preference for "real madrid" to "real estate" is inferred from signals of the installed and recently opened *Real Madrid* app, AppAware updates the ranking scores of the two queries. The top 2 queries "real estate" and "real madrid" are re-ranked. With re-ranking, "real madrid" is now at Position 1, higher than the more popular query "real estate".

The number of the pre-indexed top queries N can be set to a small positive integer in a production. Given various display sizes of mobile devices, a smaller number of top queries may be suggested. For a small constant value N, sorting N queries based on the updated ranking scores can be achieved in a constant time [30].

AppAware is designed to reuse existing Standard QAC research in computing the relevance score of a query. It can be available via an existing Standard QAC model, such as a popularity-based QAC. However, AppAware is not constrained to use any certain relevance score: in Section 3.4 we evaluate several different relevance scores with different parameter settings in these scores.

3.3.2 Problem Formulation

Recall Section 3.2 that a query composition contains information on all keystrokes (with the timestamp of the first keystroke), the submitted query, installed apps at the first keystroke time, and recently opened apps with timestamps. We assume that signals are the same at all the keystrokes of the same composition. To keep notations unclogged, an AppAware output depends on signals of a certain composition rather than an explicit keystroke of this composition. During composition c, AppAware suggests a ranked list of queries matching a given prefix in query set $\mathcal{Q}$ according to ranking scores determined by a probabilistic model. The probabilistic model is based on a combination of the relevance score and app-related signal score on mobile devices. For query q that matches a prefix in composition c, the relevance score of q is denoted as $s(q, c)$. In a composition, installed app and recently opened app signals are represented by x and y. The app-related signal score is based on x and y, and their associated signal parameters β. A collection of β form the signal parameter vector $\mathbf{w}$. This is for indexing convenience in our technical discussions: subscripts

Table 3.4: Main notations

Symbol	Description
$a \in \mathcal{A}$	App and app set.
$q \in \mathcal{Q}$	Query and query set
$c \in \mathcal{C}$	Composition and composition set
$q^{(c)}$	Submitted query in composition c
$\mathcal{A}^{(c)}$	Set of installed apps on the device of composition c
$\widetilde{\mathcal{A}}^{(c)}$	Set of recently opened apps in composition c
$\widetilde{a}_k^{(c)}$	k^{th} most recently opened app in composition c
$s(q, c)$	Relevance score of query q that matches a given prefix in composition c
$p(q, c)$	Preference for query q in composition c
$\mathcal{Q}^{(c)}$	Set of top N queries ranked by $s(q, c)$
$\mathbf{w}$	Signal parameter vector
x, y	Signals of installed apps and recently opened apps

of β correspond to queries, apps, and recency orders (Section 3.3.3), while subscripts of w locate elements in vector $\mathbf{w}$ (Section 3.3.4 and Section 3.3.5). The goal is to compute $\mathbf{w}$ by an optimization algorithm. Table 3.4 briefly summarizes the main notations. Some of them are described in Section 3.3.3.

3.3.3 Likelihood Function

To compute the signal parameter vector $\mathbf{w}$, we need a likelihood function integrating signals and $\mathbf{w}$.

As discussed in Section 3.2, installed apps may reflect users' interests or preferences. However, even if two different users both install the same app, their interests or preferences related to that app may still be at different levels. For example, one may like the app, while the other may dislike it but forget to remove it. We cannot directly observe these and we resort to the opening frequency of apps. Intuitively, more frequently opened apps may be more likely related to users' interests or preferences. For example, consider one user who opens the *Real Madrid* app every day and the other who almost never opens it after installation. The former user is more likely interested in the Real Madrid football club than the latter. Besides, suppose that different users install the same app of the same level of interests at different time. A user more likely has a higher app opening frequency aggregated from a longer app installation history. In light of this, daily opening frequency can be used for comparison. An installed app signal $x(a, c)$ with respect to app a in composition c is the average daily opening frequency of app a on the mobile device of composition c.

Note that recently opened apps in a composition are already opened by users. Recall the assumption that app openings may reflect users' interests or preferences related to the apps, signals of recently opened apps are directly built in relation to submitted queries in the same composition. So, a recently opened app signal $y(q, a)$ with respect to query q and app a is computed based on the training data set. It is the proportion of the count of q to the count of all queries for all compositions where a is a recently opened app.

Let $\mathcal{A}^{(c)}$ be the set of installed apps on the device of composition c, and $\widetilde{\mathcal{A}}^{(c)} = \{\widetilde{a}_1^{(c)}, \widetilde{a}_2^{(c)}, \ldots\}$ of size $|\widetilde{\mathcal{A}}^{(c)}|$ be the set of unique recently opened apps in composition c, where $\widetilde{a}_k^{(c)}$ is the k^{th} most recently opened app in c. If an app is opened more than once in the same composition, only the most recent one is included in $\widetilde{\mathcal{A}}^{(c)}$. We model preference $p(q, c)$ for query q in composition c by a generalized additive model [60]:

$$p(q, c) = s(q, c) + \sum_{a \in \mathcal{A}^{(c)}} \beta_{q,a} \log \left[1 + x(a, c)\right] + \sum_{k=1}^{|\widetilde{\mathcal{A}}^{(c)}|} \beta_k y(q, \widetilde{a}_k^{(c)}), \tag{3.3.1}$$

where $\beta_{q,a}$ and β_k are signal parameters. Note that every $\beta_{q,a}$ corresponds to a query–app pair for all $q \in \mathcal{Q}$ and $a \in \mathcal{A}$, where $\mathcal{Q}$ and $\mathcal{A}$ are the sets of queries and apps in the training data set. Signal parameter β_k is only related to recency order k for app opening in any composition. Values of signals x and y are pre-computed in parallel and stored distributively in a Hadoop MapReduce framework. Such values are directly fetched in training and testing without re-computing. The logarithm transformation of daily opening frequency in (3.3.1) is to dampen the effect of a higher frequency.

In general, the preference model $p(q, c)$ in (3.3.1) reflects a user's preference for query q in composition c in conjunction with installed app signals and recently opened app signals. The signal parameters $\beta_{q,a}$ and β_k are to be inferred based on maximizing the likelihood of submitted queries, together with those integrated app-related signals observed from the training data set. In order to infer such parameters, we define a likelihood function for a submitted query $q^{(c)}$ in c with a softmax function that represents a smoothed version of the "max" function [13, 179]:

$$\mathbb{P}(q^{(c)} \mid c) = \frac{\exp\left[p(q^{(c)}, c)\right]}{\sum_{q \in \mathcal{Q}^{(c)} \bigcup \{q^{(c)}\}} \exp\left[p(q, c)\right]}, \tag{3.3.2}$$

where $\mathcal{Q}^{(c)}$ represents the set of top N queries ranked by relevance score $s(q, c)$. Its union with $q^{(c)}$ ensures proper normalization. Likewise, AppAware predicts the likelihood that any query $q' \in \mathcal{Q}^{(c)}$ to be submitted in composition c by

$$\mathbb{P}\left(q' \mid c\right) = \frac{\exp\left[p(q', c)\right]}{\sum_{q \in \mathcal{Q}^{(c)}} \exp\left[p(q, c)\right]}. \tag{3.3.3}$$

After signal parameters are inferred, in practice, the simpler term $p(q', c)$ in (3.3.3) is used for re-ranking the pre-indexed query suggestions as described in Section 3.3.1. Since query suggestions are pre-indexed by relevance score s, the re-ranking stage of AppAware is determined by app-related signals in composition c,

which are captured by the last two terms of (3.3.1). We emphasize that, the preference model $p(q, c)$ in (3.3.1) is not constrained to employ any certain relevance score s. We evaluate different settings of s in Section 3.4.

Challenges. App-related signals are noisy. On one hand, for many query–app pairs, a certain installed app may not indicate a higher likelihood of a certain query submission. On the other hand, a less recently opened app may be less relevant to the query intent at the time of a query submission. To overcome the challenge of noisy signals, AppAware optimizes composite objectives with filtering out noisy signals. We describe such composite objectives in Section 3.3.4.

Besides, app signals are voluminous. Recall that signal parameter $\beta_{q,a}$ captures relationships between every query and installed app in the training data set. The number of such parameters can be as large as the product of unique query count and unique app count (20 million in our experiments) plus the maximum count of unique recently opened apps (48 in our experiments within 30 minutes before queries). Hence, processing with respect to all these parameters simultaneously consumes computational resources heavily. To overcome the challenge of voluminous signals, we describe an algorithm to compute lightly with respect to a random signal parameter at each step in Section 3.3.5.

3.3.4 Composite Objectives

As mentioned in Section 3.3.2, for indexing convenience all the signal parameters $\beta_{q,a}$ and β_k from (3.3.1) in any fixed order constitute the signal parameter vector $\mathbf{w}$. Let w_j be the j^{th} element of vector $\mathbf{w}$ of dimension d. We denote the ℓ_1 and ℓ_2 norms of vector $\mathbf{w}$ as $\|\mathbf{w}\|_1 = \sum_{k=1}^{d} |w_k|$ and $\|\mathbf{w}\|_2 = \big(\sum_{k=1}^{d} w_k^2\big)^{1/2}$.

Signal parameter vector $\mathbf{w}$ is to be inferred based on maximum likelihood. To begin with, we want to maximize the following log-likelihood for the set of compositions $\mathcal{C}$ in the training data set with respect to signal parameters:

$$\frac{1}{|\mathcal{C}|} \sum_{c \in \mathcal{C}} \log \mathbb{P}(q^{(c)} \mid c), \tag{3.3.4}$$

where $|\mathcal{C}|$ is the size of $\mathcal{C}$ and $\mathbb{P}(q^{(c)} \mid c)$ is defined in (3.3.2). By (3.3.2) and (3.3.4), an unconstrained optimization problem out of minimizing negative log-likelihood with the ℓ_1 and ℓ_2 norms is obtained:

$$\underset{\mathbf{w}}{\text{minimize}} \qquad \frac{1}{|\mathcal{C}|} \sum_{c \in \mathcal{C}} \Big[\log \sum_{q \in \mathcal{Q}^{(c)} \bigcup \{q^{(c)}\}} \exp\big[p(q, c)\big] - p(q^{(c)}, c) \Big] + \frac{\lambda_2}{2} \|\mathbf{w}\|_2^2 + \lambda_1 \|\mathbf{w}\|_1, \tag{3.3.5}$$

where λ_2 and λ_1 are regularizer weights of ℓ_2 and ℓ_1 norms. Recall that $\beta_{q,a}$ and β_k of $p(q, c)$ in (3.3.1) correspond to $\mathbf{w}$. In (3.3.5), the main purpose of introducing the ℓ_2 norm with $\lambda_2 > 0$ is to guarantee the strong convexity of the objective function in (3.3.5) excluding the last term. We denote the convexity parameter by μ. The ℓ_1 norm is for filtering out noisy signals, which is discussed in detail in Section 3.3.5 (Remark 3.3.1). Rewriting (3.3.5) in the form of a sum of a finite number of functions gives the composite objective problem:

$$\mathbf{w}^* = \underset{\mathbf{w}}{\operatorname{argmin}}\, F(\mathbf{w}) + R(\mathbf{w}), \tag{3.3.6}$$

where $F(\mathbf{w}) = (1/|\mathcal{C}|) \sum_{c \in \mathcal{C}} f_c(\mathbf{w})$ and $R(\mathbf{w}) = \sum_{j=1}^{d} r_j(\mathbf{w})$, where $f_c(\mathbf{w}) = \log \sum_{q \in \mathcal{Q}^{(c)} \bigcup \{q^{(c)}\}} \exp\left[p(q, c)\right] - p(q^{(c)}, c) + (\lambda_2/2)\|\mathbf{w}\|_2^2$ and $r_j(\mathbf{w}) = r_j(w_j) = \lambda_1 |w_j|$. Gradient $\nabla F(\mathbf{w})$ is Lipschitz continuous and we denote the Lipschitz constant by L. Same as $F(\mathbf{w})$, which is the objective function in (3.3.5) excluding the last term, each function $f_c(\mathbf{w})$ is strongly convex with convexity parameter μ. Note that $F(\mathbf{w})$ is a sum of a finite number of strongly convex and smooth functions and $R(\mathbf{w})$ is a general convex function that is non-differentiable. Each element function $f_c(\mathbf{w})$ is a negative log-likelihood function with the ℓ_2 norm for composition c, which is a single element of set $\mathcal{C}$.

3.3.5 Optimization

There are a few issues with optimizing the composite objectives in (3.3.6). Due to the large size of the training data set, an algorithm based on proximal stochastic gradient descent is preferred. However, this has a slower sublinear rate of convergence. Recently, Schmidt *et al.* trained conditional random fields using the stochastic average gradient with a faster linear rate of convergence [135]. In fact, there is another linearly-convergent stochastic variance reduced gradient that has multiple stages with two nested for-loops per iteration [70]. Such a multi-stage algorithm requires a pass through the entire data set per iteration, which is computationally expensive especially when the data set is large. In sharp contrast, the gradient update method by Schmidt *et al.* has a simpler single-stage iteration with only one for-loop and avoids the aforementioned computational complexity from a multi-stage algorithm.

We propose an optimization algorithm in Section 3.3.5 employing the single-stage stochastic average gradient from Schmidt *et al.* [135]. We highlight that their algorithm cannot be directly applied to solve (3.3.6), and our algorithm is distinct from theirs in two main aspects. First, the noisy signal challenge is addressed by optimizing composite objectives with non-differentiable $R(\mathbf{w})$ (details are in Remark 3.3.1), which can be solved by our algorithm but not their algorithm. Second, to overcome the voluminous signal

challenge, our algorithm updates the gradient with respect to only one coordinate per iteration while their algorithm updates the gradient with respect to all coordinates at each iteration. We theoretically guarantee the linear rate of convergence for our algorithm with different proof techniques from those of Schmidt *et al.*

Algorithm

First, initialize signal parameter vector $\mathbf{w}^{(0)}$ at random. Then, for iteration $t = 1, 2, \ldots$, repeat the following:

I Sample mini-batch $\mathcal{B}$ from $\{1, \ldots, |\mathcal{C}|\}$ uniformly at random with replacement.

II Set element signal parameter vector $\boldsymbol{\phi}_c^{(t)}$ to common signal parameter vector $\mathbf{w}^{(t-1)}$ for all $c \in \mathcal{B}$.

III Sample coordinate index j from $\{1, \ldots, d\}$ uniformly at random with replacement.

IV Compute the updated gradient based on the sampled mini-batch with respect to the sampled coordinate

$$g_{\mathcal{B},j}^{(t)} = \nabla_j f_{\mathcal{B}}(\boldsymbol{\phi}_{\mathcal{B}}^{(t)}) - \nabla_j f_{\mathcal{B}}(\boldsymbol{\phi}_{\mathcal{B}}^{(t-1)}) + \frac{1}{|\mathcal{C}|} \sum_{k \in \mathcal{C}} \nabla_j f_k(\boldsymbol{\phi}_k^{(t-1)}), \tag{3.3.7}$$

where by defining $|\mathcal{B}|$ as the size of mini-batch $\mathcal{B}$, for all $\boldsymbol{\phi}$, $f_{\mathcal{B}}(\boldsymbol{\phi}_{\mathcal{B}}) = (1/|\mathcal{B}|) \sum_{c \in \mathcal{B}} f_c(\boldsymbol{\phi}_c)$ and $\nabla_j f(\boldsymbol{\phi}) = [\nabla f(\boldsymbol{\phi})]_j = \partial f(\boldsymbol{\phi}) / \partial \phi_j$.

V Set $w_j^{(t)}$ to $\text{prox}_{\eta,j}(w_j^{(t-1)} - \eta g_{\mathcal{B},j}^{(t)})$, where for all $w, u \in \mathbb{R}$,

$$\text{prox}_{\eta,j}(w) = \underset{u}{\text{argmin}} \frac{1}{2\eta} \|w - u\|_2^2 + r_j(u). \tag{3.3.8}$$

VI Set $\mathbf{w}_{\setminus j}^{(t)}$ to $\mathbf{w}_{\setminus j}^{(t-1)}$, where any subvector of $\mathbf{w}$ excluding w_j is denoted by $\mathbf{w}_{\setminus j}$.

Remark 3.3.1 (Filtering out noisy signals) *The proximal operator in* (5.6.1) *facilitates proof of linear convergence. Without it, a subgradient method only gives a sublinear rate of convergence. There is a closed-form solution to* (5.6.1)*. We emphasize that this solution may clear certain signal parameter values to 0: if* $|w_j^{(t-1)} - \eta g_{\mathcal{B},j}^{(t)}| \leq \eta\lambda_1$, $w_j^{(t)} = 0$*; otherwise* $w_j^{(t)} = w_j^{(t-1)} - \eta g_{\mathcal{B},j}^{(t)} - \eta\lambda_1 (w_j^{(t-1)} - \eta g_{\mathcal{B},j}^{(t)}) / |w_j^{(t-1)} - \eta g_{\mathcal{B},j}^{(t)}|$*. Signal parameters of value 0 indicate that their corresponding noisy signals are filtered out in* (3.3.1)*.*

Remark 3.3.2 (Lighter cost for voluminous signals) *Given the voluminous app-related signals in the training data set (about 20 million in our experiments), updating the gradient of the signal parameter vector with respect to all coordinates consumes computational resources heavily per iteration, such as exceeding the memory budget. Our algorithm enjoys a lighter processing cost than either batch-style proximal gradient descent or any gradient update with respect to all coordinates per iteration. The update at each iteration of the algorithm is based on a mini-batch of element functions with only one coordinate. With a lighter processing cost, this algorithm converges to the global optimum at a linear rate.*

Computational Complexity

Multi-stage algorithms with multiple loops for each iteration requires a pass through the entire data set per iteration [70]. To avoid this high computational complexity, our algorithm is based on a single-stage update with only one loop through $t = 1, 2, \ldots$ [135]. To compare these two techniques for updating the gradient, suppose that both algorithms update the gradient with respect to the same number of element functions and coordinates. At each iteration, the inner loop of the multi-stage algorithm involves a repetitive computation of $\mathcal{O}(|\mathcal{C}|)$ time, where $|\mathcal{C}|$ is the size of the data set (number of element functions). In contrast, the single-stage algorithm requires a computation of $\mathcal{O}(1)$ time per iteration: the last term in (3.3.7) is a distributive function and its update takes a constant time without a need for re-computation at each iteration. For the same problem setting, the iteration complexity of the single-stage algorithm is lower than that of the multi-stage algorithm [135, 70].

In addition, it is notable that at each single-stage iteration, the update in (3.3.7) reduces the variance of the gradient estimator at each iteration with the stochastic average gradient. This results in a faster linear rate of convergence than a sublinear rate of the classic proximal stochastic gradient descent. We theoretically guarantee the linear rate of convergence in Section 3.3.5. Our empirical results in Section 3.4.3 reinforce that with 15 entire data passes, the objective gap value is close to 10^{-4}. Here an entire data pass is a standard measure representing the least possible iterations for passing through the entire data instances with respect to all coordinates [135, 70]. Given $|\mathcal{C}|$ compositions with d coordinates, one entire data pass of the algorithm in Section 3.3.5 is equivalent to $(|\mathcal{C}|d)/|\mathcal{B}|$ iterations, where $|\mathcal{B}|$ is the mini-batch size in the algorithm.

Optimum and Convergence

It is easy to conclude that, the global optimum $\mathbf{w}^*$ exists for the composite objective optimization problem in (3.3.6) because $F(\mathbf{w})$ is strongly convex and $R(\mathbf{w})$ is convex.

However, the theoretical analysis for the rate of convergence of the algorithm is nontrivial. In this subsection and Section 3.7, all the expectations are taken conditional on $\mathbf{w}^{(t-1)}$ and $\boldsymbol{\phi}_c^{(t-1)}$ unless otherwise stated. For the convenience of our analysis, based on (3.3.7), after removal of the coordinate index we define

$$\mathbf{h}_{\mathcal{B}}^{(t)} = \nabla f_{\mathcal{B}}(\boldsymbol{\phi}_{\mathcal{B}}^{(t)}) - \nabla f_{\mathcal{B}}(\boldsymbol{\phi}_{\mathcal{B}}^{(t-1)}) + \frac{1}{|\mathcal{C}|}\sum_{k\in\mathcal{C}} \nabla f_k(\boldsymbol{\phi}_k^{(t-1)}), \tag{3.3.9}$$

$$\mathbf{h}_{c}^{(t)} = \nabla f_{c}(\boldsymbol{\phi}_{c}^{(t)}) - \nabla f_{c}(\boldsymbol{\phi}_{c}^{(t-1)}) + \frac{1}{|\mathcal{C}|}\sum_{k\in\mathcal{C}} \nabla f_k(\boldsymbol{\phi}_k^{(t-1)}), \tag{3.3.10}$$

where $\mathcal{B}$ is a mini-batch uniformly sampled from $\{1, \ldots, |\mathcal{C}|\}$ at random with replacement and $c \in \mathcal{C}$. Before we prove the rate of convergence, we introduce two important lemmas.

Lemma 3.3.3 *For the algorithm in Section 3.3.5, with definitions in* (3.3.9) *and* (3.3.10) *we have*

$$\mathbb{E}_{\mathcal{B}}[\mathbf{h}_{\mathcal{B}}^{(t)}] = \mathbb{E}_c[\mathbf{h}_c^{(t)}] = \nabla F(\mathbf{w}^{(t-1)}).$$

The proof is in Section 3.7.1. Lemma 3.3.3 guarantees that $\mathbf{h}_{\mathcal{B}}^{(t)}$ is an unbiased gradient estimator of F.

Recall that the algorithm in Section 3.3.5 samples a mini-batch of compositions uniformly at random with replacement at every iteration. To facilitate evaluation of expectation terms with respect to randomly sampled mini-batches of compositions, we introduce the following lemma.

Lemma 3.3.4 *For the algorithm in Section 3.3.5 and for all* $\mathbf{x}$ *and* $\mathbf{y}$*,*

$$\begin{aligned}
&\mathbb{E}_{\mathcal{B}}\big[\|\nabla f_{\mathcal{B}}(\mathbf{x}) - \nabla f_{\mathcal{B}}(\mathbf{y})\|^2\big] \\
&= \frac{|\mathcal{B}| \cdot |\mathcal{C}| - |\mathcal{C}|}{|\mathcal{B}| \cdot |\mathcal{C}| - |\mathcal{B}|}\|\nabla F(\mathbf{x}) - \nabla F(\mathbf{y})\|^2 + \frac{|\mathcal{C}| - |\mathcal{B}|}{|\mathcal{B}| \cdot |\mathcal{C}| - |\mathcal{B}|}\mathbb{E}_c\big[\|\nabla f_c(\mathbf{x}) - \nabla f_c(\mathbf{y})\|^2\big].
\end{aligned}$$

Lemma 3.3.4 is proved in Section 3.7.2. Now we present the main theory for bounding the rate of convergence.

Theorem 3.3.5 *The algorithm in Section 3.3.5 is able to converge to the optimal solution at a linear rate.*

We give the detailed proof in Section 3.7.3. The empirical results in Section 3.4.3 agree with our theory that the optimization algorithm converges to the global optimum at a linear rate.

3.4 Evaluation

We comprehensively evaluate the proposed mobile QAC model, AppAware, on a large real-world commercial data set.

3.4.1 Data Description

We describe important details of our collected mobile log data set. Due to the proprietary nature of the data, some details are omitted. The mobile log data set is sampled among 5 months in 2015 and from mobile devices with the Android operating system. All queries are submitted via the search bar of the *Yahoo Aviate* homescreen in Figure 3.1(c). One million compositions are randomly sampled, then tail queries and apps are filtered out: the most popular 10,000 unique queries and most installed 2,000 unique apps (excluding the *Yahoo Aviate* homescreen) remain. The final data set contains 823,421 compositions. In one composition, all keystrokes (with the timestamp of the first keystroke), the submitted query, installed apps at the first

keystroke time, and recently opened apps with timestamps are collected. The maximum count of unique recently opened apps within 30 minutes before queries is 48.

The training and testing data sets are split in an ascending time order: the first and second half of a user's compositions are used for training and testing respectively. All the app-related signals and the relevance scores are standardized: the data standardization procedure is transforming data to zero mean and unit variance.

3.4.2 Experimental Setting

Measures for Accuracy. Mean reciprocal rank (MRR) is a standard measure to evaluate the ranking accuracy of QAC [10, 89, 69, 141, 184]. It is calculated by the average reciprocal of the submitted query's ranking in a suggestion list. Success Rate@top k (SR@k) is the average percentage of the submitted queries that can be found in the top k suggestions during testing. SR@k is also used to evaluate the QAC ranking accuracy [69, 184]. In general, a higher MRR or SR@k indicates a higher ranking accuracy of QAC [10, 89, 69, 141, 20, 184]. The statistical significance of the accuracy improvements is validated by a paired-t test ($p < 0.05$).

Methods for Comparison. The relevance scores with parameter settings in our experiments reuse the existing research as described below. None of these baseline methods uses mobile devices' exclusive signals. Thus, they are referred to as Standard QAC.

- **MPC:** Given an input prefix, Most Popular Completion (MPC) ranks suggested queries based on their historical query frequency counts. A more popular query has a higher rank. It was found competitive by various studies [10, 69, 89, 141].
- **Personal:** Personal QAC by distinguishing different users can achieve a higher accuracy [10, 20, 141]. Here the Personal relevance score is an equal-weighted linear combination of the MPC score and the standardized personal historical query frequency counts as suggested by a study [184].
- **Personal-S:** It is the Personal relevance score with an optimal combination with different weights of the MPC score and the standardized personal query frequency counts. Optimal weights achieving the highest MRR makes Personal-S more competitive.
- **TimeSense:** Time signals are useful in QAC [20, 142, 168]. TimeSense is the same as Personal except that the personal historical query frequency count is replaced by the frequency count of a query from all users within 28 days before a composition [168].

Table 3.5: Accuracy comparison of Standard QAC and AppAware (in percentage). All the boldfaced results denote that the accuracy improvements over Standard QAC are statistically significant ($p < 0.05$) for the same relevance score.

Relevance	MRR		SR@1	
	Std.	AppAware	Std.	AppAware
MPC	35.13	**41.55 (+18.27%)**	27.36	**34.08 (+24.56%)**
Personal	39.06	**43.57 (+11.55%)**	31.32	**37.16 (+18.65%)**
Personal-S	40.48	**44.62 (+10.23%)**	32.70	**38.69 (+18.32%)**
TimeSense	39.91	**43.94 (+10.10%)**	32.79	**38.48 (+17.35%)**
TimeSense-S	40.88	**44.93 (+9.91%)**	34.01	**39.98 (+17.55%)**
Relevance	SR@2		SR@3	
	Std.	AppAware	Std.	AppAware
MPC	37.09	**44.50 (+19.98%)**	41.69	**48.61 (+16.60%)**
Personal	40.52	**46.36 (+14.41%)**	46.21	**50.15 (+8.53%)**
Personal-S	42.53	**47.54 (+11.78%)**	47.53	**50.62 (+6.50%)**
TimeSense	42.10	**46.91 (+11.43%)**	46.83	**49.45 (+5.59%)**
TimeSense-S	43.76	**47.58 (+8.73%)**	47.66	**50.12 (+5.16%)**

*Std.: Standard QAC

- **TimeSense-S:** It is the same as Personal-S except that the Personal score is replaced by the TimeSense score.

We study the effect of varying parameter values in Section 4.4.5. Unless otherwise stated, the time-window size for recently opened apps before query submissions is 30 minutes, the mini-batch size is 100, the pre-indexed query count is 10, the suggested query count is 5 (considering display sizes of mobile devices), and the number of entire data passes is 15. Personal-S and TimeSense-S both linearly combine a MPC score with the optimal weight θ and the other score with the weight $1 - \theta$. The optimal weights in Personal-S and TimeSense-S enable Standard QAC to achieve the highest MRR.

3.4.3 Experimental Results

We perform comprehensive experiments to evaluate the performance of the proposed AppAware model. We first compare methods employing different relevance scores in Section 3.4.3. Then throughout the remaining Section 3.4.3—3.4.3, we study different general properties of AppAware by fixing the relevance score to MPC; the results with the other relevance scores are similar.

Boosting the Accuracy of Standard QAC with App-related Signals on Mobile Devices

Table 3.5 presents the accuracy comparison of Standard QAC and AppAware with different relevance scores as described in Section 3.4.2. All the boldfaced results denote that the accuracy improvements over Standard QAC are statistically significant ($p < 0.05$) for the same relevance score. We highlight that, for each same relevance score, mobile devices' exclusive signals of installed apps and recently opened apps significantly and

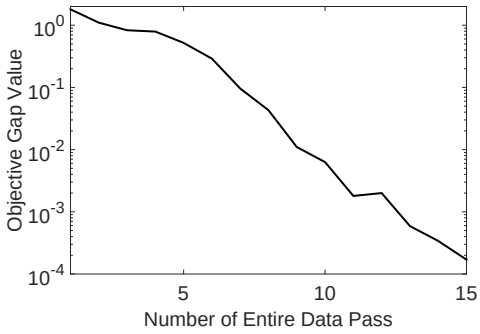


Figure 3.3: Convergence study.

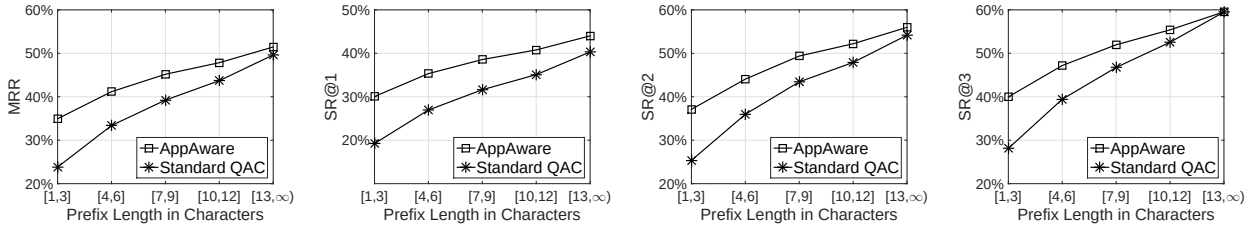


Figure 3.4: Accuracy comparison of AppAware and Standard QAC for prefixes with varying lengths.

consistently boost the accuracy of these Standard QAC models that do not use exclusive signals of mobile devices. For instance, for the same MPC relevance score, signals of installed apps and recently opened apps significantly boost Standard QAC by 18.27% in MRR. Such an improvement is significant across all the different accuracy measures.

When relevance scores become more accurate, such as Personal and TimeSense in comparison with MPC, AppAware also ranks query suggestions more accurately. Given the relevance scores with different parameter settings (Personal *vs.* Personal-S and TimeSense *vs.* TimeSense-S), AppAware has slightly varying accuracy. Such variance depends on the accuracy of the relevance scores for the chosen parameter values. We conclude that, installed app and recently opened app signals are useful in boosting the accuracy of such existing Standard QAC models on mobile devices.

Convergence Study

In Section 3.3.5 we theoretically prove that the rate of convergence for AppAware is linear. Our theory is reinforced by the experimental results averaged over 50 replications in Figure 3.3. The objective gap value is $[F(\mathbf{w}) + R(\mathbf{w})] - [F(\mathbf{w}^*) + R(\mathbf{w}^*)]$ in log scale, where $F(\mathbf{w}) + R(\mathbf{w})$ are the composite objectives and $\mathbf{w}^*$ is the global optimum in (3.3.6). Recall the definition of the entire data pass in Section 3.3.5, AppAware

converges fast by using the single-stage randomized coordinate descent with mini-batches. With iterations of 15 entire data passes, the objective gap value is close to 10^{-4}.

Varying-Length Prefix Study

We study the performance of AppAware and Standard QAC for prefixes with varying lengths. We group prefixes into five bins according to their lengths in characters. The ranking accuracy of AppAware and Standard QAC is evaluated on prefixes from the same bin. Figure 3.4 illustrates the ranking accuracy comparison of AppAware and Standard QAC for prefixes of varying lengths. It is interesting to observe that accuracy improvements by app-related signals are not constant with respect to varying-length prefixes.

In general, when prefixes are shorter, the accuracy gap between AppAware and Standard QAC is larger across different accuracy measures. So, installed app and recently opened app signals take better effect in boosting accuracy of Standard QAC when handling more challenging scenarios of shorter input prefixes. This may be explained by the declining challenges for longer prefixes due to a reduction of the matched queries: Standard QAC is more accurate for such cases and it is harder to make further improvements.

App-Related Signal Study

AppAware makes use of two types of exclusive signals to mobile devices: installed apps and recently opened apps. To more comprehensively study such signals, we compare two variants of AppAware using different subsets of such signals: installed app signals only and recently opened app signals only. In addition, we introduce another "case-by-case" variant: it uses recently opened app signals only when they exist, otherwise uses installed app signals only. The results are compared in Figure 3.5.

Although both types of signals are able to improve the ranking accuracy of Standard QAC alone, recently opened app signals are slightly better at predicting query intents than installed app signals on mobile devices. Since recently opened app signals do not always exist, the "case-by-case" variant is slightly more accurate than the variant using recently opened apps only. When recently opened app signals exist, the "case-by-case" variant uses such signals only; while AppAware integrates extra installed app signals. To illustrate, even though some apps are recently opened before query submissions, these queries may still be related to installed app signals only or both types of signals. Being capable of modeling all such potential scenarios, AppAware achieves the highest accuracy across different measures in comparison with its variants.

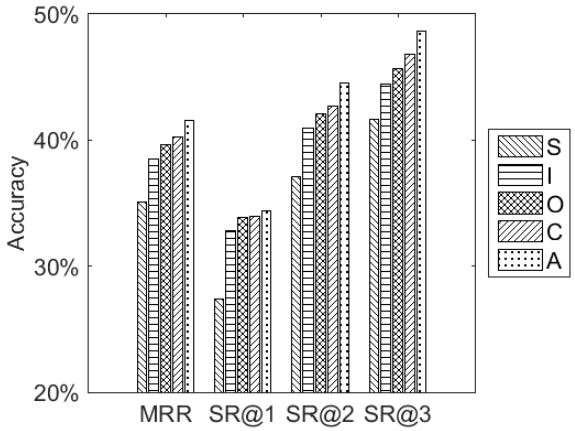


Figure 3.5: AppAware achieves the highest accuracy in comparison with its variants (S: Standard QAC; I: AppAware variant using installed app signals only; O: AppAware variant using recently opened app signals only; C: AppAware "case-by-case" variant using recently opened app signals only when they exist, otherwise using installed app signals only; A: AppAware).

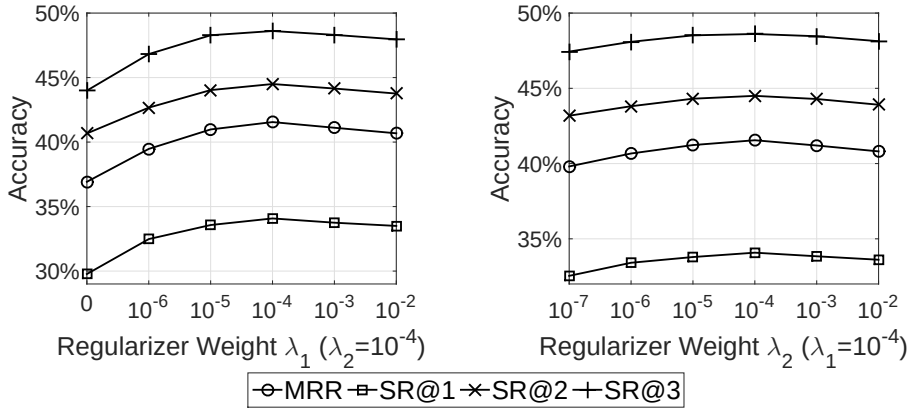


Figure 3.6: Regularizer weight study.

Regularization Study

Figure 3.6 plots the accuracy measures of AppAware with varying regularizer weights λ_1 (left) and λ_2 (right). We vary the value of one regularizer weight while fixing that of the other at 10^{-4}.

It is noteworthy from Figure 3.6 (left) that the accuracy is highest when $\lambda_1 = 10^{-4}$ but degrades sharply when $\lambda_1 = 0$. It empirically corroborates the effect of the ℓ_1 norm in filtering out noisy signals. When λ_1 gets smaller than 10^{-4}, the accuracy is lower due to a lighter penalty applied to signal parameters associated with noisy signals. However, when λ_1 is greater than 10^{-4}, a heavier penalty may suppress useful signals and result in a slightly lower accuracy.

Recall Section 3.3.4 that λ_2 must be positive to ensure the strong convexity of $F(\mathbf{w})$ in (3.3.6) to guarantee the linear convergence of the optimization algorithm. In Figure 3.6 (right), the highest accuracy is attained

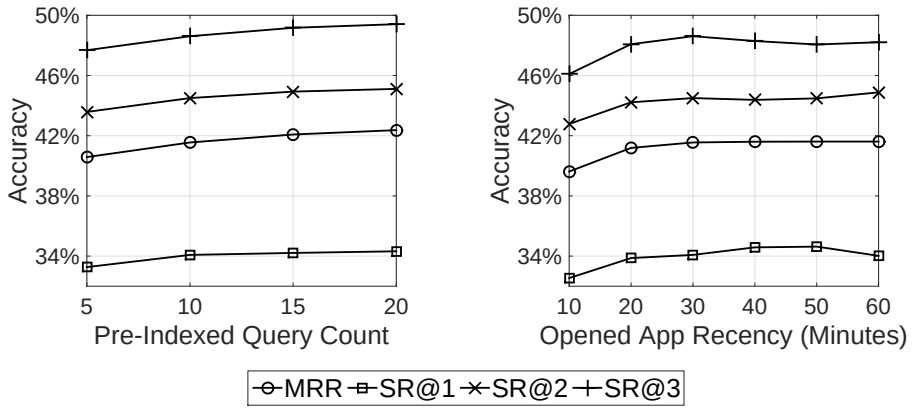


Figure 3.7: Pre-indexed query count (left) and opened app recency (right) studies.

when $\lambda_2 = 10^{-4}$. Note that the accuracy for varying λ_1 and λ_2 is stable around the optimum 10^{-4}, such as between 10^{-5} and 10^{-3}. This eases parameter tuning.

Pre-Indexed Query Count Study

Figure 3.7 (left) illustrates the growing accuracy of AppAware with more pre-indexed queries for re-ranking. This is because fewer pre-indexed queries may exclude users' potential submissions. However, re-ranking more queries is computationally more expensive. Several studies showed that re-ranking 10 pre-indexed queries is feasible in practice [141, 184] and the outperforming of AppAware is obtained with the pre-indexed query count set to 10 in Section 3.4.3.

Opened App Recency Study

Figure 3.7 (right) plots the accuracy measures of AppAware when recently opened apps come from time-windows of varying sizes before query submissions. The regularizer weights are optimal for achieving the highest MRR. On one hand, when the time-window size is smaller, all the accuracy measures are consistently lower because useful recently opened app signals are fewer. On the other hand, when its size gets larger, such as larger than 30 minutes, some measures rise slightly while some other ones start to fall. To explain, for those apps that are opened less recently, they may be less relevant to the query intents at the time of query submissions.

3.5 Related Work

QAC has received a growing attention in recent years, such as popularity-based QAC using historical frequency count signals [10], time-based QAC using time signals [142, 168], context-based QAC using user

previous query signals [10], and personalized QAC using user profile signals [141]. The relevance scores evaluated in this work make use of the existing research, such as MPC [10, 69, 89, 141], Personal(-S) [10, 20, 141], and TimeSense(-S) [20, 142, 168, 112]. More recent QAC methods also predicted the likelihood that suggested queries would be selected by users based on keystroke behaviors during query compositions [89, 184, 87], determined suggestion rankings based on query reformulation signals [69], exploited web content signals [81], or combined signals such as time and previous queries from users [20]. Specifically, Zhang *et al.* proposed adaQAC, an adaptive QAC model incorporating users' implicit negative feedback [184]. Other aspects of QAC have also been studied, such as user interactions with QAC [111, 61], space efficient indexing [66], and spelling error tolerance [26, 68, 41, 171]. However, none of the aforementioned work aimed at specifically solving the mobile QAC problem by exploiting mobile devices' exclusive signals. We take the initiative to show that mobile QAC can be more accurate by employing mobile app-related signals.

The idea of using mobile app-related signals for mobile QAC is inspired by a recent mobile app usage prediction work of Baeza-Yates *et al.* [9]. Their model used signals of relations between sequentially opened apps via the Android API. Our work answers an important open question on whether sequentially submitted queries and opened apps can boost the QAC accuracy on mobile devices.

Mobile app recommendation and usage were also studied with respect to app replacement behaviors [181], security preferences [196, 97], version descriptions [94], personalized signal discovery [92], implicit feedback [36], serendipitous apps [12], and many other aspects [31, 33, 167, 175, 178]. A joint research of both mobile queries and mobile apps sets our work apart from these studies.

3.6 Conclusion

Users tend to rely on QAC more heavily on mobile devices than on desktops. Motivated by its importance, we studied the new mobile QAC problem to exploit mobile devices' exclusive signals. We proposed a novel AppAware model employing installed app and recently opened app signals. To overcome the challenge of such noisy and voluminous signals, AppAware optimizes composite objectives at a lighter processing cost. Our algorithm converges to the global optimum at a linear rate with a theoretical guarantee. Experiments demonstrated high efficiency and effectiveness of AppAware.

Our study has provided a number of new insights that we hope will have general applicability to recommendation and search strategies on mobile devices (*e.g.,* mobile shopping and mobile search), to future models of mobile QAC, and to efficient optimization.

3.7 Proof

We provide the proof for all the lemmas and theorems (see Section 3.3) as follows.

3.7.1 Proof of Lemma 3.3.3

Proof. We start by analyzing the first two terms in (3.3.9). For all $\mathbf{w}$ we have $\mathbb{E}_{\mathcal{B}}\left[\nabla f_{\mathcal{B}}(\mathbf{w})\right] = \mathbb{E}_{\mathcal{B}}\big[(1/|\mathcal{B}|)$ $\sum_{c\in\mathcal{B}} \nabla f_c(\mathbf{w})\big]$.

By switching the order of selection in formulating mini-batches, we take expectation with respect to mini-batches and obtain

$$\begin{aligned}
\mathbb{E}_{\mathcal{B}}\left[\nabla f_{\mathcal{B}}(\mathbf{w})\right] &= \frac{1}{|\mathcal{B}|\binom{|\mathcal{C}|}{|\mathcal{B}|}} \sum_{i=1}^{\binom{|\mathcal{C}|}{|\mathcal{B}|}} \sum_{c\in\mathcal{B}_i} \nabla f_c(\mathbf{w}) \\
&= \frac{1}{|\mathcal{B}|\binom{|\mathcal{C}|}{|\mathcal{B}|}} \sum_{c\in\mathcal{C}} \tbinom{|\mathcal{C}|-1}{|\mathcal{B}|-1} \nabla f_c(\mathbf{w}) \\
&= \frac{1}{|\mathcal{C}|} \sum_{c\in\mathcal{C}} \nabla f_c(\mathbf{w}).
\end{aligned}$$

For all $\mathbf{w}$, it holds that $\mathbb{E}_{\mathcal{B}}\left[\nabla f_{\mathcal{B}}(\mathbf{w})\right] = \mathbb{E}_c\left[\nabla f_c(\mathbf{w})\right] = \nabla F(\mathbf{w})$. By the definition of $\mathbf{h}_{\mathcal{B}}^{(t)}$ and $\mathbf{h}_c^{(t)}$ in (3.3.9) and (3.3.10),

$$\begin{aligned}
&\mathbb{E}_{\mathcal{B}}[\mathbf{h}_{\mathcal{B}}^{(t)}] = \mathbb{E}_c[\mathbf{h}_c^{(t)}] \\
&= \mathbb{E}_c\big[\nabla f_c(\boldsymbol{\phi}_c^{(t)}) - \nabla f_c(\boldsymbol{\phi}_c^{(t-1)})\big] + \frac{1}{|\mathcal{C}|}\sum_{k\in\mathcal{C}} \nabla f_k(\boldsymbol{\phi}_k^{(t-1)}) \\
&= \frac{1}{|\mathcal{C}|}\sum_{c\in\mathcal{C}} \nabla f_c(\mathbf{w}^{(t-1)}) - \frac{1}{|\mathcal{C}|}\sum_{c\in\mathcal{C}} \nabla f_c(\boldsymbol{\phi}_c^{(t-1)}) + \frac{1}{|\mathcal{C}|}\sum_{k\in\mathcal{C}} \nabla f_k(\boldsymbol{\phi}_k^{(t-1)}) \\
&= \nabla F(\mathbf{w}^{(t-1)}).
\end{aligned}$$

■

3.7.2 Proof of Lemma 3.3.4

Proof. Following the mini-batch definition in the algorithm in Section 3.3.5 and for all $\mathbf{x}$ and $\mathbf{y}$, we have

$$\begin{aligned}
&\mathbb{E}_{\mathcal{B}}\big[\|\nabla f_{\mathcal{B}}(\mathbf{x}) - \nabla f_{\mathcal{B}}(\mathbf{y})\|^2\big] \\
&= \frac{1}{|\mathcal{B}|^2}\mathbb{E}_{\mathcal{B}}\bigg[\Big\|\sum_{c\in\mathcal{B}} \nabla f_c(\mathbf{x}) - \nabla f_c(\mathbf{y})\Big\|^2\bigg] \\
&= \frac{1}{|\mathcal{B}|^2}\mathbb{E}_{\mathcal{B}}\Big[\sum_{c\neq c'\in\mathcal{B}} \langle \nabla f_c(\mathbf{x}) - \nabla f_c(\mathbf{y}), \nabla f_{c'}(\mathbf{x}) - \nabla f_{c'}(\mathbf{y})\rangle\Big] + \frac{|\mathcal{B}|}{|\mathcal{B}|^2}\mathbb{E}_c\big[\|\nabla f_c(\mathbf{x}) - \nabla f_c(\mathbf{y})\|^2\big]. \qquad (3.7.1)
\end{aligned}$$

By switching the order of selection in formulating mini-batches, we take expectation with respect to mini-batches and obtain

$$\begin{aligned}
&\frac{1}{|\mathcal{B}|^2}\mathbb{E}_{\mathcal{B}}\Big[\sum_{c\neq c'\in\mathcal{B}} \langle \nabla f_c(\mathbf{x}) - \nabla f_c(\mathbf{y}), \nabla f_{c'}(\mathbf{x}) - \nabla f_{c'}(\mathbf{y})\rangle\Big] \\
&= \frac{1}{|\mathcal{B}|^2\binom{|\mathcal{C}|}{|\mathcal{B}|}} \sum_{i=1}^{\binom{|\mathcal{C}|}{|\mathcal{B}|}} \sum_{c\neq c'\in\mathcal{B}_i} \langle \nabla f_c(\mathbf{x}) - \nabla f_c(\mathbf{y}), \nabla f_{c'}(\mathbf{x}) - \nabla f_{c'}(\mathbf{y})\rangle \\
&= \frac{1}{|\mathcal{B}|^2\binom{|\mathcal{C}|}{|\mathcal{B}|}} \sum_{c\neq c'\in\mathcal{C}} \tbinom{|\mathcal{C}|-2}{|\mathcal{B}|-2} \langle \nabla f_c(\mathbf{x}) - \nabla f_c(\mathbf{y}), \nabla f_{c'}(\mathbf{x}) - \nabla f_{c'}(\mathbf{y})\rangle \\
&= \frac{|\mathcal{B}|-1}{|\mathcal{B}|\cdot|\mathcal{C}|(|\mathcal{C}|-1)} \sum_{c\neq c'\in\mathcal{C}} \langle \nabla f_c(\mathbf{x}) - \nabla f_c(\mathbf{y}), \nabla f_{c'}(\mathbf{x}) - \nabla f_{c'}(\mathbf{y})\rangle. \qquad (3.7.2)
\end{aligned}$$

Note that the right-hand size of (3.7.2) does not depend on expectation with respect to randomly sampled mini-batches.

Now we go on to replace term (3.7.1) with the right-hand side of the results in (3.7.2). Then we further obtain

$$\begin{aligned}
&\mathbb{E}_{\mathcal{B}}\big[\|\nabla f_{\mathcal{B}}(\mathbf{x}) - \nabla f_{\mathcal{B}}(\mathbf{y})\|^2\big] \\
&= \frac{|\mathcal{B}|-1}{|\mathcal{B}|\cdot|\mathcal{C}|(|\mathcal{C}|-1)} \sum_{c\neq c'\in\mathcal{C}} \langle \nabla f_c(\mathbf{x}) - \nabla f_c(\mathbf{y}), \nabla f_{c'}(\mathbf{x}) - \nabla f_{c'}(\mathbf{y})\rangle + \frac{1}{|\mathcal{B}|}\mathbb{E}_c\big[\|\nabla f_c(\mathbf{x}) - \nabla f_c(\mathbf{y})\|^2\big] \\
&= \frac{|\mathcal{B}|-1}{|\mathcal{B}|\cdot|\mathcal{C}|(|\mathcal{C}|-1)} \sum_{c,c'\in\mathcal{C}} \langle \nabla f_c(\mathbf{x}) - \nabla f_c(\mathbf{y}), \nabla f_{c'}(\mathbf{x}) - \nabla f_{c'}(\mathbf{y})\rangle \\
&\quad - \Big(\frac{|\mathcal{B}|-1}{|\mathcal{B}|(|\mathcal{C}|-1)} - \frac{1}{|\mathcal{B}|}\Big)\mathbb{E}_c\big[\|\nabla f_c(\mathbf{x}) - \nabla f_c(\mathbf{y})\|^2\big] \\
&= \frac{|\mathcal{B}|\cdot|\mathcal{C}| - |\mathcal{C}|}{|\mathcal{B}|\cdot|\mathcal{C}| - |\mathcal{B}|}\|\nabla F(\mathbf{x}) - \nabla F(\mathbf{y})\|^2 + \frac{|\mathcal{C}| - |\mathcal{B}|}{|\mathcal{B}|\cdot|\mathcal{C}| - |\mathcal{B}|}\mathbb{E}_c\big[\|\nabla f_c(\mathbf{x}) - \nabla f_c(\mathbf{y})\|^2\big],
\end{aligned}$$

where the last equality is obtained by the relation $\big[(|\mathcal{B}|-1)/[|\mathcal{B}|\cdot|\mathcal{C}|(|\mathcal{C}|-1)]\big]\|\sum_{c\in\mathcal{C}}\nabla f_c(\mathbf{x}) - \nabla f_c(\mathbf{y})\|^2 = \big[(|\mathcal{B}|\cdot|\mathcal{C}| - |\mathcal{C}|)/(|\mathcal{B}|\cdot|\mathcal{C}| - |\mathcal{B}|)\big]\|\nabla F(\mathbf{x}) - \nabla F(\mathbf{y})\|^2$. ∎

3.7.3 Proof of Theorem 3.3.5

Proof. We refer to $\mathbf{h}_{\mathcal{B}}^{(t)}$ and $\mathbf{h}_{c}^{(t)}$ defined in (3.3.9) and (3.3.10). By the orthogonality property for non-overlapped coordinates, the non-expansiveness of the proximal operator [120], and that $\mathbf{w}^*$ is the global optimum in (3.3.6), we have

$$\begin{aligned}
&\mathbb{E}_j\big[\|\mathbf{w}^{(t)} - \mathbf{w}^*\|_2^2\big] \\
&= \frac{(d-1)}{d}\|\mathbf{w}^{(t-1)} - \mathbf{w}^*\|_2^2 + \frac{1}{d}\big\|\mathrm{prox}_{\eta}(\mathbf{w}^{(t-1)} - \eta\mathbf{h}_{\mathcal{B}}^{(t)}) - \mathrm{prox}_{\eta}(\mathbf{w}^* - \eta\nabla F(\mathbf{w}^*))\big\|_2^2 \\
&\leq \frac{1}{d}\Big[(d-1)\|\mathbf{w}^{(t-1)} - \mathbf{w}^*\|_2^2 + \|\mathbf{w}^{(t-1)} - \eta\mathbf{h}_{\mathcal{B}}^{(t)} - \mathbf{w}^* + \eta\nabla F(\mathbf{w}^*)\|_2^2\Big].
\end{aligned}$$

After applying the results of Lemma 3.3.3, with a further simplification of terms, we can get

$$\begin{aligned}
&\mathbb{E}_{\mathcal{B},j}\big[\|\mathbf{w}^{(t)} - \mathbf{w}^*\|_2^2\big] = \mathbb{E}_{\mathcal{B}}\Big[\mathbb{E}_j\big[\|\mathbf{w}^{(t)} - \mathbf{w}^*\|_2^2\big]\Big] \\
&\leq \frac{1}{d}\mathbb{E}_{\mathcal{B}}\big[(d-1)\|\mathbf{w}^{(t-1)} - \mathbf{w}^*\|_2^2 + \|\mathbf{w}^{(t-1)} - \eta\mathbf{h}_{\mathcal{B}}^{(t)} - \mathbf{w}^* + \eta\nabla F(\mathbf{w}^*)\|_2^2\big] \\
&= \frac{1}{d}\Big[(d-1)\|\mathbf{w}^{(t-1)} - \mathbf{w}^*\|_2^2 + \|\mathbf{w}^{(t-1)} - \mathbf{w}^*\|_2^2 - 2\eta\langle\nabla F(\mathbf{w}^{(t-1)}) - \nabla F(\mathbf{w}^*), \mathbf{w}^{(t-1)} - \mathbf{w}^*\rangle \\
&\quad + \eta^2\mathbb{E}_{\mathcal{B}}\big[\|\mathbf{h}_{\mathcal{B}}^{(t)} - \nabla F(\mathbf{w}^*)\|_2^2\big]\Big].
\end{aligned}$$

Now we use the property that $\mathbb{E}\big[\|\mathbf{x}\|_2^2\big] = \mathbb{E}\big[\|\mathbf{x} - \mathbb{E}[\mathbf{x}]\|_2^2\big] + \big\|\mathbb{E}[\mathbf{x}]\big\|^2$ for all $\mathbf{x}$ and the property that $\|\mathbf{x}+\mathbf{y}\|_2^2 \leq (1+\zeta)\|\mathbf{x}\|_2^2 + (1+\zeta^{-1})\|\mathbf{y}\|_2^2$ for all $\mathbf{x}, \mathbf{y}$, and $\zeta > 0$. It holds that $\mathbb{E}_{\mathcal{B}}\big[\|\mathbf{h}_{\mathcal{B}}^{(t)} - \nabla F(\mathbf{w}^*)\|_2^2\big] \leq (1+\zeta)\mathbb{E}_{\mathcal{B}}\big[\|\nabla f_{\mathcal{B}}(\mathbf{w}^{(t-1)}) - \nabla f_{\mathcal{B}}(\mathbf{w}^*)\|_2^2\big] - \zeta\|\nabla F(\mathbf{w}^{(t-1)}) - \nabla F(\mathbf{w}^*)\|_2^2 + (1+\zeta^{-1})\mathbb{E}_{\mathcal{B}}\big[\|\nabla f_{\mathcal{B}}(\boldsymbol{\phi}_{\mathcal{B}}^{(t-1)}) - \nabla f_{\mathcal{B}}(\mathbf{w}^*)\|_2^2\big]$. Therefore, we have

$$\begin{aligned}
\mathbb{E}_{\mathcal{B},j}\big[\|\mathbf{w}^{(t)} - \mathbf{w}^*\|_2^2\big] \leq \frac{1}{d}\Big[&d\|\mathbf{w}^{(t-1)} - \mathbf{w}^*\|_2^2 + 2\eta\langle\nabla F(\mathbf{w}^*), \mathbf{w}^{(t-1)} - \mathbf{w}^*\rangle - 2\eta\langle\nabla F(\mathbf{w}^{(t-1)}), \mathbf{w}^{(t-1)} - \mathbf{w}^*\rangle \\
&+ \eta^2(1+\zeta)\mathbb{E}_{\mathcal{B}}\big[\|\nabla f_{\mathcal{B}}(\mathbf{w}^{(t-1)}) - \nabla f_{\mathcal{B}}(\mathbf{w}^*)\|_2^2\big] + \eta^2(1+\zeta^{-1})\mathbb{E}_{\mathcal{B}}\big[\|\nabla f_{\mathcal{B}}(\boldsymbol{\phi}_{\mathcal{B}}^{(t-1)}) - \nabla f_{\mathcal{B}}(\mathbf{w}^*)\|_2^2\big] \\
&- \eta^2\zeta\|\nabla F(\mathbf{w}^{(t-1)}) - \nabla F(\mathbf{w}^*)\|_2^2\Big]. \qquad (3.7.3)
\end{aligned}$$

Lemma 3.3.4 is used to replace the two expectation terms with respect to mini-batches on the right-hand side of (4.7.3). By the property of any function f that is convex and has a Lipschitz continuous gradient with constant L: $f(\mathbf{y}) \geq f(\mathbf{x}) + \langle\nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x}\rangle + \|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|_2^2/(2L)$ for all $\mathbf{x}$ and $\mathbf{y}$ [119], we can

further simplify (4.7.3) and multiply it by a positive constant κ:

$$\begin{aligned}\kappa\mathbb{E}_{\mathcal{B},j}\big[\|\mathbf{w}^{(t)}-\mathbf{w}^*\|_2^2\big] \leq{}& \frac{\kappa(d-\eta\mu)}{d}\|\mathbf{w}^{(t-1)}-\mathbf{w}^*\|_2^2+\left(\frac{\kappa\eta^2(1+\zeta)(|\mathcal{B}|\cdot|\mathcal{C}|-|\mathcal{C}|)}{d(|\mathcal{B}|\cdot|\mathcal{C}|-|\mathcal{B}|)}-\frac{\kappa\eta^2\zeta}{d}\right)\|\nabla F(\mathbf{w}^{(t-1)})\\ &-\nabla F(\mathbf{w}^*)\|_2^2+\frac{2\kappa L\eta^2(1+\zeta^{-1})}{d}\left[\frac{1}{|\mathcal{C}|}\sum_{c\in\mathcal{C}}f_c(\boldsymbol{\phi}_c^{(t-1)})-F(\mathbf{w}^*)-\frac{1}{|\mathcal{C}|}\sum_{c\in\mathcal{C}}\langle\nabla f_c(\mathbf{w}^*),\boldsymbol{\phi}_c^{(t-1)}-\mathbf{w}^*\rangle\right]\\ &+\left(\frac{\kappa\eta^2(1+\zeta)(|\mathcal{C}|-|\mathcal{B}|)}{d(|\mathcal{B}|\cdot|\mathcal{C}|-|\mathcal{B}|)}-\frac{\kappa\eta}{dL}\right)\mathbb{E}_c\big[\|\nabla f_c(\mathbf{w}^{(t-1)})-\nabla f_c(\mathbf{w}^*)\|_2^2\big]\\ &-\frac{2\kappa(L-\mu)\eta}{dL}\big[F(\mathbf{w}^{(t-1)})-F(\mathbf{w}^*)-\langle\nabla F(\mathbf{w}^*),\mathbf{w}^{(t-1)}-\mathbf{w}^*\rangle\big].\end{aligned} \tag{3.7.4}$$

By the property of any strongly convex function f with the convexity parameter μ that $f(\mathbf{y}) \leq f(\mathbf{x}) + \langle\nabla f(\mathbf{x}), \mathbf{y}-\mathbf{x}\rangle + \|\nabla f(\mathbf{x})-\nabla f(\mathbf{y})\|_2^2/(2\mu)$ for all $\mathbf{x}$ and $\mathbf{y}$ [119], we have $-\|\nabla F(\mathbf{w}^{(t-1)})-\nabla F(\mathbf{w}^*)\|_2^2 \leq -2\mu\big[F(\mathbf{w}^{(t-1)})-F(\mathbf{w}^*)-\langle\nabla F(\mathbf{w}^*),\mathbf{w}^{(t-1)}-\mathbf{w}^*\rangle\big]$.

With defining $Y_{\mathcal{B}}^{(t)} = (1/|\mathcal{C}|)\cdot\big[\sum_{c\in\mathcal{B}} f_c(\boldsymbol{\phi}_c^{(t)})+\sum_{c\notin\mathcal{B}\wedge c\in\mathcal{C}} f_c(\boldsymbol{\phi}_c^{(t)})\big]-F(\mathbf{w}^*)-(1/|\mathcal{C}|)\cdot\big[\sum_{c\in\mathcal{B}}\langle\nabla f_c(\mathbf{w}^*),\boldsymbol{\phi}_c^{(t)}-\mathbf{w}^*\rangle+\sum_{c\notin\mathcal{B}\wedge c\in\mathcal{C}}\langle\nabla f_c(\mathbf{w}^*),\boldsymbol{\phi}_c^{(t)}-\mathbf{w}^*\rangle\big]+\kappa\|\mathbf{w}^{(t)}-\mathbf{w}^*\|_2^2$, by (3.7.4) and for all $\alpha>0$, we obtain $\mathbb{E}_{\mathcal{B},j}[Y_{\mathcal{B}}^{(t)}]-\alpha Y_{\mathcal{B}}^{(t-1)} \leq \sum_{k=1}^4\rho_k\tau_k$, where the four constants are $\rho_1 = (\kappa/d)\cdot\big[\eta^2(1+\zeta)(|\mathcal{C}|-|\mathcal{B}|)/(|\mathcal{B}|\cdot|\mathcal{C}|-|\mathcal{B}|)-\eta/L\big]$, $\rho_2 = |\mathcal{B}|/|\mathcal{C}| + \big[2\kappa\eta^2\mu(1+\zeta)(|\mathcal{B}|\cdot|\mathcal{C}|-|\mathcal{C}|)\big]/\big[d(|\mathcal{B}|\cdot|\mathcal{C}|-|\mathcal{B}|)\big] - 2\kappa\eta^2\mu\zeta/d - 2\kappa(L-\mu)\eta/(dL)$, $\rho_3 = \kappa(1-\eta\mu/d-\alpha)$, and $\rho_4 = 2\kappa L\eta^2(1+\zeta^{-1})/d-\alpha+(|\mathcal{C}|-|\mathcal{B}|)/|\mathcal{C}|$; and their associated terms are $\tau_1 = \mathbb{E}_c\big[\|\nabla f_c(\mathbf{w}^{(t-1)})-\nabla f_c(\mathbf{w}^*)\|_2^2\big]$, $\tau_2 = F(\mathbf{w}^{(t-1)})-F(\mathbf{w}^*)-\langle\nabla F(\mathbf{w}^*),\mathbf{w}^{(t-1)}-\mathbf{w}^*\rangle$, $\tau_3 = \|\mathbf{w}^{(t-1)}-\mathbf{w}^*\|_2^2$, and $\tau_4 = (1/|\mathcal{C}|)\sum_{c\in\mathcal{C}} f_c(\boldsymbol{\phi}_c^{(t-1)}) - F(\mathbf{w}^*) - (1/|\mathcal{C}|)\sum_{c\in\mathcal{C}}\langle\nabla f_c(\mathbf{w}^*),\boldsymbol{\phi}_c^{(t-1)}-\mathbf{w}^*\rangle$.

It is obvious that $\tau_1 \geq 0$ and $\tau_3 \geq 0$. By the convexity property of F, $\tau_2 \geq 0$ and $\tau_4 \geq 0$. For the step size, we choose

$$\eta = \frac{|\mathcal{B}|\cdot|\mathcal{C}|-|\mathcal{B}|}{2(L+|\mathcal{C}|\mu)(|\mathcal{C}|-|\mathcal{B}|)}.$$

To ensure $0<\eta\mu<1$, we choose a mini-batch size satisfying

$$1 \leq |\mathcal{B}| < \frac{2|\mathcal{C}|(|\mathcal{C}|\mu+L)}{2(|\mathcal{C}|\mu+L)+(|\mathcal{C}|\mu-\mu)}.$$

By setting $\rho_1 = 0$ with $\zeta = (L+2|\mathcal{C}|\mu)/L > 0$, $\rho_2 = 0$ with $\kappa = (|\mathcal{B}|d)/\big[2|\mathcal{C}|\eta(1-\eta\mu)\big] > 0$, and $\rho_3 = 0$ with $\alpha = 1-(\eta\mu)/d$, we have $\rho_4 \leq 0$. Thus, $\mathbb{E}_{\mathcal{B},j}[Y_{\mathcal{B}}^{(t)}] - \alpha Y_{\mathcal{B}}^{(t-1)} \leq 0$, where the expectation is conditional on information from the previous iteration $t-1$. Taking expectation with this previous iteration gives $\mathbb{E}_{\mathcal{B},j}[Y_{\mathcal{B}}^{(t)}] \leq \alpha\mathbb{E}_{\mathcal{B},j}[Y_{\mathcal{B}}^{(t-1)}]$. By chaining over t, $\mathbb{E}_{\mathcal{B},j}[Y_{\mathcal{B}}^{(t)}] \leq \alpha^t Y_{\mathcal{B}}^{(0)}$. Since $\kappa\|\mathbf{w}^{(t)}-\mathbf{w}^*\|_2^2 \leq Y_{\mathcal{B}}^{(t)}$ (note that the sum of the first three terms in $Y_{\mathcal{B}}^{(t)}$ is non-negative by the convexity property of F), given the parameter settings above, for the composite objectives in (3.3.6) and the optimization algorithm in Section 3.3.5, we

have $\mathbb{E}_{\mathcal{B},j}\left[\|\mathbf{w}^{(t)} - \mathbf{w}^*\|_2^2\right] \leq \alpha^t(C_1 + C_2/\kappa)$, where C_1 and C_2 are constants determined by $\mathbf{w}^{(0)}$. Note that $0 < \alpha < 1$. The algorithm in Section 3.3.5 has a linear rate of convergence. ■

Chapter 4

Stochastic Optimization for Big Data Analysis: Strongly Convex Objectives

This chapter focuses on the algorithm design and considers the solver to the optimization problems in Chapters 2 and 3 where the objectives are strongly convex objectives. To be precise, we study the composite minimization problem where the objective function is the sum of two convex functions: one is the sum of a finite number of strongly convex and smooth functions, and the other is a general convex function that is non-differentiable. Specifically, we consider the case where the non-differentiable function is block separable and admits a simple proximal mapping for each block. This type of composite optimization is common in many data mining and machine learning problems, and can be solved by block coordinate descent algorithms. We propose an accelerated stochastic block coordinate descent (ASBCD) algorithm, which incorporates the incrementally averaged partial derivative into the stochastic partial derivative (variance reduction technique) and exploits optimal sampling. We prove that ASBCD attains a linear rate of convergence. In contrast to uniform sampling, we reveal that the optimal non-uniform sampling can be employed to achieve a lower iteration complexity. Experimental results on different large-scale real data sets support our theory.

4.1 Introduction

We consider the problem of minimizing a composite function, which is the sum of two convex functions:

$$\mathbf{w}^* = \underset{\mathbf{w}\in\mathbb{R}^d}{\operatorname{argmin}} \quad P(\mathbf{w}) = F(\mathbf{w}) + R(\mathbf{w}), \tag{4.1.1}$$

where $F(\mathbf{w}) = n^{-1}\sum_{i=1}^{n} f_i(\mathbf{w})$ is a sum of a finite number of strongly convex and smooth functions, and $R(\mathbf{w})$ is a block separable non-differential function. To explain block separability, let $\{\mathcal{G}_1, \ldots, \mathcal{G}_k\}$ be a partition of all the d coordinates where $\mathcal{G}_j$ is a block of coordinates. A subvector w_j is $[w_{k_1}, \ldots, w_{k_{|\mathcal{G}_j|}}]^\top$, where $\mathcal{G}_j = \{k_1, \ldots, k_{|\mathcal{G}_j|}\}$ and $1 \le j \le m$. The fact that $R(\mathbf{w})$ is block separable is equivalent to

$$R(\mathbf{w}) = \sum_{j=1}^{m} r_j(w_j). \tag{4.1.2}$$

The above problem is common in data mining and machine learning, such as the regularized empirical risk minimization, where $F(\mathbf{w})$ is the empirical loss function averaged over the training data sets, and $R(\mathbf{w})$ is a regularization term. For example, suppose that for a data mining problem there are n instances in a training data set $\{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_n, y_n)\}$. By choosing the squared loss $f_i(\mathbf{w}) = (\langle \mathbf{w}, \mathbf{x}_i \rangle - y_i)^2/2$ and $R(\mathbf{w}) = 0$, a least square regression is obtained. If $R(\mathbf{w})$ is chosen to be the sum of the absolute value of each coordinate in $\mathbf{w}$, it becomes a lasso regression [157]. In general, the problem in (4.1.1) can be approximately solved by proximal gradient descent algorithms [120] and proximal coordinate descent algorithms [95].

Coordinate descent algorithms have received increasing attention in the past decade in data mining and machine learning due to their successful applications in high dimensional problems with structural regularizers [50, 46, 109, 19, 158]. Randomized block coordinate descent (RBCD) [121, 132, 102, 137, 25, 65, 90] is a special block coordinate descent algorithm. At each iteration, it updates a block of coordinates in vector $\mathbf{w}$ based on evaluation of a random feature subset from the entire training data instances. The iteration complexity of RBCD was established and extended to composite minimization problems [121, 132, 102]. RBCD can choose a constant step size and converge at the same rate as gradient descent algorithms [121, 132, 102]. Compared with gradient descent, the per-iteration time complexity of RBCD is much lower. This is because RBCD computes a partial derivative restricted to only a single coordinate block at each iteration and updates just a single coordinate block of vector $\mathbf{w}$. However, it is still computationally expensive because at each iteration it requires evaluation of the gradient for all the n component functions f_i: the per-iteration computational complexity scales linearly with the training data set size n.

In view of this, stochastic block coordinate descent was proposed recently [35, 174, 163, 131]. Such algorithms compute the stochastic partial derivative restricted to one coordinate block with respect to one component function, rather than the full partial derivative with respect to all the component functions. Essentially, these algorithms employ sampling of both features and data instances at each iteration. However, they can only achieve a sublinear rate of convergence.

We propose an algorithm for stochastic block coordinate descent using optimal sampling, namely *accelerated stochastic block coordinate descent with optimal sampling* (ASBCD). On one hand, ASBCD employs a simple gradient update with optimal non-uniform sampling, which is in sharp contrast to the aforementioned stochastic block coordinate descent algorithms based on uniform sampling. On the other hand, we incorporate the incrementally averaged partial derivative into the stochastic partial derivative to achieve a linear rate of convergence rather than a sublinear rate.

To be specific, given error ϵ and number of coordinate blocks m, for strongly convex $f_i(\mathbf{w})$ with the convexity parameter μ and the Lipschitz continuous gradient constant L_i ($L_M = \max_i L_i$), the iteration

Algorithm 1 ASBCD: Accelerated Stochastic Block Coordinate Descent with Optimal Sampling

1: **Inputs:** step size η and sampling probability set $\mathcal{P} = \{p_1, \ldots, p_n\}$ of component functions $f_1, \ldots, f_n$
2: **Initialize:** $\boldsymbol{\phi}_i^{(0)} = \mathbf{w}^{(0)} \in \mathbb{R}^d$
3: **for** $t = 1, 2, \ldots$ **do**
4: Sample a component function index i from $\{1, \ldots, n\}$ at probability $p_i \in \mathcal{P}$ with replacement
5: $\boldsymbol{\phi}_c^{(t)} \leftarrow \mathbf{w}^{(t-1)}$
6: Sample a coordinate block index j from $\{1, \ldots, m\}$ uniformly at random with replacement
7: $w_j^{(t)} \leftarrow \text{prox}_{\eta,j}\big(w_j^{(t-1)} - \eta\big[(np_i)^{-1}\nabla_{\mathcal{G}_j} f_i(\boldsymbol{\phi}_c^{(t)}) - (np_i)^{-1}\nabla_{\mathcal{G}_j} f_i(\boldsymbol{\phi}_c^{(t-1)}) + n^{-1}\sum_{k=1}^n \nabla_{\mathcal{G}_j} f_k(\boldsymbol{\phi}_k^{(t-1)})\big]\big)$
8: $\mathbf{w}_{\setminus\mathcal{G}_j}^{(t)} \leftarrow \mathbf{w}_{\setminus\mathcal{G}_j}^{(t-1)}$
9: **end for**

complexity of ASBCD is

$$\mathcal{O}\bigg[m\bigg(\frac{1}{n}\sum_{i=1}^n \frac{L_i}{\mu} + n\bigg)\log\frac{1}{\epsilon}\bigg].$$

Notation. Here we define and describe the notation used through this chapter. Let w_k be the k^{th} element of a vector $\mathbf{w} = [w_1, \ldots, w_d]^\top \in \mathbb{R}^d$. We use $\|\mathbf{w}\| = \|\mathbf{w}\|_2 = \big(\sum_{k=1}^d w_k^2\big)^{1/2}$ to denote the ℓ_2 norm of a vector $\mathbf{w}$ and $\|\mathbf{w}\|_1 = \sum_{k=1}^d |w_k|$. The subvector of $\mathbf{w}$ excluding w_j is denoted by $\mathbf{w}_{\setminus\mathcal{G}_j}$. The simple proximal mapping for each coordinate block, also known as the proximal operator, is defined as

$$\text{prox}_{\eta,j}(\mathbf{w}) = \underset{\mathbf{u}\in\mathbb{R}^d}{\text{argmin}}\ \frac{1}{2\eta}\|\mathbf{w}-\mathbf{u}\|^2 + r_j(\mathbf{u}). \tag{4.1.3}$$

4.2 The Proposed Algorithm

We propose ASBCD (Algorithm 1), an accelerated algorithm for stochastic block coordinate descent with optimal sampling. It starts with known initial vectors $\boldsymbol{\phi}_i^{(0)} = \mathbf{w}^{(0)} \in \mathbb{R}^d$ for all i.

In sharp contrast to stochastic block coordinate descent with uniform sampling, ASBCD selects a component function according to non-uniform probabilities (Line 4 of Algorithm 1).

In Algorithm 1, we define the gradient of any function $f(\boldsymbol{\phi})$ with respect to a coordinate block $\mathcal{G}_j$ of $\boldsymbol{\phi}$ as $\nabla_{\mathcal{G}_j} f(\boldsymbol{\phi}) = [\nabla f(\boldsymbol{\phi})]_{\mathcal{G}_j} = [\partial f(\boldsymbol{\phi})/\partial\phi_{k_1}, \ldots, \partial f(\boldsymbol{\phi})/\partial\phi_{k_{|\mathcal{G}_j|}}]^\top$, where $\mathcal{G}_j = \{k_1, \ldots, k_{|\mathcal{G}_j|}\}$.

Algorithm 1 has a lower computational cost than either proximal gradient descent or RBCD at each iteration. The update at each iteration of Algorithm 1 is restricted to only a sampled component function (Line 4) and a sampled block of coordinates (Line 6).

The key updating step (Line 7) with respect to a stochastic block of coordinates incorporates the incrementally averaged partial derivative into the stochastic partial derivative with the third term $n^{-1}\sum_{k=1}^n \nabla_{\mathcal{G}_j} f_k(\boldsymbol{\phi}_k^{(t-1)})$ within the square bracket. At each iteration with i and j sampled, this summation term

$\sum_{k=1}^{n} \nabla_{\mathcal{G}_j} f_k(\boldsymbol{\phi}_k^{(t-1)})$ is efficiently updated by subtracting $\nabla_{\mathcal{G}_j} f_i(\boldsymbol{\phi}_i^{(t-2)})$ from itself while adding $\nabla_{\mathcal{G}_j} f_i(\boldsymbol{\phi}_i^{(t-1)})$ to itself.

Remark 4.2.1 *For many empirical risk minimization problems with each training data instance* $(\mathbf{x}_i, y_i)$ *and a loss function* ℓ*, the gradient of* $f_i(\mathbf{w})$ *with respect to* $\mathbf{w}$ *is a multiple of* $\mathbf{x}_i$*:* $\nabla f_i(\mathbf{w}) = \ell'(\langle \mathbf{w}, \mathbf{x}_i \rangle, y_i)\mathbf{x}_i$*. Therefore,* $\nabla f_i(\boldsymbol{\phi}_i)$ *can be compactly saved in memory by only saving scalars* $\ell'(\langle \boldsymbol{\phi}_i, \mathbf{x}_i \rangle, y_i)$ *with the same space cost as those of many other related algorithms MRBCD, SVRG, SAGA, SDCA, and SAG described in Section 4.5.*

Remark 4.2.2 *The sampling probability of component functions* f_i *in Line 4 of Algorithm 1 is according to a given probability set* $\mathcal{P} = \{p_1, \ldots, p_n\}$*. The uniform sampling scheme employed by stochastic block coordinate descent methods fits under this more generalized sampling framework as a special case, where* $p_i = 1/n$*. We reveal that the optimal non-uniform sampling can be employed to lower the iteration complexity in Section 4.3.*

When taking the expectation of the squared gap between the iterate $\mathbf{w}^{(t)}$ and the optimal solution $\mathbf{w}^*$ in (4.1.1) with respect to the stochastic coordinate block index, the obtained upper bound does not depend on such an index or the proximal operator. This property may lead to additional algorithmic development and here it is important for deriving a linear rate of convergence for Algorithm 1. We prove the rate of convergence bound in Section 4.7 after presenting and discussing the main theory in Section 4.3.

4.3 Main Theory

In this section, we present and discuss the main theory of our proposed algorithm (Algorithm 1). The proof of the main theory is presented in Section 4.7.

We begin with the following assumptions on $F(\mathbf{w})$ and $R(\mathbf{w})$ in the composite objective optimization problem as characterized in (4.1.1). These assumptions are mild and can be verified in many regularized empirical risk minimization problems in data mining and machine learning.

Assumption 4.3.1 (Lipschitz Continuous Gradient) *Each gradient* $\nabla f_i(\mathbf{w})$ *is Lipschitz continuous with the constant* L_i*,* i.e.*, for all* $\mathbf{w} \in \mathbb{R}^d$ *and* $\mathbf{u} \in \mathbb{R}^d$ *we have*

$$\|\nabla f_i(\mathbf{w}) - \nabla f_i(\mathbf{u})\| \leq L_i \|\mathbf{w} - \mathbf{u}\|.$$

Assumption 4.3.2 (Strong convexity) *Each function $f_i(\mathbf{w})$ is strongly convex,* i.e.*, there exists a positive constant μ such that for all $\mathbf{w} \in \mathbb{R}^d$ and $\mathbf{u} \in \mathbb{R}^d$ we have*

$$f_i(\mathbf{u}) - f_i(\mathbf{w}) - \langle \nabla f_i(\mathbf{w}), \mathbf{u} - \mathbf{w} \rangle \geq \frac{\mu}{2} \|\mathbf{u} - \mathbf{w}\|^2 .$$

Assumption 3.2 implies that $F(\mathbf{w})$ is also strongly convex, i.e.*, there exists a positive constant μ such that for all $\mathbf{w} \in \mathbb{R}^d$ and $\mathbf{u} \in \mathbb{R}^d$ we have*

$$F(\mathbf{u}) - F(\mathbf{w}) - \langle \nabla F(\mathbf{w}), \mathbf{u} - \mathbf{w} \rangle \geq \frac{\mu}{2} \|\mathbf{u} - \mathbf{w}\|^2 .$$

Assumption 4.3.3 (Block separability) *The regularization function $R(\mathbf{w})$ is convex but non-differentiable, and a closed-form solution can be obtained for the proximal operator defined in* (5.6.1)*. Importantly, $R(\mathbf{w})$ is block separable as defined in* (4.1.2)*.*

With the above assumptions being made, now we establish the linear rate of convergence for Algorithm 1, which is stated in the following theorem.

Theorem 4.3.4 *Let $L_M = \max_i L_i$ and $p_I = \min_i p_i$. Suppose that Assumptions 5.2.1—4.3.3 hold. Based on Algorithm 1 and with $\mathbf{w}^*$ defined in* (4.1.1)*, by setting $\eta = \max_i np_i/[2(n\mu + L_i)]$, $\zeta = np_I/(L_M\eta) - 1 > 0$, $\kappa = L_M^2 m/[2n\eta(L_M - \mu + L_M\eta\mu\zeta)] > 0$, and $0 < \alpha = 1 - \eta\mu/m < 1$, it holds that*

$$\mathbb{E}_{i,j}[\|\mathbf{w}^{(t)} - \mathbf{w}^*\|^2] \leq \alpha^t \big[\|\mathbf{w}^{(0)} - \mathbf{w}^*\|^2 + \frac{1}{\kappa}[F(\mathbf{w}^{(0)}) - F(\mathbf{w}^*) - \langle \nabla F(\mathbf{w}^*), \mathbf{w}^{(0)} - \mathbf{w}^* \rangle]\big].$$

Remark 4.3.5 *Theorem 4.3.4 justifies the linear rate of convergence for Algorithm 1. Parameter α depends on the number of coordinate blocks m. It may be tempting to set $m = 1$ for faster convergence. However, this is improper due to lack of considerations for the computational cost at each iteration. When $m = 1$, at each iteration the gradient is updated with respect to all coordinates. When $m > 1$, at each iteration of Algorithm 1 the gradient is updated with respect to only a sampled coordinate block among all coordinates, so the computational cost is lower than that of $m = 1$ per iteration. Therefore, comparing algorithms that update the gradient with respect to different numbers of coordinates per iteration should be based on the same number of entire data passes (the least possible iterations for passing through the entire data instances with respect to all coordinates). We perform experiments to compare such different algorithms in Section 4.4.*

Remark 4.3.6 *Theorem 4.3.4 implies a more generalized iteration complexity of Algorithm 1, which is*

$$\mathcal{O}\bigg[m\bigg(\min_i \frac{L_i/\mu + n}{np_i}\bigg)\log\frac{1}{\epsilon}\bigg] \tag{4.3.1}$$

given the error $\epsilon > 0$. *The uniform sampling scheme fits this more generalized result with* $p_i = 1/n$. *With* $L_M = \max_i L_i$, *by setting* $p_i = 1/n$, $\eta = 1/\left[2(L_M + n\mu)\right] > 0$, $\zeta = \left(L_M + 2n\mu\right)/L_M > 0$, $\kappa = m/\left[2n\eta(1-\eta\mu)\right] > 0$, *and* $0 < \alpha = 1 - \mu/[2m(L_M + n\mu)] < 1$, *Theorem 4.3.4 still holds. The iteration complexity of ASBCD with uniform sampling is*

$$\mathcal{O}\bigg[m\bigg(\frac{L_M}{\mu} + n\bigg)\log\frac{1}{\epsilon}\bigg]. \tag{4.3.2}$$

Now we show that the iteration complexity in (4.3.2) *can be further improved by optimal sampling. To begin with, minimizing* α *can be achieved by maximizing* η *with respect to* p_i. *It is easy to show that* η *is maximized when* $p_i = (n + L_i/\mu)/\sum_{k=1}^n (n + L_k/\mu)$. *Then, by setting* $\eta = n/\big[2\sum_{i=1}^n (n\mu + L_i)\big] > 0$ *we obtain the iteration complexity of ASBCD with optimal sampling:*

$$\mathcal{O}\bigg[m\bigg(\frac{1}{n}\sum_{i=1}^n \frac{L_i}{\mu} + n\bigg)\log\frac{1}{\epsilon}\bigg]. \tag{4.3.3}$$

Corollary 4.3.7 *Let* $L_M = \max_i L_i$. *Suppose that Assumptions 5.2.1—4.3.3 hold. Based on Algorithm 1 and with* $\mathbf{w}^*$ *defined in* (4.1.1), *by setting* $p_i = (n + L_i/\mu)/\sum_{k=1}^n (n + L_k/\mu)$, $\zeta = \sum_{i=1}^n L_i^{-1}/\sum_{i=1}^n (2n\mu + 2L_i)^{-1} - 1 > 0$, *and* $0 < \alpha = 1 - n\mu/[2m\sum_{i=1}^n (n\mu + L_i)] < 1$, *we chose* $\eta = n/[2\sum_{i=1}^n (n\mu + L_i)] > 0$ *and it holds that*

$$\mathbb{E}_{i,j}[\|\mathbf{w}^{(t)} - \mathbf{w}^*\|^2] \leq \alpha^t\Big[\|\mathbf{w}^{(0)} - \mathbf{w}^*\|^2 + \frac{n}{m(L_M + n\mu)}[F(\mathbf{w}^{(0)}) - F(\mathbf{w}^*) - \langle\nabla F(\mathbf{w}^*), \mathbf{w}^{(0)} - \mathbf{w}^*\rangle]\Big].$$

Comparing the iteration complexity of ASBCD in (4.3.3) and (4.3.2), it is clear that the optimal sampling scheme results in a lower iteration complexity than uniform sampling.

4.4 Evaluation

We conduct experiments to evaluate the performance of our proposed ASBCD algorithm in comparison with different algorithms on large-scale real data sets.

4.4.1 Problems and Measures

We define the problems and measures used in the empirical evaluation. Classification and regression are two corner-stone data mining and machine learning problems. We evaluate the performance of the proposed ASBCD algorithm in solving these two problems.

Classification and Regression Problems

As a case study, the classification problem is $\ell_{1,2}$-regularized logistic regression:

$$\begin{aligned}\mathbf{w}^* &= \operatorname*{argmin}_{\mathbf{w}\in\mathbb{R}^d} P(\mathbf{w}) \\ &= \operatorname*{argmin}_{\mathbf{w}\in\mathbb{R}^d} \frac{1}{n}\sum_{i=1}^{n} \log\left[1+\exp(-y_i\langle \mathbf{w}, \mathbf{x}_i\rangle)\right] + \frac{\lambda_2}{2}\|\mathbf{w}\|^2 + \lambda_1\|\mathbf{w}\|_1 .\end{aligned}$$

For the the regression problem in this empirical study, the elastic net is used:

$$\begin{aligned}\mathbf{w}^* &= \operatorname*{argmin}_{\mathbf{w}\in\mathbb{R}^d} P(\mathbf{w}) \\ &= \operatorname*{argmin}_{\mathbf{w}\in\mathbb{R}^d} \frac{1}{n}\sum_{i=1}^{n} \frac{(\langle \mathbf{w}, \mathbf{x}_i\rangle - y_i)^2}{2} + \frac{\lambda_2}{2}\|\mathbf{w}\|^2 + \lambda_1\|\mathbf{w}\|_1 .\end{aligned}$$

The regularization parameters λ_1 and λ_2 in both problems are tuned by proximal gradient descent using five-fold cross-validation on the training data sets.

Measures for Convergence

Recall the problem of composite function minimization as formalized in (4.1.1). In evaluation of the algorithm performance on the convergence effect, we use the measure of objective gap value: $P(\mathbf{w}) - P(\mathbf{w}^*)$.

4.4.2 Large-Scale Real Data Sets

The empirical studies are conducted on the following three real data sets that are downloaded using the LIBSVM software [24]:

- **COVTYPE**: Data set for predicting forest cover type from cartographic variables [93].
- **RCV1**: Reuters Corpus Volume I data set for text categorization research [86].
- **E2006-TFIDF**: Data set for predicting risk from financial reports from thousands of publicly traded U.S. companies [76].

Table 4.1: Summary statistics of three large-scale real data sets in the experiments. These data sets are used for evaluating performance of algorithms in solving two corner-stone data mining and machine learning problems: classification and regression.

Data Set	#Training Instances	#Testing Instances	#Features	Problem
COVTYPE	290,506	290,506	54	Classification
RCV1	20,242	677,399	47,236	Classification
E2006-TFIDF	16,087	3,308	150,360	Regression

Each of these real data sets has a large size in either its instance count or feature size, or both. Summary statistics of these data sets are provided in Table 4.1.

4.4.3 Algorithms for Comparison

We evaluate the performance of ASBCD in comparison with recently proposed competitive algorithms. To comprehensively evaluate ASBCD, we also compare variants of ASBCD with different sampling schemes.

Below are the seven algorithms for comparison.

- **SGD (SG)**: Proximal stochastic gradient descent. This algorithm has a sublinear rate of convergence. To ensure the high competitiveness of this algorithm, the implementation is based on a recent work [16].
- **SBCD (SB)**: Stochastic block coordinate descent. It is the same as SGD except that SBCD updates the gradient with respect to a randomly sampled block of coordinates at each iteration. SBCD also converges at a sublinear rate.
- **SAGA (SA)**: Advanced stochastic gradient method [37]. This algorithm is based on uniform sampling of component functions. It updates the gradient with respect to all coordinates at each iteration. SAGA has a linear rate of convergence.
- **SVRG (SV)**: (Proximal) stochastic variance reduced gradient [70, 173]. This algorithm is based on uniform sampling of component functions. It updates the gradient with respect to all coordinates at each iteration. Likewise, SVRG converges to the optimum at a linear rate.
- **MRBCD (MR)**: Mini-batch randomized block coordinate descent [191]. This algorithm uses uniform sampling of component functions. MRBCD converges linearly to the optimum.
- **ASBCD-U (U)**: The proposed ASBCD algorithm with uniform sampling of component functions. The sampling probability p_i for component function f_i is $p_i = 1/n$. The sampling probability p_i for component function f_i: $p_i = L_i / \sum_{k=1}^{n} L_k$.
- **ASBCD-O (O)**: The proposed ASBCD algorithm with optimal sampling as described in Corollary 4.3.7. The sampling probability p_i for component function f_i is $p_i = (n + L_i/\mu) / \sum_{k=1}^{n} (n + L_k/\mu)$.

4.4.4 Experimental Setting

Note that algorithms SBCD, MRBCD, and ASBCD update the gradient with respect to a sampled block of coordinates at each iteration. In contrast, SGD, SAGA, and SVRG update the gradient with respect to all the coordinates per iteration. Recalling Remark 4.3.5, comparison of these algorithms is based on the same entire data passes.

Equipment Configuration

We evaluate convergence and testing accuracy with respect to training time. The experiments are conducted on a computer with an 8-core 3.4GHz CPU and a 32GB RAM.

Parameter Setting

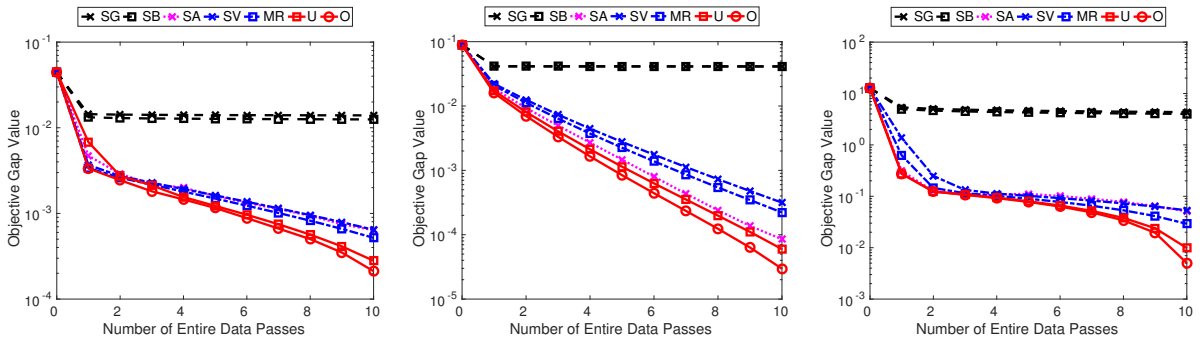
Different from the other algorithms in comparison, the SVRG and MRBCD algorithms both have multiple stages with two nested loops. The inner-loop counts in SVRG and MRBCD are set to the training data instance counts as suggested in a few recent studies [70, 173, 191].

For each algorithm, its parameters, such as the step size (η in this chapter), are chosen around the theoretical values to give the fastest convergence under the five-fold cross validation. Here we describe the details. The training data set is divided into five subsets of approximately the same size. One validation takes five trials on different subsets: in each trial, one subset is left out and the remaining four subsets are used. The convergence effect in one cross-validation is estimated by the averaged performance of the five trials.

4.4.5 Experimental Results

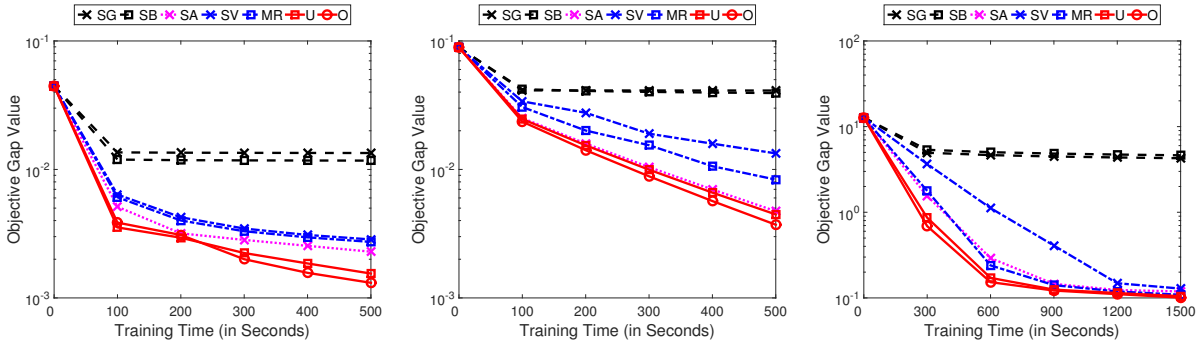
All the experimental results are obtained from 10 replications. For clarity of exposition, Figures 4.1 and 4.2 plot the mean values of the results from all these replications.

We compare the algorithms on three data sets COVTYPE, RCV1, and E2006-TFIDF as described in Section 4.4.2 and summarized in Table 4.1. COVTYPE and RCV1 are used for the classification problem, while E2006-TFIDF is for the regression problem. Figures 4.1 and 4.2 compare convergence of algorithms for the same entire data passes and for the same training time. In general, ASBCD with optimal sampling (O) converges fastest to the optimum for both the same number of entire data passes and the same training time.



(a) Classification on *COVTYPE* (b) Classification on *RCV1* (c) Regression on *E2006-TFIDF*

Figure 4.1: Convergence comparison of algorithms for the same number of entire data passes for classification and regression on three data sets. In general, ASBCD with optimal sampling (O) converges fastest to the optimum for the same number of entire data passes.



(a) Classification on *COVTYPE* (b) Classification on *RCV1* (c) Regression on *E2006-TFIDF*

Figure 4.2: Convergence comparison of algorithms for the same training time for classification and regression on three data sets. In general, ASBCD with optimal sampling (O) converges fastest to the optimum for the same training time.

4.5 Related Work

The first line of research in modern optimization is randomized block coordinate descent (RBCD) algorithms [46, 169, 98, 137, 132]. These algorithms exploit the block separability of regularization function $R(\mathbf{w})$. With separable coordinate blocks, such algorithms only compute the gradient of $F(\mathbf{w})$ with respect to a randomly selected block at each iteration rather than the full gradient with respect to all coordinates: they are faster than the full gradient descent at each iteration [46, 169, 98, 137, 132]. However, such algorithms still compute the exact partial gradient based on all the n component functions per iteration, though accessing the entire component functions is computationally more expensive when the training data set has a larger number of instances [183].

Recently, an MRBCD algorithm was proposed for randomized block coordinate descent using mini-batches [191]. At each iteration, both a block of coordinates and a mini-batch of component functions are sampled but there are multiple stages with two nested loops. For each iteration of the outer loop, the exact gradient is computed once; while in the follow-up inner loop, gradient estimation is computed multiple times to help adjust the exact gradient. MRBCD has a linear rate of convergence for strongly convex and smooth $F(\mathbf{w})$ only when the batch size is "large enough" although batches of larger sizes increase the per-iteration computational cost [191] (Theorem 4.2). Similar algorithms and theoretical results to those of MRBCD were also proposed [163, 79]. Chen and Gu further considered related but different sparsity constrained non-convex problems and studied stochastic optimization algorithms with block coordinate gradient descent [28].

Our work departs from the related work in the above line of research by attaining a linear convergence using optimally and non-uniformly sampling of a single data instance at each of iterations.

The second line of research in modern optimization is proximal gradient descent. In each iteration, a proximal operator is used in the update, which can be viewed as a special case of splitting algorithms [96, 27, 131]. Proximal gradient descent is computationally expensive at each iteration, hence proximal stochastic gradient descent is often used when the data set is large. At each iteration, only one of the n component functions f_i is sampled, or a subset of f_i are sampled, which is also known as mini-batch proximal stochastic gradient [147]. Advantages for proximal stochastic gradient descent are obvious: at each iteration much less computation of the gradient is needed in comparison with proximal gradient descent. However, due to the variance in estimating the gradient by stochastic sampling, proximal stochastic gradient descent has a sublinear rate of convergence even when $P(\mathbf{w})$ is strongly convex and smooth.

To accelerate proximal stochastic gradient descent, variance reduction methods were proposed recently. Such accelerated algorithms include stochastic average gradient (SAG) [136], stochastic dual coordinate ascent (SDCA) [140], stochastic variance reduced gradient (SVRG) [70], semi-stochastic gradient descent (S2GD) [80], permutable incremental gradient (Finito) [38], minimization by incremental surrogate optimization (MISO) [106], and advanced stochastic gradient method (SAGA) [37]. There are also some more recent extensions in this line of research, such as proximal SDCA (ProxSDCA) [138], accelerated mini-batch SDCA (ASDCA) [139], adaptive variant of SDCA (AdaSDCA) [32], randomized dual coordinate ascent (Quartz) [130], mini-batch S2GD (mS2GD) [78], and proximal SVRG (ProxSVRG) [173].

Besides, several studies show that non-uniform sampling can be used to improve the rate of convergence of stochastic optimization algorithms [148, 121, 118, 173, 130, 190, 135, 129]. However, the proposed sampling schemes in these studies cannot be directly applied to our algorithm, because they are limited in at least

one of the following two aspects: (1) the algorithm does not apply to composite objectives with a non-differentiable function; (2) it does not support randomized block coordinate descent.

4.6 Conclusion

Research on big data is increasingly important and common. Training data mining and machine learning models often involve minimizing empirical risk or maximizing likelihood over the training data set, especially in solving classification and regression problems. Thus, big data research may rely on optimization algorithms, such as proximal gradient descent algorithms. At each iteration, proximal gradient descent algorithms have a much higher computational cost due to updating gradients based on all the data instances and features. Randomized block coordinate descent algorithms are still computationally expensive at each iteration when the data instance size is large. Therefore, we focused on stochastic block coordinate descent that samples both data instances and features at every iteration.

We proposed the ASBCD algorithm to accelerate stochastic block coordinate descent. ASBCD incorporates the incrementally averaged partial derivative into the stochastic partial derivative. For smooth and strongly convex functions with non-differentiable regularization functions, ASBCD is able to achieve a linear rate of convergence. The optimal sampling achieves a lower iteration complexity for ASBCD. The empirical evaluation with both classification and regression problems on three large-scale real data sets supported our theory.

4.7 Proof of the Main Theory

We provide the proof for the main theory delivered in Section 4.3. Note that all the expectations are taken conditional on $\mathbf{w}^{(t-1)}$ and each $\boldsymbol{\phi}_c^{(t-1)}$ unless otherwise stated. For brevity, we define

$$\mathbf{g}_i = \frac{1}{np_i}\nabla f_i(\boldsymbol{\phi}_c^{(t)}) - \frac{1}{np_i}\nabla f_i(\boldsymbol{\phi}_c^{(t-1)}) + \frac{1}{n}\sum_{k=1}^{n}\nabla f_k(\boldsymbol{\phi}_k^{(t-1)}). \tag{4.7.1}$$

Let us introduce several important lemmas. The proof of lemmas are presented in Section 4.8. To begin with, since Algorithm 1 leverages randomized coordinate blocks, the following lemma is needed for taking the expectation of the squared gap between the iterate $\mathbf{w}^{(t)}$ and the optimal solution $\mathbf{w}^*$ in (4.1.1) with respect to the coordinate block index j.

Lemma 4.7.1 *Suppose that Assumption 4.3.3 holds. Let j be a coordinate block index. With $\mathbf{g}_i$ defined in* (4.7.1) *and $\mathbf{w}^*$ defined in* (4.1.1)*, based on Algorithm 1 we have*

$$\mathbb{E}_j[\|\mathbf{w}^{(t)} - \mathbf{w}^*\|^2] \leq \frac{1}{m}\big[(m-1)\|\mathbf{w}^{(t-1)} - \mathbf{w}^*\|^2 + \|\mathbf{w}^{(t-1)} - \eta\mathbf{g}_i - \mathbf{w}^* + \eta\nabla F(\mathbf{w}^*)\|^2\big].$$

Lemma 4.7.1 takes the expectation of the squared gap between the iterate $\mathbf{w}^{(t)}$ and the optimal solution $\mathbf{w}^*$ in (4.1.1) with respect to the randomized coordinate block index. The obtained upper bound does not have a randomized coordinate block index or the proximal operator. Block separability and non-expansiveness of the proximal operator are both exploited in deriving the upper bound. This upper bound is used for deriving a linear rate of convergence for Algorithm 1.

Lemma 4.7.2 *Based on Algorithm 1 and as defined in* (4.7.1)*, we have $\mathbb{E}_i[\mathbf{g}_i] = \nabla F(\mathbf{w}^{(t-1)})$.*

Lemma 4.7.2 guarantees that $\mathbf{g}_i$ is an unbiased gradient estimator of $F(\mathbf{w})$. The proof is strictly based on the definition of $\mathbf{g}_i$ in (4.7.1).

Lemma 4.7.3 *With $\mathbf{g}_i$ defined in* (4.7.1) *and $\mathbf{w}^*$ defined in* (4.1.1)*, based on Algorithm 1 and for all $\zeta > 0$ we have*

$$\begin{aligned}\mathbb{E}_i\big[\,\|\mathbf{g}_i - \nabla F(\mathbf{w}^*)\|^2\,\big] \leq (1+\zeta)\mathbb{E}_i\left[\left\|\frac{1}{np_i}\nabla f_i(\mathbf{w}^{(t-1)}) - \frac{1}{np_i}\nabla f_i(\mathbf{w}^*)\right\|^2\right] - \zeta\|\nabla F(\mathbf{w}^{(t-1)}) - \nabla F(\mathbf{w}^*)\|^2 \\ + (1+\zeta^{-1})\mathbb{E}_i\left[\left\|\frac{1}{np_i}\nabla f_i(\boldsymbol{\phi}_c^{(t-1)}) - \frac{1}{np_i}\nabla f_i(\mathbf{w}^*)\right\|^2\right].\end{aligned}$$

Lemma 4.7.3 makes use of the property that $\mathbb{E}[\|\mathbf{x}\|^2] = \mathbb{E}[\|\mathbf{x}-\mathbb{E}[\mathbf{x}]\|^2] + \|\mathbb{E}[\mathbf{x}]\|^2$ for all $\mathbf{x}$ and the property that $\|\mathbf{x}+\mathbf{y}\|^2 \leq (1+\zeta)\|\mathbf{x}\|^2 + (1+\zeta^{-1})\|\mathbf{y}\|^2$ for all $\mathbf{x}, \mathbf{y}$, and $\zeta > 0$.

Lemma 4.7.4 *Let f be strongly convex with the convexity parameter μ and its gradient be Lipschitz continuous with the constant L. For all $\mathbf{x}$ and $\mathbf{y}$, it holds that*

$$\begin{aligned}\langle\nabla f(\mathbf{y}), \mathbf{x}-\mathbf{y}\rangle \leq f(\mathbf{x}) - f(\mathbf{y}) - \frac{1}{2(L-\mu)}\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|^2 \\ - \frac{\mu}{L-\mu}\langle\nabla f(\mathbf{x}) - \nabla f(\mathbf{y}), \mathbf{y}-\mathbf{x}\rangle - \frac{L\mu}{2(L-\mu)}\|\mathbf{y}-\mathbf{x}\|^2.\end{aligned}$$

Lemma 4.7.4 leverages properties of strongly convex functions with Lipschitz continuous gradient.

Lemma 4.7.5 *Algorithm 1 implies that*

$$\mathbb{E}_i\left[\frac{1}{n}\sum_{i=1}^n \frac{L_i}{np_i} f_i(\boldsymbol{\phi}_c^{(t)})\right] = \frac{1}{n}\sum_{i=1}^n \frac{L_i}{n} f_i(\mathbf{w}^{(t-1)}) + \frac{1}{n}\sum_{i=1}^n \frac{(1-p_i)L_i}{np_i} f_i(\boldsymbol{\phi}_c^{(t-1)}).$$

Lemma 4.7.5 is obtained according to the non-uniform sampling of component functions in Algorithm 1.

Remark 4.7.6 *Similar to Lemma 4.7.5, we have*

$$\begin{aligned}
&\mathbb{E}_i\left[\frac{1}{n}\sum_{i=1}^{n}\left\langle\frac{L_i}{np_i}\nabla f_i(\mathbf{w}^*),\boldsymbol{\phi}_c^{(t)}-\mathbf{w}^*\right\rangle\right]\\
&=\frac{1}{n}\sum_{i=1}^{n}\left\langle\frac{L_i}{n}\nabla f_i(\mathbf{w}^*),\mathbf{w}^{(t-1)}-\mathbf{w}^*\right\rangle+\frac{1}{n}\sum_{i=1}^{n}\left\langle\frac{(1-p_i)L_i}{np_i}\nabla f_i(\mathbf{w}^*),\boldsymbol{\phi}_c^{(t-1)}-\mathbf{w}^*\right\rangle. \qquad (4.7.2)
\end{aligned}$$

Now we develop the main theorem of bounding the rate of convergence for Algorithm 1.

4.7.1 Proof of Theorem 4.3.4

Proof. By applying Lemma 4.7.1, 4.7.2, and Lemma 4.7.3,

$$\begin{aligned}
&\mathbb{E}_{i,j}\left[\|\mathbf{w}^{(t)}-\mathbf{w}^*\|^2\right]\\
&\leq\frac{1}{m}\Bigg[m\|\mathbf{w}^{(t-1)}-\mathbf{w}^*\|^2+2\eta\langle\nabla F(\mathbf{w}^*),\mathbf{w}^{(t-1)}-\mathbf{w}^*\rangle-2\eta\langle\nabla F(\mathbf{w}^{(t-1)}),\mathbf{w}^{(t-1)}-\mathbf{w}^*\rangle\\
&\quad+\eta^2(1+\zeta)\mathbb{E}_i\left[\left\|\frac{1}{np_i}\nabla f_i(\mathbf{w}^{(t-1)})-\frac{1}{np_i}\nabla f_i(\mathbf{w}^*)\right\|^2\right]\\
&\quad+\eta^2(1+\zeta^{-1})\mathbb{E}_i\left[\left\|\frac{1}{np_i}\nabla f_i(\boldsymbol{\phi}_c^{(t-1)})-\frac{1}{np_i}\nabla f_i(\mathbf{w}^*)\right\|^2\right]-\eta^2\zeta\|\nabla F(\mathbf{w}^{(t-1)})-\nabla F(\mathbf{w}^*)\|^2\Bigg]. \qquad (4.7.3)
\end{aligned}$$

Substituting $\mathbf{x},\mathbf{y}$, and f with $\mathbf{w}^*,\mathbf{w}^{(t-1)}$, and f_i in Lemma 4.7.4, and taking average on both sides of the inequality in Lemma 4.7.4, we obtain

$$\begin{aligned}
&-2\eta\langle\nabla F(\mathbf{w}^{(t-1)}),\mathbf{w}^{(t-1)}-\mathbf{w}^*\rangle\\
&\leq\frac{2\eta}{n}\sum_{i=1}^{n}\frac{L_i-\mu}{L_i}[f_i(\mathbf{w}^*)-f_i(\mathbf{w}^{(t-1)})]-\frac{\eta}{n}\sum_{i=1}^{n}\frac{1}{L_i}\|\nabla f_i(\mathbf{w}^*)-\nabla f_i(\mathbf{w}^{(t-1)})\|^2\\
&\quad-\frac{2\eta\mu}{n}\sum_{i=1}^{n}\frac{1}{L_i}\langle\nabla f_i(\mathbf{w}^*),\mathbf{w}^{(t-1)}-\mathbf{w}^*\rangle-\eta\mu\|\mathbf{w}^*-\mathbf{w}^{(t-1)}\|^2. \qquad (4.7.4)
\end{aligned}$$

Recall the property of any function f that is convex and has a Lipschitz continuous gradient with the constant L: $f(\mathbf{y})\geq f(\mathbf{x})+\langle\nabla f(\mathbf{x}),\mathbf{y}-\mathbf{x}\rangle+\|\nabla f(\mathbf{x})-\nabla f(\mathbf{y})\|^2/(2L)$ for all $\mathbf{x}$ and $\mathbf{y}$ [119] (Theorem 2.1.5). Taking average on both sides, we have

$$\mathbb{E}_i\left[\left\|\frac{1}{np_i}\nabla f_i(\boldsymbol{\phi}_c^{(t-1)})-\frac{1}{np_i}\nabla f_i(\mathbf{w}^*)\right\|^2\right]\leq\frac{2}{n}\sum_{i=1}^{n}\frac{L_i}{np_i}\left[f_i(\boldsymbol{\phi}_c^{(t-1)})-f_i(\mathbf{w}^*)-\langle\nabla f_i(\mathbf{w}^*),\boldsymbol{\phi}_c^{(t-1)}-\mathbf{w}^*\rangle\right] \qquad (4.7.5)$$

after substituting $\mathbf{y}, \mathbf{x}$, and f with $\boldsymbol{\phi}_c^{(t-1)}, \mathbf{w}^*$, and f_i while re-arranging terms.

Before further proceeding with the proof, we define

$$H^{(t)} = \frac{1}{n}\sum_{i=1}^{n} \frac{L_i}{np_i}\big[f_i(\boldsymbol{\phi}_c^{(t)}) - f_i(\mathbf{w}^*) - \langle \nabla f_i(\mathbf{w}^*), \boldsymbol{\phi}_c^{(t)} - \mathbf{w}^* \rangle\big] + \kappa \|\mathbf{w}^{(t)} - \mathbf{w}^*\|^2. \tag{4.7.6}$$

Following the definition in (4.7.6), for all $\alpha > 0$,

$$\begin{aligned}
&\mathbb{E}_{i,j}[H^{(t)}] - \alpha H^{(t-1)} \\
&= \mathbb{E}_{i,j}\left[\frac{1}{n}\sum_{i=1}^{n} \frac{L_i}{np_i} f_i(\boldsymbol{\phi}_c^{(t)})\right] - \frac{1}{n}\sum_{i=1}^{n} \frac{L_i}{np_i} f_i(\mathbf{w}^*) - \mathbb{E}_{i,j}\left[\frac{1}{n}\sum_{i=1}^{n} \frac{L_i}{np_i} \cdot \right. \\
&\qquad \left. \langle \nabla f_i(\mathbf{w}^*), \boldsymbol{\phi}_c^{(t)} - \mathbf{w}^* \rangle\right] + \mathbb{E}_{i,j}\big[\kappa \|\mathbf{w}^{(t)} - \mathbf{w}^*\|^2\big] - \alpha H^{(t-1)}.
\end{aligned}$$

Recall the property of any strongly convex function f with the convexity parameter μ that $f(\mathbf{y}) \leq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|^2/(2\mu)$ for all $\mathbf{x}$ and $\mathbf{y}$ [119] (Theorem 2.1.10). We can obtain $-\|\nabla f_i(\mathbf{w}^{(t-1)}) - \nabla f_i(\mathbf{w}^*)\|^2 \leq -2\mu\big[f_i(\mathbf{w}^{(t-1)}) - f_i(\mathbf{w}^*) - \langle \nabla f_i(\mathbf{w}^*), \mathbf{w}^{(t-1)} - \mathbf{w}^* \rangle\big]$.

Combining (4.7.3) with a positive constant κ, (4.7.4), and (4.7.5), after simplifying terms, by Lemma 4.7.5 and (4.7.2), with defining $L_M = \max_i L_i$ and $p_I = \min_i p_i$ we have

$$\mathbb{E}_{i,j}[H^{(t)}] - \alpha H^{(t-1)} \leq \sum_{k=1}^{4} c_k T_k, \tag{4.7.7}$$

where the four constant factors are

$$\begin{aligned}
c_1 &= \frac{\kappa\eta}{mn}\left(\frac{\eta(1+\zeta)}{np_I} - \frac{1}{L_M}\right), \\
c_2 &= \frac{1}{n}\left(\frac{L_M}{n} - \frac{2\kappa\eta(L_M - \mu)}{L_M m} - \frac{2\beta\kappa\eta^2\mu}{m}\right), \\
c_3 &= \kappa\left(1 - \frac{\eta\mu}{m} - \alpha\right), \\
c_4 &= \frac{L_M}{n^2}\left(\frac{2\kappa\eta^2(1+\zeta^{-1})}{mp_I} + \frac{1-\alpha}{p_I} - 1\right),
\end{aligned}$$

and the four corresponding terms are

$$
\begin{aligned}
T_1 &= \sum_{i=1}^{n} \|\nabla f_i(\mathbf{w}^{(t-1)}) - \nabla f_i(\mathbf{w}^*)\|^2, \\
T_2 &= \sum_{i=1}^{n} \big[f_i(\mathbf{w}^{(t-1)}) - f_i(\mathbf{w}^*) - \langle \nabla f_i(\mathbf{w}^*), \mathbf{w}^{(t-1)} - \mathbf{w}^* \rangle\big], \\
T_3 &= \|\mathbf{w}^{(t-1)} - \mathbf{w}^*\|^2, \\
T_4 &= \sum_{i=1}^{n} \big[f_i(\boldsymbol{\phi}_c^{(t-1)}) - f_i(\mathbf{w}^*) - \langle \nabla f_i(\mathbf{w}^*), \boldsymbol{\phi}_c^{(t-1)} - \mathbf{w}^* \rangle\big].
\end{aligned}
$$

There are four constant factors associated with four terms on the right-hand side of (4.7.7). Among the four terms, obviously $T_1 \geq 0$ and $T_3 \geq 0$. By the convexity property of f_i, we have $T_2 \geq 0$ and $T_4 \geq 0$. We choose $\eta = \max_i np_i/[2(n\mu + L_i)]$. By setting $c_1 = 0$ with $\zeta = np_I/(L_M\eta) - 1 > 0$, $c_2 = 0$ with $\kappa = L_M^2 m/\big[2n\eta(L_M - \mu + L_M\eta\mu\zeta)\big] > 0$, and $c_3 = 0$ with $0 < \alpha = 1 - \eta\mu/m < 1$, it can be verified that $c_4 \leq 0$.

With the aforementioned constant factor setting, $\mathbb{E}_{i,j}[H^{(t)}] - \alpha H^{(t-1)} \leq 0$, where the expectation is conditional on information from the previous iteration $t-1$. Taking expectation with this previous iteration gives $\mathbb{E}_{i,j}[H^{(t)}] \leq \alpha \mathbb{E}_{i,j}[H^{(t-1)}]$. By chaining over t iteratively, $\mathbb{E}_{i,j}[H^{(t)}] \leq \alpha^t H^{(0)}$. Since the sum of the first three terms in (4.7.6) is non-negative by the convexity of F, we have $\kappa\|\mathbf{w}^{(t)} - \mathbf{w}^*\|^2 \leq H^{(t)}$. Together with the aforementioned results by chaining over t, the proof is complete. ∎

4.8 Proof of Lemmas

This section provides proof of all the lemmas in Chapter 4.

4.8.1 Proof of Lemma 4.7.1

Proof.

Recall Assumption 4.3.3 that $R(\mathbf{w})$ is block separable. We first define

$$
\text{prox}_{\eta}(\mathbf{w}) = \big[\text{prox}_{\eta,1}(\mathbf{w}_{\mathcal{G}_1})^\top, \ldots, \text{prox}_{\eta,j}(w_j)^\top\big]^\top, \tag{4.8.1}
$$

$$
\mathbf{g}_{i,\mathcal{G}_j} = \nabla_{\mathcal{G}_j} f_i(\boldsymbol{\phi}_c^{(t)}) - \nabla_{\mathcal{G}_j} f_i(\boldsymbol{\phi}_c^{(t-1)}) + \frac{1}{n}\sum_{k=1}^{n} \nabla_{\mathcal{G}_j} f_k(\boldsymbol{\phi}_k^{(t-1)}), \tag{4.8.2}
$$

$$\boldsymbol{\delta}^{\mathcal{G}_j} = \Big[0, \ldots, 0, \mathrm{prox}_{\eta,j}(w_j^{(t-1)} - \eta \mathbf{g}_{i,\mathcal{G}_j})^\top - \mathrm{prox}_{\eta,j}\big(\mathbf{w}^*_{\mathcal{G}_j} - \eta \nabla_{\mathcal{G}_j} F(\mathbf{w}^*)\big)^\top, 0, \ldots, 0\Big]^\top, \tag{4.8.3}$$

$$\text{and} \quad \boldsymbol{\delta} = \mathrm{prox}_\eta(\mathbf{w}^{(t-1)} - \eta \mathbf{g}_i) - \mathrm{prox}_\eta\big(\mathbf{w}^* - \eta \nabla F(\mathbf{w}^*)\big). \tag{4.8.4}$$

Since $R(\mathbf{w})$ is block separable, $\boldsymbol{\delta}^{\mathcal{G}_j}$ and $\boldsymbol{\delta}^{\mathcal{G}_{j'}}$ are orthogonal to each other for all $j \neq j'$, and by (4.8.3) and (4.8.4) we have

$$\mathbb{E}_j\big[\|\boldsymbol{\delta}^{\mathcal{G}_j}\|^2\big] = \frac{1}{m}\sum_{j=1}^m \|\boldsymbol{\delta}^{\mathcal{G}_j}\|^2 = \frac{\|\boldsymbol{\delta}\|^2}{m}. \tag{4.8.5}$$

Similarly, for convenience of technical discussions we further define

$$\boldsymbol{\psi}^{\mathcal{G}_j} = \big[0, \ldots, 0, (w_j^{(t-1)} - \mathbf{w}^*_{\mathcal{G}_j})^\top, 0, \ldots, 0\big]^\top \tag{4.8.6}$$

$$\text{and} \quad \boldsymbol{\psi} = \mathbf{w}^{(t-1)} - \mathbf{w}^*, \tag{4.8.7}$$

then we are able to obtain their relation:

$$\mathbb{E}_j\big[\|\boldsymbol{\psi}^{\mathcal{G}_j}\|^2\big] = \frac{1}{m}\sum_{j=1}^m \|\boldsymbol{\psi}^{\mathcal{G}_j}\|^2 = \frac{\|\boldsymbol{\psi}\|^2}{m}. \tag{4.8.8}$$

From the definition in (4.8.2), by exploiting the block separability of $R(\mathbf{w})$, we have

$$\begin{aligned}
&\mathbb{E}_j\big[\|\mathbf{w}^{(t)} - \mathbf{w}^*\|^2\big] \\
&\quad = \sum_{k\neq j} \mathbb{E}_k\big[\|\mathbf{w}^{(t-1)}_{\mathcal{G}_k} - \mathbf{w}^*_{\mathcal{G}_k}\|^2\big] + \mathbb{E}_j\left[\Big\|\mathrm{prox}_{\eta,j}(w_j^{(t-1)} - \eta \mathbf{g}_{i,\mathcal{G}_j}) - \mathrm{prox}_{\eta,j}\big(\mathbf{w}^*_{\mathcal{G}_j} - \eta \nabla_{\mathcal{G}_j} F(\mathbf{w}^*)\big)\Big\|^2\right].
\end{aligned}$$

After substitution with (4.8.3), (4.8.4), (4.8.6), and (4.8.7), according to (4.8.5) and (4.8.8), since

$$\sum_{k\neq j} \mathbb{E}_k\big[\|\boldsymbol{\psi}^{\mathcal{G}_k}\|^2\big] + \mathbb{E}_j\big[\|\boldsymbol{\delta}^{\mathcal{G}_j}\|^2\big] = \frac{(m-1)\|\boldsymbol{\psi}\|^2}{m} + \frac{\|\boldsymbol{\delta}\|^2}{m},$$

by the non-expansiveness of the proximal operator (4.8.1) [120] and that $\mathbf{w}^*$ is the optimal value in (4.1.1),

$$\begin{aligned}
&\mathbb{E}_j\big[\|\mathbf{w}^{(t)} - \mathbf{w}^*\|^2\big] \\
&= \frac{(m-1)}{m}\|\mathbf{w}^{(t-1)} - \mathbf{w}^*\|^2 + \frac{1}{m}\Big\|\mathrm{prox}_\eta(\mathbf{w}^{(t-1)} - \eta\mathbf{g}_i) - \mathrm{prox}_\eta\big(\mathbf{w}^* - \eta\nabla F(\mathbf{w}^*)\big)\Big\|^2 \\
&\leq \frac{1}{m}\big[(m-1)\|\mathbf{w}^{(t-1)} - \mathbf{w}^*\|^2 + \|\mathbf{w}^{(t-1)} - \eta\mathbf{g}_i - \mathbf{w}^* + \eta\nabla F(\mathbf{w}^*)\|^2\big]. \qquad (4.8.9)
\end{aligned}$$

■

4.8.2 Proof of Lemma 4.7.2

Proof. The proof is straightforward using the definition of $\mathbf{g}_i$ in (4.7.1).

$$\begin{aligned}
\mathbb{E}_i[\mathbf{g}_i] &= \mathbb{E}_i\Big[\frac{1}{np_i}\nabla f_i(\boldsymbol{\phi}_c^{(t)}) - \frac{1}{np_i}\nabla f_i(\boldsymbol{\phi}_c^{(t-1)})\Big] + \frac{1}{n}\sum_{k=1}^n \nabla f_k(\boldsymbol{\phi}_k^{(t-1)}) \\
&= \sum_{i=1}^n \frac{p_i}{np_i}\nabla f_i(\mathbf{w}^{(t-1)}) - \sum_{i=1}^n \frac{p_i}{np_i}\nabla f_i(\boldsymbol{\phi}_c^{(t-1)}) + \frac{1}{n}\sum_{k=1}^n \nabla f_k(\boldsymbol{\phi}_k^{(t-1)}) \\
&= \frac{1}{n}\sum_{i=1}^n \nabla f_i(\mathbf{w}^{(t-1)}) - \frac{1}{n}\sum_{i=1}^n \nabla f_i(\boldsymbol{\phi}_c^{(t-1)}) + \frac{1}{n}\sum_{k=1}^n \nabla f_k(\boldsymbol{\phi}_k^{(t-1)}) \\
&= \nabla F(\mathbf{w}^{(t-1)}).
\end{aligned}$$

■

4.8.3 Proof of Lemma 4.7.3

Proof. To prove Lemma 4.7.3, we begin by computing $\mathbb{E}_i[\mathbf{g}_i - \nabla F(\mathbf{w}^*)]$ with $\mathbf{g}_i$ defined in (4.7.1) and Lemma 4.7.2:

$$\mathbb{E}_i[\mathbf{g}_i - \nabla F(\mathbf{w}^*)] = \nabla F(\mathbf{w}^{(t-1)}) - \nabla F(\mathbf{w}^*). \qquad (4.8.10)$$

By variance decomposition that $\mathbb{E}\big[\,\|\mathbf{x}\|^2\,\big] = \mathbb{E}\big[\,\|\mathbf{x}-\mathbb{E}[\mathbf{x}]\|^2\,\big] + \big\|\mathbb{E}[\mathbf{x}]\big\|^2$ for all $\mathbf{x}$, using (4.8.10),

$$
\begin{aligned}
&\mathbb{E}_i\big[\,\|\mathbf{g}_i - \nabla F(\mathbf{w}^*)\|^2\,\big] \\
&= \mathbb{E}_i\Big[\big\|\mathbf{g}_i - \nabla F(\mathbf{w}^*) - \mathbb{E}_i[\mathbf{g}_i - \nabla F(\mathbf{w}^*)]\big\|^2\Big] + \Big\|\mathbb{E}_i\big[\mathbf{g}_i - \nabla F(\mathbf{w}^*)\big]\Big\|^2 \\
&= \mathbb{E}_i\Bigg[\Big\|\Big[\frac{1}{np_i}\nabla f_i(\mathbf{w}^{(t-1)}) - \frac{1}{np_i}\nabla f_i(\mathbf{w}^*) - \nabla F(\mathbf{w}^{(t-1)}) + \nabla F(\mathbf{w}^*)\Big] \\
&\quad - \Big[\frac{1}{np_i}\nabla f_i(\boldsymbol{\phi}_c^{(t-1)}) - \frac{1}{np_i}\nabla f_i(\mathbf{w}^*) + \nabla F(\mathbf{w}^*) - \frac{1}{n}\sum_{k=1}^n f_k(\boldsymbol{\phi}_k^{(t-1)})\Big]\Big\|^2\Bigg] + \|\nabla F(\mathbf{w}^{(t-1)}) - \nabla F(\mathbf{w}^*)\|^2.
\end{aligned}
\tag{4.8.11}
$$

Applying the property that $\|\mathbf{x}+\mathbf{y}\|^2 \le (1+\zeta)\,\|\mathbf{x}\|^2 + (1+\zeta^{-1})\,\|\mathbf{y}\|^2$ for all $\mathbf{x}, \mathbf{y}$, and $\zeta > 0$ to (4.8.11),

$$
\begin{aligned}
\mathbb{E}_i\big[\,\|\mathbf{g}_i - \nabla F(\mathbf{w}^*)\|^2\,\big] &\le \|\nabla F(\mathbf{w}^{(t-1)}) - \nabla F(\mathbf{w}^*)\|^2 \\
&\quad + (1+\zeta)\mathbb{E}_i\Big[\Big\|\frac{1}{np_i}\nabla f_i(\mathbf{w}^{(t-1)}) - \frac{1}{np_i}\nabla f_i(\mathbf{w}^*) - \nabla F(\mathbf{w}^{(t-1)}) + \nabla F(\mathbf{w}^*)\Big\|^2\Big] \\
&\quad + (1+\zeta^{-1})\mathbb{E}_i\Big[\Big\|\frac{1}{np_i}\nabla f_i(\boldsymbol{\phi}_c^{(t-1)}) - \frac{1}{np_i}\nabla f_i(\mathbf{w}^*) + \nabla F(\mathbf{w}^*) - \frac{1}{n}\sum_{k=1}^n f_k(\boldsymbol{\phi}_k^{(t-1)})\Big\|^2\Big].
\end{aligned}
\tag{4.8.12}
$$

To simplify terms on the right-hand side of (4.8.12) using variance decomposition, we have

$$
\begin{aligned}
&\mathbb{E}_i\Big[\Big\|\frac{1}{np_i}\nabla f_i(\mathbf{w}^{(t-1)}) - \frac{1}{np_i}\nabla f_i(\mathbf{w}^*) - \nabla F(\mathbf{w}^{(t-1)}) + \nabla F(\mathbf{w}^*)\Big\|^2\Big] \\
&= \mathbb{E}_i\Big[\Big\|\frac{1}{np_i}\nabla f_i(\mathbf{w}^{(t-1)}) - \frac{1}{np_i}\nabla f_i(\mathbf{w}^*) - \mathbb{E}_i\big[\nabla f_i(\mathbf{w}^{(t-1)}) - \nabla f_i(\mathbf{w}^*)\big]\Big\|^2\Big] \\
&= \mathbb{E}_i\Big[\Big\|\frac{1}{np_i}\nabla f_i(\mathbf{w}^{(t-1)}) - \frac{1}{np_i}\nabla f_i(\mathbf{w}^*)\Big\|^2\Big] - \Big\|\mathbb{E}_i\big[\nabla f_i(\mathbf{w}^{(t-1)}) - \nabla f_i(\mathbf{w}^*)\big]\Big\|^2 \\
&= \mathbb{E}_i\Big[\Big\|\frac{1}{np_i}\nabla f_i(\mathbf{w}^{(t-1)}) - \frac{1}{np_i}\nabla f_i(\mathbf{w}^*)\Big\|^2\Big] - \big\|\nabla F(\mathbf{w}^{(t-1)}) - \nabla F(\mathbf{w}^*)\big\|^2,
\end{aligned}
\tag{4.8.13}
$$

and we obtain the following inequality by dropping a non-positive term:

$$
\begin{aligned}
&\mathbb{E}_i\Big[\Big\|\frac{1}{np_i}\nabla f_i(\boldsymbol{\phi}_c^{(t-1)}) - \frac{1}{np_i}\nabla f_i(\mathbf{w}^*) + \nabla F(\mathbf{w}^*) - \frac{1}{n}\sum_{k=1}^n f_k(\boldsymbol{\phi}_k^{(t-1)})\Big\|^2\Big] \\
&= \mathbb{E}_i\Big[\Big\|\frac{1}{np_i}\nabla f_i(\boldsymbol{\phi}_c^{(t-1)}) - \frac{1}{np_i}\nabla f_i(\mathbf{w}^*) - \mathbb{E}_i\big[\nabla f_i(\boldsymbol{\phi}_c^{(t-1)}) - \nabla f_i(\mathbf{w}^*)\big]\Big\|^2\Big] \\
&= \mathbb{E}_i\Big[\Big\|\frac{1}{np_i}\nabla f_i(\boldsymbol{\phi}_c^{(t-1)}) - \frac{1}{np_i}\nabla f_i(\mathbf{w}^*)\Big\|^2\Big] - \Big\|\mathbb{E}_i\big[\nabla f_i(\boldsymbol{\phi}_c^{(t-1)}) - \nabla f_i(\mathbf{w}^*)\big]\Big\|^2 \\
&\le \mathbb{E}_i\Big[\Big\|\frac{1}{np_i}\nabla f_i(\boldsymbol{\phi}_c^{(t-1)}) - \frac{1}{np_i}\nabla f_i(\mathbf{w}^*)\Big\|^2\Big].
\end{aligned}
\tag{4.8.14}
$$

Plugging (4.8.13) and (4.8.14) into (4.8.12), we complete the proof with

$$\mathbb{E}_i\big[\|\mathbf{g}_i - \nabla F(\mathbf{w}^*)\|^2\big] \leq (1+\zeta)\mathbb{E}_i\left[\left\|\frac{1}{np_i}\nabla f_i(\mathbf{w}^{(t-1)}) - \frac{1}{np_i}\nabla f_i(\mathbf{w}^*)\right\|^2\right] - \zeta\|\nabla F(\mathbf{w}^{(t-1)}) - \nabla F(\mathbf{w}^*)\|^2 + (1+\zeta^{-1})\mathbb{E}_i\left[\left\|\frac{1}{np_i}\nabla f_i(\boldsymbol{\phi}_c^{(t-1)}) - \frac{1}{np_i}\nabla f_i(\mathbf{w}^*)\right\|^2\right].$$

■

4.8.4 Proof of Lemma 4.7.4

Proof.

For the convenience of this proof, we first define a function

$$h(\mathbf{x}) = f(\mathbf{x}) - \frac{\mu}{2}\|\mathbf{x}\|^2. \tag{4.8.15}$$

Recall that f is strongly convex with the convexity parameter μ and its gradient is Lipschitz continuous with the constant L. By twice differentiating $h(\mathbf{w})$, we obtain that the gradient of h is Lipschitz continuous with the constant $L-\mu$.

By the property of f that is convex and has a Lipschitz continuous gradient: $f(\mathbf{y}) \geq f(\mathbf{x}) + \langle\nabla f(\mathbf{x}), \mathbf{y}-\mathbf{x}\rangle + \|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|^2/(2L)$ for all $\mathbf{x}$ and $\mathbf{y}$ [119] (Theorem 2.1.5), we have

$$h(\mathbf{x}) \geq h(\mathbf{y}) + \langle\nabla h(\mathbf{y}), \mathbf{x}-\mathbf{y}\rangle + \frac{1}{2(L-\mu)}\|\nabla h(\mathbf{x}) - \nabla h(\mathbf{y})\|^2.$$

By substitution of $h(\mathbf{x})$ according to (4.8.15),

$$f(\mathbf{x}) - \frac{\mu}{2}\|\mathbf{x}\|^2 \geq f(\mathbf{y}) - \frac{\mu}{2}\|\mathbf{y}\|^2 + \langle\nabla f(\mathbf{y}) - \mu\mathbf{y}, \mathbf{x}-\mathbf{y}\rangle + \frac{1}{2(L-\mu)}\big[\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|^2 + \mu^2\|\mathbf{y}-\mathbf{x}\|^2 + 2\mu\langle\nabla f(\mathbf{x}) - \nabla f(\mathbf{y}), \mathbf{y}-\mathbf{x}\rangle\big].$$

Re-arranging terms gives the following relation:

$$\begin{aligned}\langle\nabla f(\mathbf{y}), \mathbf{x}-\mathbf{y}\rangle \leq{}& f(\mathbf{x}) - f(\mathbf{y}) - \frac{1}{2(L-\mu)}\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|^2 - \frac{\mu}{L-\mu}\langle\nabla f(\mathbf{x}) - \nabla f(\mathbf{y}), \mathbf{y}-\mathbf{x}\rangle \\ &- \Big(\frac{\mu}{2}\|\mathbf{x}\|^2 - \frac{\mu}{2}\|\mathbf{y}\|^2 - \mu\langle\mathbf{y}, \mathbf{x}-\mathbf{y}\rangle\Big) - \frac{\mu^2}{2(L-\mu)}\|\mathbf{y}-\mathbf{x}\|^2.\end{aligned} \tag{4.8.16}$$

After simplifying terms on the right-hand side of (4.8.16) by

$$\begin{aligned}
&\frac{\mu}{2}\|\mathbf{x}\|^2 - \frac{\mu}{2}\|\mathbf{y}\|^2 - \mu\langle \mathbf{y}, \mathbf{x}-\mathbf{y}\rangle \\
&= \frac{\mu}{2}\|\mathbf{x}\|^2 - \frac{\mu}{2}\|\mathbf{y}\|^2 - \mu\langle \mathbf{x}, \mathbf{y}\rangle + \mu\|\mathbf{y}\|^2 \\
&= \frac{\mu}{2}\|\mathbf{y}-\mathbf{x}\|^2,
\end{aligned}$$

we are able to obtain the conclusion of Lemma 4.7.4:

$$\begin{aligned}
&\langle \nabla f(\mathbf{y}), \mathbf{x}-\mathbf{y}\rangle \\
&\leq f(\mathbf{x}) - f(\mathbf{y}) - \frac{1}{2(L-\mu)}\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|^2 - \frac{\mu}{L-\mu}\langle \nabla f(\mathbf{x}) - \nabla f(\mathbf{y}), \mathbf{y}-\mathbf{x}\rangle - \frac{L\mu}{2(L-\mu)}\|\mathbf{y}-\mathbf{x}\|^2.
\end{aligned}$$

■

4.8.5 Proof of Lemma 4.7.5

Proof. Recall that in Algorithm 1, at each iteration one component function f_i is sampled at probability p_i from n functions. Thus,

$$\mathbb{E}_i[f_i(\boldsymbol{\phi}_c^{(t)})] = p_i f_i(\boldsymbol{\phi}_c^{(t)}) + (1-p_i) f_i(\boldsymbol{\phi}_c^{(t-1)}). \tag{4.8.17}$$

Plugging (4.8.17) and $\boldsymbol{\phi}_c^{(t)} = \mathbf{w}^{(t-1)}$ into $\mathbb{E}_i[n^{-1}\sum_{i=1}^n L_i(np_i)^{-1} f_i(\boldsymbol{\phi}_c^{(t)})]$, we obtain

$$\begin{aligned}
&\mathbb{E}_i\Big[\frac{1}{n}\sum_{i=1}^n \frac{L_i}{np_i} f_i(\boldsymbol{\phi}_c^{(t)})\Big] \\
&= \frac{1}{n}\sum_{i=1}^n p_i \frac{L_i}{np_i} f_i(\mathbf{w}^{(t-1)}) + \frac{1}{n}\sum_{i=1}^n (1-p_i)\frac{L_i}{np_i} f_i(\boldsymbol{\phi}_c^{(t-1)}) \\
&= \frac{1}{n}\sum_{i=1}^n \frac{L_i}{n} f_i(\mathbf{w}^{(t-1)}) + \frac{1}{n}\sum_{i=1}^n \frac{(1-p_i)L_i}{np_i} f_i(\boldsymbol{\phi}_c^{(t-1)}).
\end{aligned}$$

■

Chapter 5

Stochastic Optimization for Big Data Analysis: Non-Strongly Convex Objectives

This Chapter further studies the same problem setting as depicted in Chapter 4 except for the fact that the smooth functions can be non-strongly convex, which is a more relaxed constraint than strong convexity. We propose a stochastic variance reduced alternating direction method of multipliers with the doubling-trick: SVR-ADMM-D. SVR-ADMM-D is a more efficient variant of the ADMM algorithm, which is scalable when multiple computational nodes are available to tackle the big data challenge [5]. The proposed algorithm leverages past variable values to progressively reduce the variance of the gradient estimator. The algorithm also incorporates the doubling-trick to enable itself to be a theoretically-sound anytime algorithm: it can be interrupted anytime while the training error converges to zero with increasing iterations. Experimental results on different real data sets demonstrate that SVR-ADMM-D converges faster than several baseline stochastic alternating direction methods of multipliers.

5.1 Introduction

We consider the constrained optimization problem for a composite function, which is the sum of two convex functions:

$$\begin{aligned} \underset{\mathbf{x},\mathbf{y}}{\text{minimize}} \quad & f(\mathbf{x}) + g(\mathbf{y}) \\ \text{subject to} \quad & \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{y} = \mathbf{c}, \end{aligned} \tag{5.1.1}$$

where $\mathbf{x} \in \mathbb{R}^{d_1}$, $\mathbf{y} \in \mathbb{R}^{d_2}$, $\mathbf{A} \in \mathbb{R}^{k \times d_1}$, $\mathbf{B} \in \mathbb{R}^{k \times d_2}$, $\mathbf{c} \in \mathbb{R}^k$, $f(\mathbf{x}) = (1/n)\sum_{i=1}^{n} f_i(\mathbf{x})$ is a sum of a finite number of convex and smooth component functions, and $g(\mathbf{y})$ can be non-differentiable. Here $g(\mathbf{y})$ is simple: the optimization problem

$$\underset{\mathbf{y}}{\text{minimize}} \quad g(\mathbf{y}) + \frac{\rho}{2}\|\mathbf{B}\mathbf{y} - \mathbf{c}\|^2 \tag{5.1.2}$$

has a closed-form solution, where $\rho > 0$.

The constrained optimization problem in (5.1.1) can be solved by the alternating direction method of multipliers (ADMM) [51, 17]. As pointed out by Boyd *et al.*, ADMM naturally fits the problem in (5.1.1) under the large-scale distributed convex optimization setting [17]. Under this setting, each of $f(\mathbf{x})$ and $g(\mathbf{y})$ matches a subproblem on a local data set with the constraint for ensuring the global consensus. Hence, ADMM is a significant tool for large-scale optimization problems. Besides, ADMM demonsrtated faster convergence than proximal gradient methods [144, 172] for solving several regularized empirical risk minimization problems with sparsity [153]. Here $f(\mathbf{x})$ and $g(\mathbf{y})$ correspond to the loss function and the regularizer respectively, and the constraint encodes the sparsity pattern of parameters. One can recover the lasso problem by setting $f(\mathbf{x})$ and $g(\mathbf{x})$ to be the squared loss function and the ℓ_1 norm with the constraint $\mathbf{x}-\mathbf{y}=\mathbf{0}$. Thus, when solving such a cornerstone problem in machine learning, ADMM seems an appealing option. Moreover, ADMM has been applied in various practical problems, such as compressed sensing, image deblurring, background extraction from surveillance video, and matrix completion with grossly corrupted data [53, 177, 52]. However, when the data set size is larger, the computational cost at each iteration generally becomes higher for ADMM, such as in solving the overlapping group lasso problem [128].

In view of this, online ADMM algorithms were first proposed to reduce the per-iteration computational cost [162, 153]. However, these algorithms may implicitly assume full accessibility of true data values without noises, which may not hold in practice [124]. Without such an assumption, the stochastic setting is a more natural way for achieving a low per-iteration computational cost over a large number of data instances. Therefore, stochastic ADMM algorithms have been recently studied and proposed [124, 193, 154]. To contrast ADMM against its stochastic variants, we refer to ADMM as batch ADMM in the rest of the chapter.

Transforming batch ADMM under the new stochastic setting, Ouyang *et al.* first proposed STOC-ADMM[1] [124]. Zhong and Kwok proposed stochastic average ADMM (SA-ADMM) that incrementally approximates the full gradient in the linearized ADMM formulation [193]. SA-ADMM integrates the stochastic average gradient [134] into the design of stochastic variants of ADMM. STOC-ADMM attains a sublinear rate of convergence for both strongly and non-strongly convex objectives. For non-strongly convex objectives, SA-ADMM attains a slightly accelerated sublinear rate of convergence than STOC-ADMM. However, theoretically it still remains unknown whether SA-ADMM can achieve an even faster convergence rate for strongly convex objectives. Another accelerated stochastic ADMM algorithm, stochastic dual coordinate ascent for ADMM (SDCA-ADMM), was proposed by Suzuki [154]. SDCA-ADMM applies the stochastic dual coordinate ascent [140] to transform batch ADMM into its stochastic variant. SDCA-ADMM attains

[1]We will consistently refer to this algorithm by Ouyang *et al.* [124] as *STOC-ADMM* while we use the term *stochastic ADMM algorithms* to refer to all stochastic variants of batch ADMM.

a linear rate of convergence for strongly convex objectives. Different from STOC-ADMM and SA-ADMM, SDCA-ADMM requires a dual formulation to solve the original problem. As we will demonstrate later in the experiments, the dual formulation may result in extra computational complexity at each iteration.

Another line of research in modern optimization is variance reduction methods for accelerating (proximal) stochastic gradient descent. Such accelerated algorithms include semi-stochastic gradient descent [80], permutable incremental gradient [38], minimization by incremental surrogate optimization [106], (proximal) stochastic variance reduced gradient [70, 173], and advanced stochastic average gradient method [37].

We propose a stochastic variance reduced alternating direction method of multipliers with the doubling-trick for non-strongly convex objectives: SVR-ADMM-D. SVR-ADMM-D leverages past variable values to progressively reduce the variance of the gradient estimator. The algorithm also incorporates the doubling-trick to enable itself to be a theoretically-sound anytime algorithm: it can be interrupted anytime while the training error converges to zero with increasing iterations.

After the completion of this chapter during the Ph.D. study, we learnt that a similar method was independently proposed by Zheng and Kwok [192]. Their work also applies the variance reduction technique into stochastic ADMM and obtains similar theoretical results to ours. However, the use of the doubling-trick sets this chapter apart from their work.

5.2 Notations and Assumptions

Here we define and state the notations and assumptions that are used throughout the technical discussions in this chapter. In the end, we briefly describe the assumptions that are made in the generalized problem setting: non-strongly convex objectives.

5.2.1 Notations

Let x_j be the j^{th} element of vector $\mathbf{x} = [x_1, \ldots, x_d]^\top \in \mathbb{R}^d$. We use $\|\mathbf{x}\| = \|\mathbf{x}\|_2 = \left(\sum_{k=1}^d x_k^2\right)^{1/2}$ to denote the ℓ_2 norm of a vector $\mathbf{x}$ and $\|\mathbf{x}\|_1 = \sum_{k=1}^d |x_k|$. An identity matrix is denoted as $\mathbf{I}$. For a matrix $\mathbf{A}$, we denote its minimum and maximum eigenvalues by $\lambda_m(\mathbf{A})$ and $\lambda_M(\mathbf{A})$ respectively, its minimum and maximum singular values by $\sigma_m(\mathbf{A})$ and $\sigma_M(\mathbf{A})$ respectively, its condition number by $\kappa(\mathbf{A}) = \sigma_M(\mathbf{A})/\sigma_m(\mathbf{A})$, and its spectral norm by $\|\mathbf{A}\| = \sigma_M(\mathbf{A})$. For positive definite matrix $\mathbf{P}$, we refer to $\|\mathbf{x}\|_{\mathbf{P}} = (\mathbf{x}^\top \mathbf{P}\mathbf{x})^{1/2}$ as the $\mathbf{P}$-quadratic norm [18]. Given a function $f(\mathbf{x})$, its subdifferential $\partial f(\mathbf{x})$ is the set of all its subderivatives $f'(\mathbf{x})$, and its gradient is denoted by $\nabla f(\mathbf{x})$ if $f(\mathbf{x})$ is differentiable. We use the superscript $*$ to denote the optimal value of a variable, *e.g.*, $\mathbf{x}^*$ is the optimal value of a variable $\mathbf{x}$.

5.2.2 Assumptions

We consider the generalized problem setting of non-strongly convex objectives. To illustrate this type of convex objectives, we present the following assumptions on functions $f(\mathbf{x})$ and $g(\mathbf{y})$ in the constrained optimization problem in (5.1.1). These assumptions are mild and can be verified in many regularized empirical risk minimization problems for machine learning.

Assumption 5.2.1 (Lipschitz continuous gradient) *Each gradient $\nabla f_i(\mathbf{x})$ is Lipschitz continuous with constant L_i, i.e., for all $\mathbf{x}$ and $\mathbf{y}$ we have*

$$\|\nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y})\| \leq L_i \|\mathbf{x} - \mathbf{y}\|.$$

Hence, $\nabla f(\mathbf{x})$ is also Lipschitz continuous with constant L, i.e., for all $\mathbf{x}$ and $\mathbf{y}$ we have

$$\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\| \leq L \|\mathbf{x} - \mathbf{y}\|.$$

Note that $L \leq L_M$, where $L_M = \max_i L_i$.

Assumption 5.2.2 (Convexity) *Functions $f_i(\mathbf{x})$ and $g(\mathbf{x})$ are convex, i.e., for all $\mathbf{x}$ and $\mathbf{y}$ we have*

$$f_i(\mathbf{y}) - f_i(\mathbf{x}) - \langle \nabla f_i(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle \geq 0 \quad \textit{and} \quad g(\mathbf{y}) - g(\mathbf{x}) - \langle g'(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle \geq 0.$$

Note that $f(\mathbf{x})$ is also convex, *i.e.*, for all $\mathbf{x}$ and $\mathbf{y}$ we have

$$f(\mathbf{y}) - f(\mathbf{x}) - \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle \geq 0.$$

5.3 Background of Batch ADMM and STOC-ADMM

Our proposed algorithms enrich the options in the existing pool of stochastic ADMM algorithms. Here we provide the background of batch ADMM and its stochastic variant that is essential for the design and analysis of our algorithms.

The constrained optimization problem in (5.1.1) can be solved by batch ADMM [17]. Here we describe the update steps in batch ADMM. To solve the problem in (5.1.1), batch ADMM performs the following

updates at an iteration t:

$$\mathbf{x}_t = \underset{\mathbf{x}}{\operatorname{argmin}}\, f(\mathbf{x}) + \frac{\rho}{2}\|\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{y}_{t-1} - \mathbf{c} + \mathbf{z}_{t-1}\|^2, \tag{5.3.1}$$

$$\mathbf{y}_t = \underset{\mathbf{y}}{\operatorname{argmin}}\, g(\mathbf{y}) + \frac{\rho}{2}\|\mathbf{A}\mathbf{x}_t + \mathbf{B}\mathbf{y} - \mathbf{c} + \mathbf{z}_{t-1}\|^2, \tag{5.3.2}$$

$$\mathbf{z}_t = \mathbf{z}_{t-1} + \mathbf{A}\mathbf{x}_t + \mathbf{B}\mathbf{y}_t - \mathbf{c}, \tag{5.3.3}$$

where ρ is a pre-defined penalty parameter. Note that (5.3.1) and (5.3.2) are updates for the primal variables $\mathbf{x}$ and $\mathbf{y}$, and (5.3.3) is the update for the dual variable $\mathbf{z}$ [17]. For the optimization problem in (5.1.1), let $\mathbf{x}^*$ and $\mathbf{y}^*$ be the optimal values of the primal variables $\mathbf{x}$ and $\mathbf{y}$, and $\mathbf{z}^*$ be the optimal value of the dual variable $\mathbf{z}$. According to the necessary and sufficient optimality conditions, we have

$$\mathbf{A}\mathbf{x}^* + \mathbf{B}\mathbf{y}^* = \mathbf{c}, \tag{5.3.4}$$

$$\nabla f(\mathbf{x}^*) + \rho\mathbf{A}^\top\mathbf{z}^* = 0, \tag{5.3.5}$$

$$g'(\mathbf{y}^*) + \rho\mathbf{B}^\top\mathbf{z}^* = 0, \tag{5.3.6}$$

where (5.3.4) is by the primal feasibility, while (5.3.5) and (5.3.6) are obtained by the Lagrangian optimality [17]. The optimality conditions (5.3.4)—(5.3.6) are useful for the convergence analysis of the proposed algorithms in the remaining technical discussions of this chapter.

Since $g(\mathbf{y})$ is a simple function and the problem in (5.3.2) is in the same form of (5.1.2), there is a closed-form solution for (5.3.2). However, the problem in (5.3.1) may not have a closed-form solution, such as when $f(\mathbf{x})$ is nonlinear. Hence, generalized linearization approaches are usually considered to replace (5.3.1) with

$$\mathbf{x}_t^{(s)} = \underset{\mathbf{x}}{\operatorname{argmin}}\langle\nabla f(\mathbf{x}_{t-1}), \mathbf{x}\rangle + \frac{\rho}{2}\big\|\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{y}_{t-1} - c + \mathbf{z}_{t-1}\big\|^2 + \frac{1}{2\eta}\big\|\mathbf{x} - \mathbf{x}_{t-1}\big\|_{\mathbf{P}}^2 \tag{5.3.7}$$

as the update for the primal variable $\mathbf{x}$. The parameter $\eta > 0$ in (5.3.7) denotes the step size.

Recall the problem (5.1.1) where $f(\mathbf{x}) = (1/n)\sum_{i=1}^n f_i(\mathbf{x})$ and n is the number of convex and smooth component functions $f_i(\mathbf{x})$. The above batch ADMM is less feasible for solving the problem (5.1.1) when n is larger. This is because at each iteration the computational cost for the primal variable update step (5.3.7) is higher, especially when evaluating the gradient of $f(\mathbf{x})$ with a larger n. Since n corresponds to the number of the provided data instances, it can be large in practice.

To this end, STOC-ADMM, a stochastic variant of ADMM was proposed [124]. In STOC-ADMM, only the update step for the primal variable $\mathbf{x}$ is re-designed. In contrast to (5.3.7) where $\nabla f(\mathbf{x})$ is evaluated, in STOC-ADMM a component function $f_i(\mathbf{x})$ is sampled at random then $\nabla f_i(\mathbf{x})$ replaces $\nabla f(\mathbf{x})$ in (5.3.7):

$$\mathbf{x}_t = \underset{\mathbf{x}}{\operatorname{argmin}} \langle \nabla f_i(\mathbf{x}_{t-1}), \mathbf{x} \rangle + \frac{\rho}{2} \left\| \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{y}_{t-1} - c + \mathbf{z}_{t-1} \right\|^2 + \frac{1}{2\eta} \left\| \mathbf{x} - \mathbf{x}_{t-1} \right\|_{\mathbf{P}}^2. \tag{5.3.8}$$

According to the update step for the primal variable $\mathbf{x}$ in (5.3.8), STOC-ADMM needs to evaluate $\nabla f_i(\mathbf{x})$ at each iteration. The gradient estimator $\nabla f_i(\mathbf{x})$ is evaluated based on a randomly sampled function $f_i(\mathbf{x})$ and is used to approximate the gradient $\nabla f(\mathbf{x}) = (1/n) \sum_{i=1}^{n} \nabla f_i(\mathbf{x})$. Thus, the per-iteration computational cost of STOC-ADMM is lighter than that of batch ADMM. This makes STOC-ADMM feasible for solving constrained optimization problems over a large number of data instances.

However, since random sampling introduces variance for the stochastic gradient estimator $\nabla f_i(\mathbf{x})$, STOC-ADMM has to choose a diminishing step size η to ensure asymptotic convergence [124]. Such a step size decays to zero and leads to slower convergence: STOC-ADMM can only attain a sublinear rate of convergence. In Section 5.4, we will propose new stochastic variants of ADMM with constant step sizes and faster convergence.

5.4 The SVR-ADMM-D Algorithm

We start by shedding light on the key insight behind the design for the new algorithm. Then we propose and describe our algorithms under two generalized problem settings: strongly convex objectives and non-strongly convex objectives.

5.4.1 Key Insight

As discussed in Section 5.3, STOC-ADMM has to choose a diminishing step size to mitigate the variance of the gradient estimator $\nabla f_i(\mathbf{x})$ introduced by random sampling. Such a decaying step size results in slower convergence.

In view of this, we are interested in progressively reducing the variance of the gradient estimator throughout iterations with constant step sizes. To illustrate the idea of variance reduction, suppose that we need to approximate a random variable ξ based on observed values of both random variables ξ and ψ. Let us denote the approximated estimator for ξ by

$$\widehat{\xi} = \xi - \psi + \mathbb{E}[\psi]. \tag{5.4.1}$$

Note that the approximated estimator $\widehat{\xi}$ for ξ is unbiased because the following equality holds:

$$\mathbb{E}\big[\widehat{\xi}\big] = \mathbb{E}[\xi]. \tag{5.4.2}$$

Since $\mathbb{E}[\psi]$ is a constant, we obtain the variance of the approximated estimator $\widehat{\xi}$ as the following relation:

$$\mathrm{Var}\left[\widehat{\xi}\right] = \mathrm{Var}[\xi] + \mathrm{Var}[\psi] - 2\,\mathrm{Cov}[\xi, \psi]. \tag{5.4.3}$$

At a high level, according to (5.4.3), when $\mathrm{Cov}[\xi, \psi]$ is large enough, the variance of the approximated estimator $\widehat{\xi}$ can be reduced. Thus, a good starting point would be choosing a random variable ψ that is highly correlated with ξ.

Recall that the gradient estimator is to be re-designed with an approximated stochastic gradient estimator. To choose a random variable that is highly correlated with the stochastic gradient, we consider taking snapshots of the stochastic gradient throughout all the stages of iterations with a stage index s. Later we will describe how such snapshots are taken in the proposed algorithms.

Plugging the stage index s into the stochastic gradient in (5.3.8), we denote by $\mathbf{h}_{t-1}^{(s)}$ the approximated estimator for the stochastic gradient $\nabla f_i(\mathbf{x}_{t-1}^{(s)})$ at a step $t-1$ of the stage s. Denote by $\nabla f_i(\widetilde{\mathbf{x}}^{(s-1)})$ the snapshot of the stochastic gradient taken in the previous stage $s-1$. Recall that an approximated estimator in the form (5.4.1) is unbiased according to (5.4.2). Following the same form we have

$$\mathbf{h}_{t-1}^{(s)} = \nabla f_i(\mathbf{x}_{t-1}^{(s)}) - \nabla f_i(\widetilde{\mathbf{x}}^{(s-1)}) + \mathbb{E}[\nabla f_i(\widetilde{\mathbf{x}}^{(s-1)})]. \tag{5.4.4}$$

Recall that $\nabla f_i(\widetilde{\mathbf{x}}^{(s-1)})$ seems to be highly correlated with $\nabla f_i(\mathbf{x}_{t-1}^{(s)})$ given that the snapshots of stochastic gradients are taken properly. To transform this intuitive understanding into formal rigor, we can formalize the progressively reduced variance of $\mathbf{h}_{t-1}^{(s)}$ over iterations in the stochastic ADMM settings. According to Lemma 5.6.4 and Lemma 5.6.5 (the original proof is available in the work by Johnson and Zhang [70]),

$$\begin{aligned}
\mathrm{Var}[\mathbf{h}_{t-1}^{(s)}] &= \mathbb{E}\big[\big\|\mathbf{h}_{t-1}^{(s)} - \mathbb{E}\big[\mathbf{h}_{t-1}^{(s)}\big]\big\|^2\big] \\
&\leq 4L_M\Big[\big[f(\mathbf{x}_{t-1}^{(s)}) - f(\mathbf{x}^*) - \langle \nabla f(\mathbf{x}^*), \mathbf{x}_{t-1}^{(s)} - \mathbf{x}^* \rangle\big] \\
&\qquad + \big[f(\widetilde{\mathbf{x}}^{(s-1)}) - f(\mathbf{x}^*) - \langle \nabla f(\mathbf{x}^*), \widetilde{\mathbf{x}}^{(s-1)} - \mathbf{x}^* \rangle\big]\Big],
\end{aligned} \tag{5.4.5}$$

Algorithm 2 SVR-ADMM-D: stochastic variance reduced alternating direction method of multipliers with the doubling-trick for non-strongly convex objectives

1: **Inputs:** $\rho > 0$, $\eta > 0$, $m^{(0)} \in \{x/2 \mid x \in \mathbb{Z} \wedge x > 0\}$, $\mathbf{P} \succeq \mathbf{I}$
2: **Initialize:** $\widetilde{\mathbf{x}}^{(0)}$, $\widetilde{\mathbf{y}}^{(0)}$, $\widetilde{\mathbf{z}}^{(0)}$
3: $\mathbf{x}_0^{(1)} \leftarrow \widetilde{\mathbf{x}}^{(0)}$
4: $\mathbf{z}_0^{(1)} \leftarrow \widetilde{\mathbf{z}}^{(0)}$
5: **for** $s = 1, 2, \ldots$ **do**
6: $\quad \mathbf{y}_0^{(s)} \leftarrow \widetilde{\mathbf{y}}^{(s-1)}$
7: $\quad m^{(s)} \leftarrow 2m^{(s-1)}$
8: $\quad$ **for** $t = 1, 2, \ldots, m^{(s)}$ **do**
9: $\quad\quad \mathbf{y}_t^{(s)} \leftarrow \operatorname{argmin}_{\mathbf{y}} g(\mathbf{y}) + \rho \left\| \mathbf{A}\mathbf{x}_{t-1}^{(s)} + \mathbf{B}\mathbf{y} - \mathbf{c} + \mathbf{z}_{t-1}^{(s)} \right\|^2 / 2$
10: $\quad\quad$ sample i from $\{1, \ldots, n\}$ uniformly at random with replacement
11: $\quad\quad \mathbf{h}_{t-1}^{(s)} \leftarrow \nabla f_i(\mathbf{x}_{t-1}^{(s)}) - \nabla f_i(\widetilde{\mathbf{x}}^{(s-1)}) + \nabla f(\widetilde{\mathbf{x}}^{(s-1)})$
12: $\quad\quad \mathbf{x}_t^{(s)} \leftarrow \operatorname{argmin}_{\mathbf{x}} \langle \mathbf{h}_{t-1}^{(s)}, \mathbf{x} \rangle + \rho \left\| \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{y}_t^{(s)} - \mathbf{c} + \mathbf{z}_{t-1}^{(s)} \right\|^2 / 2 + \left\| \mathbf{x} - \mathbf{x}_{t-1}^{(s)} \right\|_{\mathbf{P}}^2 / (2\eta)$
13: $\quad\quad \mathbf{z}_t^{(s)} \leftarrow \mathbf{z}_{t-1}^{(s)} + \mathbf{A}\mathbf{x}_t^{(s)} + \mathbf{B}\mathbf{y}_t^{(s)} - \mathbf{c}$
14: $\quad$ **end for**
15: $\quad \widetilde{\mathbf{x}}^{(s)} \leftarrow (1/m^{(s)}) \sum_{t=1}^{m^{(s)}} \mathbf{x}_t^{(s)}$
16: $\quad \widetilde{\mathbf{y}}^{(s)} \leftarrow (1/m^{(s)}) \sum_{t=1}^{m^{(s)}} \mathbf{y}_t^{(s)}$
17: $\quad \mathbf{x}_0^{(s+1)} \leftarrow \mathbf{x}_{m^{(s)}}^{(s)}$
18: $\quad \mathbf{z}_0^{(s+1)} \leftarrow \mathbf{z}_{m^{(s)}}^{(s)}$
19: **end for**
20: **Outputs:** $\widetilde{\mathbf{x}}^{(s)}$, $\widetilde{\mathbf{y}}^{(s)}$

where L_M is a positive constant. The right-hand side of the inequality in (5.4.5) is non-negative due to the convexity of the $f(\mathbf{x})$. When $\mathbf{x}_{t-1}^{(s)}$ and $\widetilde{\mathbf{x}}^{(s-1)}$ approximate the optimal value $\mathbf{x}^*$, the upper bound of the variance of the re-designed gradient estimator $\mathbf{h}_{t-1}^{(s)}$ reduces from a positive value to zero.

We propose SVR-ADMM-D in Algorithm 2 for non-strongly convex objectives. This algorithm is a stochastic variant of ADMM with constant step sizes and faster convergence.

5.4.2 SVR-ADMM-D for Non-Strongly Convex Objectives

For non-strongly convex objectives, we propose SVR-ADMM-D in Algorithm 2.

Algorithm 2 takes snapshots of the primal and dual variables throughout all the stages of iterations. Leveraging such snapshots $\widetilde{\mathbf{x}}^{(s)}$, $\widetilde{\mathbf{y}}^{(s)}$, and $\widetilde{\mathbf{z}}^{(s)}$ is also inspired by our discussions in Section 5.4.1. At the termination of the iterations, the final snapshots of the primal variables are the outputs of this algorithm.

A salient feature of Algorithm 2 is the doubling-trick: the doubling growth in the number of steps between consecutive iteration stages. An iteration stage s consists of $m^{(s)}$ steps. Line 7 of Algorithm 2 doubles the number of iteration steps at the beginning of every stage. The doubling growth in iteration steps over stages sets Algorithm 2 apart from all the stochastic ADMM algorithms reviewed in Section 5.1. This doubling-trick was also invoked by some other non-ADMM algorithm [6].

Besides, Algorithm 2 does not take snapshots of the dual variable $\mathbf{z}$. At the step 0 of a stage $s+1$, iterates $\mathbf{x}_0^{(s+1)}$ and $\mathbf{z}_0^{(s+1)}$ are initialized as the iterates $\mathbf{x}_{m^{(s)}}^{(s)}$ and $\mathbf{z}_{m^{(s)}}^{(s)}$ at the final step of the stage s (Lines 17 and 18 of Algorithm 2).

Remark 5.4.1 *SVR-ADMM-D is an anytime algorithm for non-strongly convex objectives. When $f(\mathbf{x})$ in the problem* (5.1.1) *is non-strongly convex, such as in the lasso and logistic regression problems, appending a perturbation term $\lambda_2\|\mathbf{x}\|^2/2$ ($\lambda_2 > 0$) after $f(\mathbf{x})$ produces strongly convex objectives because the second-order derivative becomes $\lambda_2 > 0$. Thus, although SDCA-ADMM [154] is designed for strongly convex objectives, it may still apply to non-strongly convex after appending the perturbation term. However, the error of this algorithm converges to $\mathcal{O}(\lambda_2)$ rather than 0 over the iteration time. Since the error of SVR-ADMM-D converges to 0 over the iteration time as we will prove later, the iteration can be terminated at any time.*

Remark 5.4.2 *We illustrate the space complexity of SVR-ADMM-D by considering two concrete problems in machine learning. Recall the problem in* (5.1.1) *where $\mathbf{x} \in \mathbb{R}^{d_1}$, $\mathbf{y} \in \mathbb{R}^{d_2}$, $\mathbf{A} \in \mathbb{R}^{k\times d_1}$, $\mathbf{B} \in \mathbb{R}^{k\times d_2}$, and $\mathbf{c} \in \mathbb{R}^k$. When comparing space complexities of stochastic ADMM algorithms, the storage of design matrices $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{c}$ are required by all these algorithms but are typically not considered since they are usually sparse in practice. Consider using stochastic ADMM algorithms to solve the lasso problem [157]. In the canonical form of ADMM, $\mathbf{x}, \mathbf{y} \in \mathbb{R}^{d_1}$, $\mathbf{A} = \mathbf{I}_{d_1}$, $\mathbf{B} = -\mathbf{I}_{d_1}$, and $\mathbf{c} = \mathbf{0}$. Among those accelerated stochastic ADMM algorithms, SA-ADMM [193] and SDCA-ADMM [154] require space complexities of $\mathcal{O}(nd_1)$ and $\mathcal{O}(n)$ respectively, while the space complexity of SVR-ADMM-D is $\mathcal{O}(d_1)$. Consider using stochastic ADMM algorithms to solve the graph-guided fused lasso problem [75]. In the canonical form of ADMM, $\mathbf{x} \in \mathbb{R}^{d_1}$, $\mathbf{y} \in \mathbb{R}^k$, $\mathbf{A} = \mathbb{R}^{k\times d_1}$, $\mathbf{B} = -\mathbf{I}_k$, and $\mathbf{c} = \mathbf{0}$. Among those accelerated stochastic ADMM algorithms, SA-ADMM [193] and SDCA-ADMM [154] require space complexities of $\mathcal{O}(nd_1 + k)$ and $\mathcal{O}(n + k)$ respectively, while the space complexity of SVR-ADMM-D is $\mathcal{O}(d_1 + k)$. Hence, when the number of data instances is so large that it dominates the space complexity, SVR-ADMM-D is preferred because it enjoys a lower space complexity than both SA-ADMM and SDCA-ADMM.*

5.5 Main Theory

We present the main theory on the convergence of Algorithm 2 together with their iteration complexity bounds. We provide proof of our main theory in Section 5.6.

In the convergence analysis for Algorithm 2, all the expectations are taken over i sampled in the a step t of the a stage s, conditional on information prior to the step t of the stage s, such as $\mathbf{x}_{t-1}^{(s)}, \mathbf{y}_{t-1}^{(s)}, \mathbf{z}_{t-1}^{(s)}, \widetilde{\mathbf{x}}^{(s-1)}, \widetilde{\mathbf{y}}^{(s-1)}$, and $\widetilde{\mathbf{z}}^{(s-1)}$, unless otherwise stated.

5.5.1 Gap Function

We define a gap function according to a recent ADMM study by Ouyang *et al.* [125, Section 2.1]:

Definition 5.5.1 (Gap function)

$$Q(\mathbf{x},\mathbf{y}) = \big[f(\mathbf{x}) + g(\mathbf{y}) + \rho\langle \mathbf{A}\mathbf{x}+\mathbf{B}\mathbf{y}-\mathbf{c}, \mathbf{z}^*\rangle\big] - \big[f(\mathbf{x}^*) + g(\mathbf{y}^*) + \rho\langle \mathbf{A}\mathbf{x}^*+\mathbf{B}\mathbf{y}^*-\mathbf{c}, \mathbf{z}^*\rangle\big]. \tag{5.5.1}$$

Applying the optimality conditions in (5.3.4)—(5.3.6) and re-arranging terms, we obtain the following equivalent gap function that is the sum of two component gap functions $Q_f(\mathbf{x})$ and $Q_g(\mathbf{y})$:

$$\begin{aligned} Q(\mathbf{x},\mathbf{y}) &= Q_f(\mathbf{x}) + Q_g(\mathbf{y}), \quad \text{where} \\ Q_f(\mathbf{x}) &= f(\mathbf{x}) - f(\mathbf{x}^*) - \langle \nabla f(\mathbf{x}^*), \mathbf{x}-\mathbf{x}^*\rangle, \\ Q_g(\mathbf{y}) &= g(\mathbf{y}) - g(\mathbf{y}^*) - \langle g'(\mathbf{y}^*), \mathbf{y}-\mathbf{y}^*\rangle. \end{aligned} \tag{5.5.2}$$

Remark 5.5.2 *Suppose that Assumption 5.2.2 holds. In* (5.5.2)*, both component gap functions $Q_f(\mathbf{x})$ and $Q_g(\mathbf{y})$ are non-negative due to the convexity of these functions. Hence, the gap function $Q(\mathbf{x},\mathbf{y})$ is also non-negative.*

5.5.2 Convergence of Algorithm 2

We present the following theorem on the convergence results of SVR-ADMM-D.

Theorem 5.5.3 *Suppose that Assumptions 5.2.1 and 5.2.2 hold. For Algorithm 2, by setting $\eta = 1/(12L_M)$, where $L_M = \max_i L_i$, and with the gap function $Q(\mathbf{x},\mathbf{y})$ defined in* (5.5.2)*, we have*

$$\begin{aligned} \mathbb{E}\big[Q(\widetilde{\mathbf{x}}^{(s)}, \widetilde{\mathbf{y}}^{(s)})\big] \le \Big(\frac{1}{2}\Big)^s \Big[Q(\widetilde{\mathbf{x}}^{(0)}, \widetilde{\mathbf{y}}^{(0)}) + \frac{9L_M}{m^{(0)}} \big\|\widetilde{\mathbf{x}}^{(0)} - \mathbf{x}^*\big\|^2_{\mathbf{P}+\rho\mathbf{A}^\top\mathbf{A}/(12L_M)} \\ + \frac{1}{2m^{(0)}} Q_f(\widetilde{\mathbf{x}}^{(0)}) + \frac{3\rho}{4m^{(0)}} \big\|\widetilde{\mathbf{z}}^{(0)} - \mathbf{z}^*\big\|^2 \Big]. \end{aligned}$$

It is noteworthy that the step size η of SVR-ADMM-D is a constant. We present the following corollary for the derived iteration complexity bound of SVR-ADMM-D.

Corollary 5.5.4 *Suppose that Assumptions 5.2.1 and 5.2.2 hold and an error $\epsilon > 0$ is given. For Algorithm 2 with the gap function $Q(\mathbf{x},\mathbf{y})$ defined in* (5.5.2)*, suppose that constants C_1, C_2, and C_3 satisfy relations $Q(\widetilde{\mathbf{x}}^{(0)}, \widetilde{\mathbf{y}}^{(0)}) \le C_1$, $37\big\|\widetilde{\mathbf{x}}^{(0)} - \mathbf{x}^*\big\|^2/4 \le C_2$, and $3\big\|\widetilde{\mathbf{z}}^{(0)} - \mathbf{z}^*\big\|^2/4 \le C_3$. Let $\sigma_M(\mathbf{A})$ be the maximal singular value of matrix $\mathbf{A}$ and $L_M = \max_i L_i$. By setting $m^{(0)} = \big[C_2\big(L_M\|\mathbf{P}\| + \rho\sigma_M(\mathbf{A})^2/12\big) + C_3\rho\big]/C_1$*

and $\eta = 1/(12L_M)$*, when* $s \geq \log(C_1/\epsilon)$ *we have*

$$\mathbb{E}\big[Q(\widetilde{\mathbf{x}}^{(s)}, \widetilde{\mathbf{y}}^{(s)})\big] \leq \mathcal{O}(\epsilon),$$

and obtain the iteration complexity of Algorithm 2:

$$\mathcal{O}\Big(n \log \frac{1}{\epsilon} + \frac{L_M \|\mathbf{P}\| + \rho \sigma_M(\mathbf{A})^2}{\epsilon}\Big). \tag{5.5.3}$$

5.6 Proof of the Main Theory

In this section, we prove the main theory as delivered in Section 5.5.

First, let us introduce several important lemmas. The proof of all the following lemmas are provided in Section 5.9.

5.6.1 Minimizing x with the Proximal Operator

For brevity, we start by describing the SVR-ADMM-D x-minimization step (Line 12 of Algorithm 2) with the proximal operator as defined below.

Definition 5.6.1 (Proximal operator)

$$\mathrm{prox}_\eta(\mathbf{w}) = \operatorname*{argmin}_{\mathbf{x}} \frac{1}{2\eta} \|\mathbf{w} - \mathbf{x}\|^2 + \frac{1}{2\eta} \big\|\mathbf{x} - \mathbf{x}_{t-1}^{(s)}\big\|_{\mathbf{P}-\mathbf{I}}^2 + \frac{\rho}{2} \big\|\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{y}_t^{(s)} - \mathbf{c} + \mathbf{z}_{t-1}^{(s)}\big\|^2. \tag{5.6.1}$$

Remark 5.6.2 *It can be verified that the proximal operator in* (5.6.1) *is non-expansive,* i.e.*, for all* $\mathbf{x}$ *and* $\mathbf{y}$*, we have* $\|\mathrm{prox}_\eta(\mathbf{x}) - \mathrm{prox}_\eta(\mathbf{y})\| \leq \|\mathbf{x} - \mathbf{y}\|$.

Lemma 5.6.3 *With the proximal operator defined in* (5.6.1)*, for Algorithm 2, we have*

$$\mathbf{x}_t^{(s)} = \mathrm{prox}_\eta\big(\mathbf{x}_{t-1}^{(s)} - \eta \mathbf{h}_{t-1}^{(s)}\big).$$

5.6.2 Unbiasedness of Gradient Estimator $\mathbf{h}_{t-1}^{(s)}$

Recall (5.4.2) the unbiasedness of an approximated estimator. Indeed, the gradient estimator $\mathbf{h}_{t-1}^{(s)}$ is unbiased.

Lemma 5.6.4 *For Algorithm 2, the gradient estimator* $\mathbf{h}_{t-1}^{(s)}$ *is unbiased:*

$$\mathbb{E}\big[\mathbf{h}_{t-1}^{(s)}\big] = \nabla f\big(\mathbf{x}_{t-1}^{(s)}\big).$$

5.6.3 Reduced Variance of the Gradient Estimator

By Lemma 5.6.4, variance of the gradient estimator $\mathbf{h}_{t-1}^{(s)}$ in Algorithm 2 can be represented by $\mathbb{E}\big[\big\|\mathbf{h}_{t-1}^{(s)} - \nabla f(\mathbf{x}_{t-1}^{(s)})\big\|^2\big]$.

Recall that Section 5.4.1 elucidates the idea of the progressively reduced variance of the gradient estimator. We present a lemma to formalize this intuitive understanding.

Lemma 5.6.5 *Suppose that Assumptions 5.2.1 and 5.2.2 hold. With defining* $L_M = \max_i L_i$ *and* Q_f *in* (5.5.2)*, variance of the gradient estimator* $\mathbf{h}_{t-1}^{(s)}$ *in Algorithm 2 can be bounded as:*

$$\mathbb{E}\big[\big\|\mathbf{h}_{t-1}^{(s)} - \nabla f(\mathbf{x}_{t-1}^{(s)})\big\|^2\big] \leq 4L_M\big[Q_f(\mathbf{x}_{t-1}^{(s)}) + Q_f(\widetilde{\mathbf{x}}^{(s-1)})\big].$$

Lemma 5.6.5 on the reduced variance of the gradient estimator $\mathbf{h}_{t-1}^{(s)}$ is useful for theoretical development: the expected inner product term $\mathbb{E}\big[\big\langle \mathbf{h}_{t-1}^{(s)} - \nabla f(\mathbf{x}_{t-1}^{(s)}), \mathbf{x} - \mathbf{x}_t^{(s)}\big\rangle\big]$ can be bounded with the progressively reduced variance of $\mathbf{h}_{t-1}^{(s)}$ in the following lemma.

Lemma 5.6.6 *For Algorithm 2, and for all* $\mathbf{x}$*, we have*

$$\mathbb{E}\big[\big\langle \mathbf{h}_{t-1}^{(s)} - \nabla f(\mathbf{x}_{t-1}^{(s)}), \mathbf{x} - \mathbf{x}_t^{(s)}\big\rangle\big] \leq \eta\mathbb{E}\big[\big\|\mathbf{h}_{t-1}^{(s)} - \nabla f(\mathbf{x}_{t-1}^{(s)})\big\|^2\big].$$

5.6.4 Bounding Quadratic Norms

We go on to present a lemma for bounding quadratic norms that are from the last terms in Line 12 of Algorithm 2.

Lemma 5.6.7 *Suppose that Assumptions 5.2.1 and 5.2.2 hold. For Algorithm 2 with* $0 < \eta < 1/L$*, and for all* $\mathbf{x}$*, we have*

$$\big\|\mathbf{x} - \mathbf{x}_t^{(s)}\big\|_{\mathbf{P}}^2 - \big\|\mathbf{x} - \mathbf{x}_{t-1}^{(s)}\big\|_{\mathbf{P}}^2 \leq 2\eta\Big[\big\langle \mathbf{h}_{t-1}^{(s)} - \nabla f(\mathbf{x}_{t-1}^{(s)}) + \rho\mathbf{A}^\top \mathbf{z}_t^{(s)}, \mathbf{x} - \mathbf{x}_t^{(s)}\big\rangle + f(\mathbf{x}) - f(\mathbf{x}_t^{(s)})\Big].$$

5.6.5 Bounding the Expected Value of the Gap Function

Recall (5.5.2) that $Q(\mathbf{x}, \mathbf{y}) = Q_f(\mathbf{x}) + Q_g(\mathbf{y})$. The goal is to bound $\mathbb{E}\big[Q(\widetilde{\mathbf{x}}^{(s)}, \widetilde{\mathbf{y}}^{(s)})\big]$: the expected value of the gap function with respect to the output of Algorithm 2 at a stage s.

Note that the gap function $Q(\mathbf{x}, \mathbf{y})$ consists of two components. To begin with, the following lemma establish the bound for the expected value of the component gap function $Q_f(\mathbf{x})$.

Lemma 5.6.8 *Suppose that Assumptions 5.2.1 and 5.2.2 hold. For Algorithm 2, by setting $0 < \eta < 1/(4L_M)$, where $L_M = \max_i L_i$, and with the component gap function $Q_f(\mathbf{x})$ defined in (5.5.2), we have*

$$\mathbb{E}\big[Q_f(\widetilde{\mathbf{x}}^{(s)})\big] \leq \frac{1}{2m^{(s)}\eta(1-4L_M\eta)}\Big[\mathbb{E}\big[\big\|\mathbf{x}_0^{(s)} - \mathbf{x}^*\big\|_{\mathbf{P}}^2 - \big\|\mathbf{x}_{m^{(s)}}^{(s)} - \mathbf{x}^*\big\|_{\mathbf{P}}^2\big] + 8L_M\eta^2 Q_f(\mathbf{x}_0^{(s)}) \\ - 8L_M\eta^2\mathbb{E}\big[Q_f(\mathbf{x}_{m^{(s)}}^{(s)})\big] + 8L_M m^{(s)}\eta^2 Q_f(\widetilde{\mathbf{x}}^{(s-1)}) + 2\eta\sum_{t=1}^{m^{(s)}}\mathbb{E}\big[\big\langle \rho\mathbf{A}^\top(\mathbf{z}_t^{(s)} - \mathbf{z}^*), \mathbf{x}^* - \mathbf{x}_t^{(s)}\big\rangle\big]\Big].$$

Next, we present a lemma to bound the expected value of the other component gap function $Q_g(\mathbf{x})$.

Lemma 5.6.9 *Suppose that Assumption 5.2.2 holds. For Algorithm 2, with the component gap function $Q_g(\mathbf{y})$ defined in (5.5.2) we have*

$$\mathbb{E}\big[Q_g(\widetilde{\mathbf{y}}^{(s)})\big] \leq \frac{1}{2m^{(s)}\eta}\Big[-2\eta\sum_{t=1}^{m^{(s)}}\mathbb{E}\big[\big\langle \rho\mathbf{A}^\top(\mathbf{z}_t^{(s)} - \mathbf{z}^*), \mathbf{x}^* - \mathbf{x}_t^{(s)}\big\rangle\big] \\ + \eta\rho\mathbb{E}\big[\big\|\mathbf{x}_0^{(s)} - \mathbf{x}^*\big\|_{\mathbf{A}^\top\mathbf{A}}^2 - \big\|\mathbf{x}_{m^{(s)}}^{(s)} - \mathbf{x}^*\big\|_{\mathbf{A}^\top\mathbf{A}}^2\big] + \eta\rho\mathbb{E}\big[\big\|\mathbf{z}_0^{(s)} - \mathbf{z}^*\big\|^2 - \big\|\mathbf{z}_{m^{(s)}}^{(s)} - \mathbf{z}^*\big\|^2\big]\Big].$$

By combining the results of Lemma 5.6.8 and Lemma 5.6.9 on the expected values of the two component gap functions, we obtain the following lemma on the bound of the expected value of the gap function $Q(\widetilde{\mathbf{x}}^{(s)}, \widetilde{\mathbf{y}}^{(s)})$.

Lemma 5.6.10 *Suppose that Assumptions 5.2.1 and 5.2.2 hold. For Algorithm 2, by setting $0 < \eta < 1/(4L_M)$, where $L_M = \max_i L_i$, and with the gap function $Q(\mathbf{x}, \mathbf{y})$ and its component $Q_f(\mathbf{x})$ defined in (5.5.2), we have*

$$\mathbb{E}\big[Q(\widetilde{\mathbf{x}}^{(s)}, \widetilde{\mathbf{y}}^{(s)})\big] \\ \leq \frac{1}{2m^{(s)}\eta(1-4L_M\eta)}\Big[\mathbb{E}\big[\big\|\mathbf{x}_0^{(s)} - \mathbf{x}^*\big\|_{\mathbf{P}+\eta\rho\mathbf{A}^\top\mathbf{A}}^2 - \big\|\mathbf{x}_{m^{(s)}}^{(s)} - \mathbf{x}^*\big\|_{\mathbf{P}+\eta\rho\mathbf{A}^\top\mathbf{A}}^2\big] + 8L_M\eta^2 Q_f(\mathbf{x}_0^{(s)}) \\ - 8L_M\eta^2\mathbb{E}\big[Q_f(\mathbf{x}_{m^{(s)}}^{(s)})\big] + 8L_M m^{(s)}\eta^2 Q_f(\widetilde{\mathbf{x}}^{(s-1)}) + \eta\rho\mathbb{E}\big[\big\|\mathbf{z}_0^{(s)} - \mathbf{z}^*\big\|^2 - \big\|\mathbf{z}_{m^{(s)}}^{(s)} - \mathbf{z}^*\big\|^2\big]\Big].$$

With these lemmas, now we are ready to prove the main theory stated in Section 5.5 in the rest of this section.

5.6.6 Proof of Theorem 5.5.3

Proof. Recall the definition of the gap function $Q(\mathbf{x}, \mathbf{y})$ in (5.5.2) and Remark 5.5.2 that the component gap function $Q_g(\mathbf{y})$ is non-negative. Hence, $Q_f(\widetilde{\mathbf{x}}^{(s-1)}) \leq Q(\widetilde{\mathbf{x}}^{(s-1)}, \widetilde{\mathbf{y}}^{(s-1)})$. By Lemma 5.6.10, with $\mathbf{x}_{m^{(s)}}^{(s)} = \mathbf{x}_0^{(s+1)}$ (Line 17 of Algorithm 2) and $\mathbf{z}_{m^{(s)}}^{(s)} = \mathbf{z}_0^{(s+1)}$ (Line 18 of Algorithm 2), we have

$$
\begin{aligned}
&\mathbb{E}\big[Q(\widetilde{\mathbf{x}}^{(s)}, \widetilde{\mathbf{y}}^{(s)})\big] \\
&\leq \frac{1}{2m^{(s)}\eta(1-4L_M\eta)}\Big[\mathbb{E}\big[\big\|\mathbf{x}_0^{(s)} - \mathbf{x}^*\big\|^2_{\mathbf{P}+\eta\rho\mathbf{A}^\top\mathbf{A}} - \big\|\mathbf{x}_0^{(s+1)} - \mathbf{x}^*\big\|^2_{\mathbf{P}+\eta\rho\mathbf{A}^\top\mathbf{A}}\big] + 8L_M\eta^2 Q_f(\mathbf{x}_0^{(s)}) \\
&\qquad - 8L_M\eta^2\mathbb{E}\big[Q_f(\mathbf{x}_0^{(s+1)})\big] + \eta\rho\mathbb{E}\big[\big\|\mathbf{z}_0^{(s)} - \mathbf{z}^*\big\|^2 - \big\|\mathbf{z}_0^{(s+1)} - \mathbf{z}^*\big\|^2\big] + 8L_M m^{(s)}\eta^2 Q(\widetilde{\mathbf{x}}^{(s-1)}, \widetilde{\mathbf{y}}^{(s-1)})\Big].
\end{aligned}
$$

Multiplying both sides by 2, a re-arrangement of terms gives

$$
\begin{aligned}
&2\mathbb{E}\big[Q(\widetilde{\mathbf{x}}^{(s)}, \widetilde{\mathbf{y}}^{(s)})\big] \\
&\leq \frac{8L_M\eta}{1-4L_M\eta}Q(\widetilde{\mathbf{x}}^{(s-1)}, \widetilde{\mathbf{y}}^{(s-1)}) + \frac{1}{m^{(s)}\eta(1-4L_M\eta)}\Big[\big\|\mathbf{x}_0^{(s)} - \mathbf{x}^*\big\|^2_{\mathbf{P}+\eta\rho\mathbf{A}^\top\mathbf{A}} - \mathbb{E}\big[\big\|\mathbf{x}_0^{(s+1)} - \mathbf{x}^*\big\|^2_{\mathbf{P}+\eta\rho\mathbf{A}^\top\mathbf{A}}\big]\Big] \\
&\quad + \frac{8L_M\eta}{m^{(s)}(1-4L_M\eta)}\Big[Q_f(\mathbf{x}_0^{(s)}) - \mathbb{E}\big[Q_f(\mathbf{x}_0^{(s+1)})\big]\Big] + \frac{\rho}{m^{(s)}(1-4L_M\eta)}\Big[\big\|\mathbf{z}_0^{(s)} - \mathbf{z}^*\big\|^2 - \mathbb{E}\big[\big\|\mathbf{z}_0^{(s+1)} - \mathbf{z}^*\big\|^2\big]\Big].
\end{aligned}
$$

Leveraging the doubling-trick in Algorithm 2 that $m^{(s)} = 2m^{(s-1)}$ (Line 7 of Algorithm 2), with setting $\eta = 1/(12L_M)$, where $L_M = \max_i L_i$, we further arrange terms and obtain

$$
\begin{aligned}
&2\mathbb{E}\Big[Q(\widetilde{\mathbf{x}}^{(s)}, \widetilde{\mathbf{y}}^{(s)}) + \frac{9L_M}{m^{(s)}}\big\|\mathbf{x}_0^{(s+1)} - \mathbf{x}^*\big\|^2_{\mathbf{P}+\rho\mathbf{A}^\top\mathbf{A}/(12L_M)} + \frac{1}{2m^{(s)}}Q_f(\mathbf{x}_0^{(s+1)}) + \frac{3\rho}{4m^{(s)}}\big\|\mathbf{z}_0^{(s+1)} - \mathbf{z}^*\big\|^2\Big] \\
&\leq Q(\widetilde{\mathbf{x}}^{(s-1)}, \widetilde{\mathbf{y}}^{(s-1)}) + \frac{9L_M}{m^{(s-1)}}\big\|\mathbf{x}_0^{(s)} - \mathbf{x}^*\big\|^2_{\mathbf{P}+\rho\mathbf{A}^\top\mathbf{A}/(12L_M)} + \frac{1}{2m^{(s-1)}}Q_f(\mathbf{x}_0^{(s)}) + \frac{3\rho}{4m^{(s-1)}}\big\|\mathbf{z}_0^{(s)} - \mathbf{z}^*\big\|^2.
\end{aligned}
$$

By chaining over s, since the last three terms within the square brackets are non-negative (recall Remark 5.5.2 that $Q_f(\mathbf{x}) \geq 0$ for all $\mathbf{x}$), we drop these three non-negative terms on the left-hand side and obtain

$$
\begin{aligned}
\mathbb{E}\big[Q(\widetilde{\mathbf{x}}^{(s)}, \widetilde{\mathbf{y}}^{(s)})\big] \leq \Big(\frac{1}{2}\Big)^s\Big[&Q(\widetilde{\mathbf{x}}^{(0)}, \widetilde{\mathbf{y}}^{(0)}) + \frac{9L_M}{m^{(0)}}\big\|\mathbf{x}_0^{(1)} - \mathbf{x}^*\big\|^2_{\mathbf{P}+\rho\mathbf{A}^\top\mathbf{A}/(12L_M)} \\
&+ \frac{1}{2m^{(0)}}Q_f(\mathbf{x}_0^{(1)}) + \frac{3\rho}{4m^{(0)}}\big\|\mathbf{z}_0^{(1)} - \mathbf{z}^*\big\|^2\Big].
\end{aligned}
$$

Now we complete the proof according to Line 3 and 4 of Algorithm 2:

$$\mathbb{E}\big[Q(\widetilde{\mathbf{x}}^{(s)},\widetilde{\mathbf{y}}^{(s)})\big] \leq \left(\frac{1}{2}\right)^s \Big[Q(\widetilde{\mathbf{x}}^{(0)},\widetilde{\mathbf{y}}^{(0)}) + \frac{9L_M}{m^{(0)}}\big\|\widetilde{\mathbf{x}}^{(0)}-\mathbf{x}^*\big\|^2_{\mathbf{P}+\rho\mathbf{A}^\top\mathbf{A}/(12L_M)} + \frac{1}{2m^{(0)}}Q_f(\widetilde{\mathbf{x}}^{(0)}) + \frac{3\rho}{4m^{(0)}}\big\|\widetilde{\mathbf{z}}^{(0)}-\mathbf{z}^*\big\|^2\Big].$$

■

5.6.7 Proof of Corollary 5.5.4

Proof. Following the results of Theorem 5.5.3, we have

$$\begin{aligned}\mathbb{E}\big[Q(\widetilde{\mathbf{x}}^{(s)},\widetilde{\mathbf{y}}^{(s)})\big] &\leq \left(\frac{1}{2}\right)^s \Big[Q(\widetilde{\mathbf{x}}^{(0)},\widetilde{\mathbf{y}}^{(0)}) + \frac{9L_M}{m^{(0)}}\big\|\widetilde{\mathbf{x}}^{(0)}-\mathbf{x}^*\big\|^2_{\mathbf{P}+\rho\mathbf{A}^\top\mathbf{A}/(12L_M)} \\ &\quad + \frac{1}{2m^{(0)}}Q_f(\widetilde{\mathbf{x}}^{(0)}) + \frac{3\rho}{4m^{(0)}}\big\|\widetilde{\mathbf{z}}^{(0)}-\mathbf{z}^*\big\|^2\Big] \\ &\leq \left(\frac{1}{2}\right)^s \Big[Q(\widetilde{\mathbf{x}}^{(0)},\widetilde{\mathbf{y}}^{(0)}) + \frac{9L_M}{m^{(0)}}\big\|\widetilde{\mathbf{x}}^{(0)}-\mathbf{x}^*\big\|^2_{\mathbf{P}+\rho\mathbf{A}^\top\mathbf{A}/(12L_M)} \\ &\quad + \frac{L_M}{4m^{(0)}}\big\|\widetilde{\mathbf{x}}^{(0)}-\mathbf{x}^*\big\|^2_{\mathbf{P}+\rho\mathbf{A}^\top\mathbf{A}/(12L_M)} + \frac{3\rho}{4m^{(0)}}\big\|\widetilde{\mathbf{z}}^{(0)}-\mathbf{z}^*\big\|^2\Big],\end{aligned}$$

where the second inequality is obtained by the property $f(\mathbf{y}) - f(\mathbf{x}) - \langle\nabla f(\mathbf{x}), \mathbf{y}-\mathbf{x}\rangle \leq (L/2)\|\mathbf{x}-\mathbf{y}\|^2$ for all $\mathbf{x}$ and $\mathbf{y}$ under Assumptions 5.2.1 and 5.2.2 [119, Theorem 2.1.5] and relations $L \leq L_M$ and $\|\mathbf{x}\|^2 \leq \|\mathbf{x}\|^2_{\mathbf{P}+\rho\mathbf{A}^\top\mathbf{A}/(12L_M)}$ for all $\mathbf{x}$.

Suppose that constants C_1, C_2, and C_3 satisfy relations $Q(\widetilde{\mathbf{x}}^{(0)},\widetilde{\mathbf{y}}^{(0)}) \leq C_1$, $37\big\|\widetilde{\mathbf{x}}^{(0)}-\mathbf{x}^*\big\|^2/4 \leq C_2$, and $3\big\|\widetilde{\mathbf{z}}^{(0)}-\mathbf{z}^*\big\|^2/4 \leq C_3$, since

$$\big\|\widetilde{\mathbf{x}}^{(0)}-\mathbf{x}^*\big\|^2_{\mathbf{P}+\rho\mathbf{A}^\top\mathbf{A}/(12L_M)} \leq \big\|\widetilde{\mathbf{x}}^{(0)}-\mathbf{x}^*\big\|^2 \cdot \left\|\mathbf{P}+\frac{\rho\mathbf{A}^\top\mathbf{A}}{12L_M}\right\|$$

by the Cauchy-Schwarz inequality, we have

$$\begin{aligned}\frac{37L_M}{4m^{(0)}}\big\|\widetilde{\mathbf{x}}^{(0)}-\mathbf{x}^*\big\|^2_{\mathbf{P}+\rho\mathbf{A}^\top\mathbf{A}/(12L_M)} &\leq \frac{C_2L_M}{m^{(0)}}\left\|\mathbf{P}+\frac{\rho\mathbf{A}^\top\mathbf{A}}{12L_M}\right\| \\ &\leq \frac{C_2L_M}{m^{(0)}}\left(\|\mathbf{P}\|+\frac{\rho\|\mathbf{A}^\top\mathbf{A}\|}{12L_M}\right) \\ &= \frac{C_2L_M}{m^{(0)}}\left(\|\mathbf{P}\|+\frac{\rho\sigma_M(\mathbf{A})^2}{12L_M}\right).\end{aligned}$$

where with notations for extreme eigenvalues and singular values in Section 5.2.1 the equality is obtained by relations

$$\|\mathbf{A}^\top\mathbf{A}\| = \sigma_M(\mathbf{A}^\top\mathbf{A}) = \lambda_M(\mathbf{A}^\top\mathbf{A}) = \sigma_M(\mathbf{A})^2,$$

thus we get

$$\mathbb{E}\big[Q(\widetilde{\mathbf{x}}^{(s)}, \widetilde{\mathbf{y}}^{(s)})\big] \leq \Big(\frac{1}{2}\Big)^s \Big(C_1 + \frac{C_2 L_M \|\mathbf{P}\|}{m^{(0)}} + \frac{C_2\rho\sigma_M(\mathbf{A})^2}{12m^{(0)}} + \frac{C_3\rho}{m^{(0)}}\Big).$$

Setting $m^{(0)} = \big[C_2\big(L_M\|\mathbf{P}\| + \rho\sigma_M(\mathbf{A})^2/12\big) + C_3\rho\big]/C_1$, when $s \geq \log(C_1/\epsilon)$ we have

$$\mathbb{E}\big[Q(\widetilde{\mathbf{x}}^{(s)}, \widetilde{\mathbf{y}}^{(s)})\big] \leq \mathcal{O}(\epsilon).$$

The matrix computation cost involving the design matrices $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{c}$ can be assumed to be dominated by costly operations such as gradient evaluations [173]. Recall Line 11 of Algorithm 2 and for all $\mathbf{x}$: for each outer loop, gradient $\nabla f(\mathbf{x})$, which is an averaged sum of n component gradients $\nabla f_i(\mathbf{x})$, only needs to be evaluated once. For each inner loop, a randomly sampled component gradient $\nabla f_i(\mathbf{x})$ has to be evaluated twice in Line 11; thus, there are $2m^{(s)}$ evaluations of $\nabla f_i(\mathbf{x})$ for every outer loop.

Given $s \geq \log(C_1/\epsilon)$, the iteration complexity is

$$\mathcal{O}\bigg[n\log\Big(\frac{C_1}{\epsilon}\Big) + 2\sum_{s=1}^{\log(C_1/\epsilon)} m^{(s)}\bigg].$$

Since $m^{(s)} = 2m^{(s-1)}$ (Line 7 of Algorithm 2) and we set $m^{(0)} = \big[C_2\big(L_M\|\mathbf{P}\| + \rho\sigma_M(\mathbf{A})^2/12\big) + C_3\rho\big]/C_1$, the iteration complexity is equivalent to

$$\mathcal{O}\bigg[n\log\Big(\frac{C_1}{\epsilon}\Big) + \Big(\frac{2}{C_1}\Big)\cdot\Big(\frac{C_1}{\epsilon}\Big)\cdot\bigg[C_2\Big(L_M\|\mathbf{P}\| + \frac{\rho\sigma_M(\mathbf{A})^2}{12}\Big) + C_3\rho\bigg]\bigg].$$

With simplifications while dropping constants in the big O notation, we obtain the iteration complexity of Algorithm 2: $\mathcal{O}\big(n\log\epsilon^{-1} + (L_M\|\mathbf{P}\| + \rho\sigma_M(\mathbf{A})^2)/\epsilon\big)$. ∎

5.7 Evaluation

In this section, we implement SVR-ADMM-D and experimentally compare its convergence performance with those of baseline stochastic ADMM algorithms.

We begin by describing the implementation details of the x-minimization step for SVR-ADMM-D.

5.7.1 Linearized Preconditioned Approach for Implementation

We refer to Line 12 of Algorithm 2 as the SVR-ADMM-D x-minimization step. The analytical solution for $\mathbf{x}_t^{(s)}$ satisfies the relation

$$\mathbf{h}_{t-1}^{(s)} + \rho \mathbf{A}^\top \big(\mathbf{A}\mathbf{x}_t^{(s)} + \mathbf{B}\mathbf{y}_t^{(s)} - \mathbf{c} + \mathbf{z}_{t-1}^{(s)}\big) + \frac{1}{\eta}\mathbf{P}\big(\mathbf{x}_t^{(s)} - \mathbf{x}_{t-1}^{(s)}\big) = 0. \tag{5.7.1}$$

A re-arrangement of terms gives

$$\mathbf{x}_t^{(s)} = \left(\frac{1}{\eta}\mathbf{P} + \rho \mathbf{A}^\top \mathbf{A}\right)^{-1} \left[\frac{1}{\eta}\mathbf{P}\mathbf{x}_{t-1}^{(s)} - \big[\mathbf{h}_{t-1}^{(s)} + \rho \mathbf{A}^\top (\mathbf{B}\mathbf{y}_t^{(s)} - \mathbf{c} + \mathbf{z}_{t-1}^{(s)})\big]\right]. \tag{5.7.2}$$

Note that the step size η is a fixed constant and the term $(\mathbf{P}/\eta + \rho \mathbf{A}^\top \mathbf{A})^{-1}$ can be calculated and stored beforehand. However, such a memory consumption may be costly and the inversion operation may be computationally expensive. In view of this, a linearized preconditioned approach is recommended in the implementation [125]. Specifically, equivalent to (5.7.2) we have

$$\mathbf{x}_t^{(s)} = \mathbf{x}_{t-1}^{(s)} - \frac{\eta}{\theta}\big[\mathbf{h}_{t-1}^{(s)} + \rho \mathbf{A}^\top (\mathbf{A}\mathbf{x}_{t-1}^{(s)} + \mathbf{B}\mathbf{y}_t^{(s)} - \mathbf{c} + \mathbf{z}_{t-1}^{(s)})\big]$$

by setting

$$\mathbf{P} = \theta \mathbf{I} - \eta \rho \mathbf{A}^\top \mathbf{A} \succeq \mathbf{I}, \quad \text{where} \quad \theta \geq \eta \rho \|\mathbf{A}^\top \mathbf{A}\| + 1.$$

We highlight that the linearized preconditioned approach for implementation does not affect the theoretical results presented in this chapter. We adopt this approach in the implementation for our experiments.

5.7.2 Problem and Measures

We define the problem and measures for convergence used in the empirical evaluation. Specifically, we consider a classification problem of the graph-guided fused lasso with the logistic loss in the setting of non-strongly convex objectives [75]. We evaluate the convergence performance of the proposed SVR-ADMM-D algorithm in comparison with baseline stochastic ADMM algorithms in solving this problem.

Graph-Guided Fused Lasso

For evaluation, we consider the graph-guided fused lasso problem [75]. This problem instantiates from the generalized lasso framework [160]. The key feature of the graph-guided fused lasso is that it appends a fused penalty term [159] to the original loss function. This new term penalizes the differences between pairs of features with the ℓ_1-regularization.

Consider a given training data set $\big\{(\boldsymbol{\xi}_1, \psi_1), (\boldsymbol{\xi}_2, \psi_2), \ldots, (\boldsymbol{\xi}_n, \psi_n)\big\}$ consisting of n data instances, where $\boldsymbol{\xi}_i$ and ψ_i are the feature vector and the class label for the i^{th} instance. The class label is binary and takes a value from 0 and 1. Given the feature space, consider an undirected edge exists between a pair of features and such a collection of edges forms a set $\mathcal{E}$. The graph-guided fused lasso problem is formalized in the following original form:

$$\underset{\mathbf{x}}{\text{minimize}} \quad \frac{1}{n}\sum_{i=1}^{n} \log\left[1 + \exp(\langle \mathbf{x}^\top, \boldsymbol{\xi}_i \rangle)\right] - \psi_i \mathbf{x}^\top \boldsymbol{\xi}_i + \gamma_1 \sum_{(i,j)\in\mathcal{E}} W_{ij}|x_i - x_j|, \tag{5.7.3}$$

where $\gamma_1 > 0$ is a pre-defined regularization constant in the problem, and W_{ij} can be the pre-defined similarity measure between the i^{th} and the j^{th} features. The regularizer $\gamma_1 = 10^{-3}$ is fixed in the evaluation.

We go on to re-formulate the original form of the graph-guided fused lasso problem in (5.7.3) as in the canonical ADMM form. Suppose that the cardinality of the set of edges $\mathcal{E}$ is k. For any edge $(i, j) \in \mathcal{E}$, it has a unique index p, where $p \in \{p \in \mathbb{Z} \mid 1 \leq p \leq k\}$. Define the design matrix

$$\mathbf{A} \in \mathbb{R}^{k \times d_1}, \tag{5.7.4}$$

where $A_{pi} = W_{ij}$, $A_{pj} = -W_{ij}$, and the rest of elements are 0. The fused penalty term in (5.7.3) can be re-formulated as

$$g(\mathbf{A}\mathbf{x}) = \gamma_1 \|\mathbf{A}\mathbf{x}\|_1 = \gamma_1 \sum_{(i,j)\in\mathcal{E}} W_{ij}|x_i - x_j|. \tag{5.7.5}$$

We can obtain the re-formulated graph-guided fused lasso problem in the canonical ADMM form:

$$\begin{aligned} \underset{\mathbf{x},\mathbf{y}}{\text{minimize}} \quad & f(\mathbf{x}) + g(\mathbf{y}) \\ \text{subject to} \quad & \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{y} = \mathbf{c}, \end{aligned} \tag{5.7.6}$$

Table 5.1: Summary statistics of three real data sets in the experiments.

Data Set	**20news**	**a9a**	**ijcnn1**
#Training Instances	10,066	32,561	49,990
#Features	100	123	22

where $\mathbf{x} \in \mathbb{R}^{d_1}$, $\mathbf{y} \in \mathbb{R}^k$, the design matrix $\mathbf{A}$ is given in (5.7.4), $\mathbf{B} = -\mathbf{I}_k$, $\mathbf{c} = \mathbf{0}$, $f(\mathbf{x}) = (1/n)\sum_{i=1}^n f_i(\mathbf{x})$, where

$$f_i(\mathbf{x}) = \log\left[1 + \exp(\langle \mathbf{x}^\top, \boldsymbol{\xi}_i \rangle)\right] - \psi_i \mathbf{x}^\top \boldsymbol{\xi}_i,$$

and $g(\mathbf{y}) = \gamma_1 \|\mathbf{y}\|_1$ according to (5.7.5).

The problem in (5.7.6) will be solved by stochastic ADMM algorithms in the experiments.

Measures for Convergence

Recall the constrained optimization problem as formalized in (5.1.1). In the evaluation of the algorithm performance on the convergence effect, we use the measure of objective gap defined as

$$[f(\widetilde{\mathbf{x}}^{(s)}) + g(\widetilde{\mathbf{y}}^{(s)})] - [f(\mathbf{x}^*) + g(\mathbf{y}^*)].$$

We also evaluate the convergence of algorithms using the measure of gap function Q defined in (5.5.1):

$$Q(\widetilde{\mathbf{x}}^{(s)}, \widetilde{\mathbf{y}}^{(s)}) = \left[f(\widetilde{\mathbf{x}}^{(s)}) + g(\widetilde{\mathbf{y}}^{(s)}) + \rho\langle \mathbf{A}\widetilde{\mathbf{x}}^{(s)} + \mathbf{B}\widetilde{\mathbf{y}}^{(s)} - \mathbf{c}, \mathbf{z}^* \rangle\right] - \left[f(\mathbf{x}^*) + g(\mathbf{y}^*) + \rho\langle \mathbf{A}\mathbf{x}^* + \mathbf{B}\mathbf{y}^* - \mathbf{c}, \mathbf{z}^* \rangle\right].$$

The convergence effect of different algorithms is compared based on the same number of entire data passes [185]. One entire data pass is the least possible iterations for passing through the entire data instances with respect to all coordinates. We also report the measures for convergence of different algorithms with respect to the same training time.

5.7.3 Real Data Sets

The empirical studies are conducted on the following three real data sets:

- ***20news***: A small version of the 20newsgroups data [85]. Each data instance is represented by the binary occurrence for 100 words across 10,066 postings. For the binary classification problem two class labels *comp* and *talk* are considered.

- ***a9a***: UCI data set for predicting whether income exceeds \$50K/year based on census data [77]. The data set can be downloaded via the LIBSVM software [24].
- ***ijcnn1***: IJCNN 2001 neural network competition data set [127].

Summary statistics of these data sets are provided in Table 5.1. The *20news* data set can be downloaded from the Web[2]. Both *a9a* and *ijcnn1* data sets can be downloaded via the LIBSVM software [24].

5.7.4 Algorithms for Comparison

We evaluate the convergence performance of SVR-ADMM-s and SVR-ADMM-D in comparison with the baseline stochastic ADMM algorithms. Below are the three algorithms for comparison.

- **STOC-ADMM (SG)**: The stochastic ADMM algorithm proposed by Ouyang *et al.* [124]. This algorithm has a sublinear rate of convergence for both strongly and non-strongly convex objectives.
- **SA-ADMM (SA)**: Stochastic average ADMM proposed by Zhong and Kwok [193]. This algorithm integrates the stochastic average gradient [134] into the design of stochastic ADMM algorithms. SA-ADMM converges at a sublinear rate for non-strongly convex objectives. It remains unknown whether its theoretical convergence can be improved for strongly convex objectives.
- **SDCA-ADMM (SD)**: Stochastic dual coordinate ascent for ADMM proposed by Suzuki [154]. This algorithm combines the stochastic dual coordinate ascent [140] and ADMM. SDCA-ADMM converges to the optimum at a linear rate for strongly convex objectives. Different from the other algorithms in comparison, SDCA-ADMM has to employ a dual formulation to solve the problem (5.7.3).
- **SVR-ADMM-D (D)**: The proposed SVR-ADMM-D algorithm for non-strongly convex objectives.

5.7.5 Experimental Setting

We describe the configuration of the experimental equipment and the procedure for setting parameters of the proposed algorithm.

Equipment Configuration

For the evaluation of the convergence effect with respect to training time, the experiments are conducted on a computer with an 8-core 3.4GHz CPU and a 32GB RAM.

[2] `http://cs.nyu.edu/~roweis/data.html`

Parameter Setting

The given matrix $\mathbf{P}$ is fixed as the identity matrix $\mathbf{I}$ in SVR-ADMM-D. For each algorithm, its parameters are chosen around the theoretical values to give the fastest convergence with the grid search under the five-fold cross validation. Here we describe the details as follows. First, the training data set is divided into five subsets of approximately the same size. One validation takes five trials on different subsets: in each trial, one subset is left out as the testing data set and the remaining four subsets are used for training. The convergence effect in one cross-validation is estimated by the averaged performance of the five trials.

Taking the *ijcnn1* data set as an example, for SVR-ADMM-s, the parameters are set as $\eta = 5 \times 10^{-3}$, $\rho = 10^{-4}$, and $m = n$; for SVR-ADMM-D, the parameters are set as $\eta = 5 \times 10^{-3}$, $\rho = 10^{-4}$, and $m^{(0)} = 1$.

5.7.6 Experimental Results

All the experimental results are obtained from 30 replications. For clarity of exposition, Figures 5.1 plots the mean value of the results from all these replications.

For non-strongly objectives, Figure 5.1 compares the convergence performance of SVR-ADMM-D with that of STOC-ADMM, SA-ADMM, and SDCA-ADMM on three data sets *20news*, *a9a*, and *ijcnn1* as described in Section 5.7.3.

The measures of objective gap and gap function Q (defined in Section 5.7.2) of different algorithms are depicted for the same number of entire data passes in Figures 5.1(a)—5.1(f) and for the same training time in Figures 5.1(g)— 5.1(l). Since SDCA-ADMM employs a dual formulation, the measure of gap function Q does not apply to it.

In general, among all the four algorithms in comparison for non-strongly convex objectives, SVR-ADMM-D converges fastest for the same number of entire data passes or for the same training time. Both measures of objective gap and gap function Q agree with each other when different algorithms are compared on the same data set for the same number of entire data passes or training time.

It is also observed that, although SDCA-ADMM generally converges faster than SA-ADMM for the same number of entire data passes in Figures 5.1(a)—5.1(c), for the same training time in Figures 5.1(g)—5.1(i) SDCA-ADMM generally converges slower than SA-ADMM. This is because in SDCA-ADMM, the per-iteration variable update may not have a closed-form solution due to its employed dual formulation. Such an added per-iteration complexity makes SDCA-ADMM generally converge slower than SA-ADMM when training time is considered.

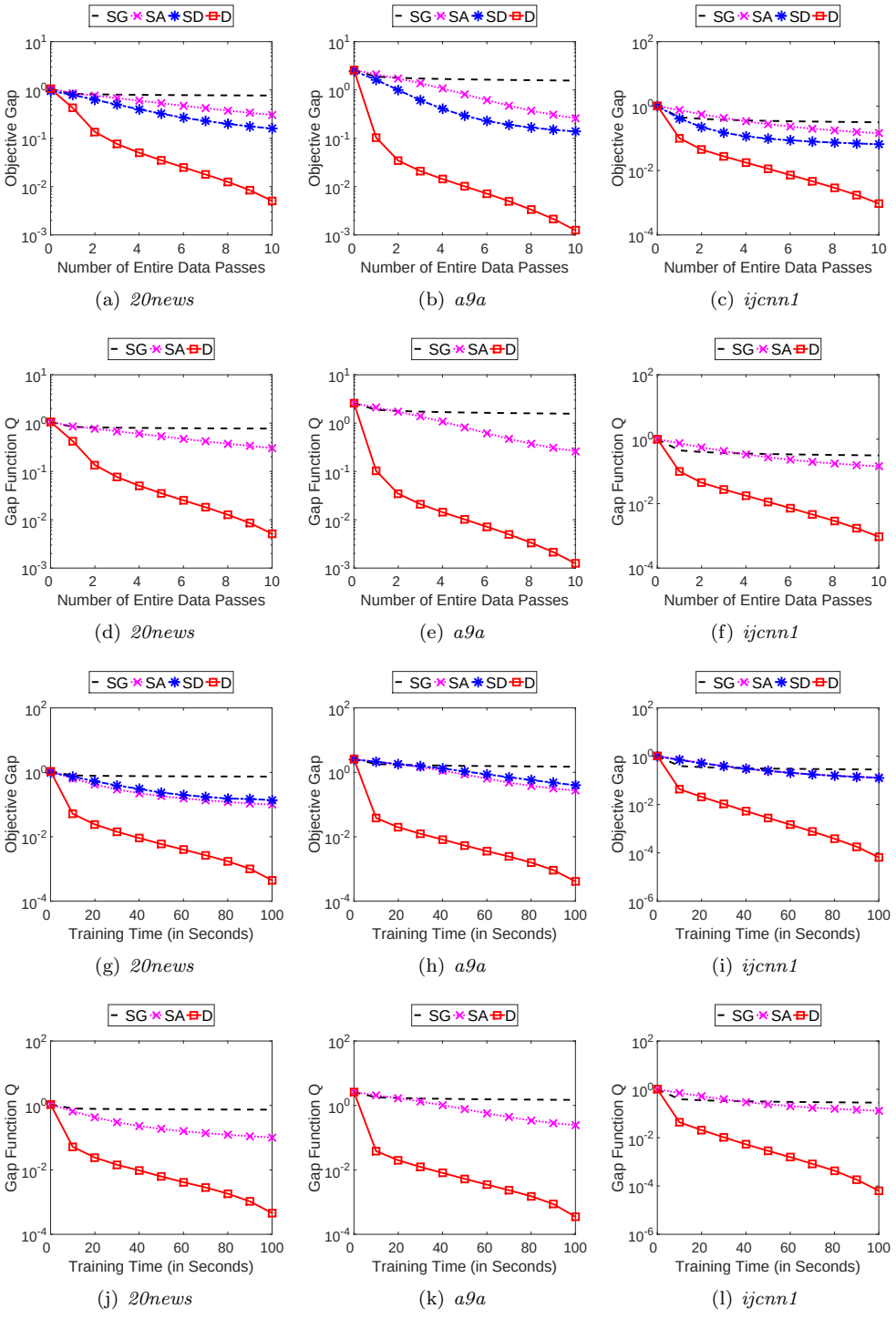


Figure 5.1: Convergence comparison of algorithms for the non-strongly convex objective problem on three data sets. In general, SVR-ADMM-D (D) converges fastest to the optimum for the same number of entire data passes (top 2 rows) or for the same training time (bottom 2 rows).

5.8 Conclusion

We focused on stochastic ADMM algorithms due to their significance in scenarios where lighter per-iteration computational costs are preferred. We proposed a variance reduced algorithm for stochastic alternating direction methods of multipliers: SVR-ADMM-D for non-strongly convex objectives. The proposed algorithm leverages past variable values to progressively reduce the variance of the gradient estimator. The algorithm also incorporates the doubling-trick to enable itself to be a theoretically-sound anytime algorithm: it can be interrupted anytime while the training error converges to zero with increasing iterations.

The SVR-ADMM-D algorithm developed in this chapter can be useful in many constrained optimization problems. For instance, one cornerstone class of problems in machine learning are regularized empirical risk minimizations. The empirical evaluation with graph-guided fused lasso on three real data sets supported our theory. The experimental results also revealed that SVR-ADMM-D converges faster than various baseline stochastic ADMM algorithms.

5.9 Proof of Lemmas

This section provides proof of all the lemmas in Chapter 5.

5.9.1 Proof of Lemma 5.6.3

We reproduce the SVR-ADMM-D x-minimization step (Line 12 of Algorithm 2), and re-arrange the terms as follows.

$$\begin{aligned}\mathbf{x}_t^{(s)} &= \operatorname*{argmin}_{\mathbf{x}} \left\langle \mathbf{h}_{t-1}^{(s)}, \mathbf{x} \right\rangle + \frac{\rho}{2}\left\|\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{y}_t^{(s)} - \mathbf{c} + \mathbf{z}_{t-1}^{(s)}\right\|^2 + \frac{1}{2\eta}\left\|\mathbf{x} - \mathbf{x}_{t-1}^{(s)}\right\|_{\mathbf{P}}^2 \\ &= \operatorname*{argmin}_{\mathbf{x}} \left[\left\langle \mathbf{h}_{t-1}^{(s)}, \mathbf{x} \right\rangle + \frac{1}{2\eta}\left\|\mathbf{x} - \mathbf{x}_{t-1}^{(s)}\right\|^2\right] + \frac{1}{2\eta}\left\|\mathbf{x} - \mathbf{x}_{t-1}^{(s)}\right\|_{\mathbf{P}-\mathbf{I}}^2 \\ &\quad + \frac{\rho}{2}\left\|\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{y}_t^{(s)} - \mathbf{c} + \mathbf{z}_{t-1}^{(s)}\right\|^2. \end{aligned} \tag{5.9.1}$$

By modifying the terms in the brackets above, (5.9.1) can be re-written as

$$\mathbf{x}_t^{(s)} = \operatorname*{argmin}_{\mathbf{x}} \frac{1}{2\eta}\left\|\mathbf{x}_{t-1}^{(s)} - \eta\mathbf{h}_{t-1}^{(s)} - \mathbf{x}\right\|^2 + \frac{1}{2\eta}\left\|\mathbf{x} - \mathbf{x}_{t-1}^{(s)}\right\|_{\mathbf{P}-\mathbf{I}}^2 + \frac{\rho}{2}\left\|\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{y}_t^{(s)} - \mathbf{c} + \mathbf{z}_{t-1}^{(s)}\right\|^2 \tag{5.9.2}$$

because after taking derivatives with respect to $\mathbf{x}$, (5.9.1) and (5.9.2) have the same analytical solution for $\mathbf{x}_t^{(s)}$ satisfying

$$\begin{aligned}&\mathbf{h}_{t-1}^{(s)} + \frac{1}{\eta}\big(\mathbf{x}_t^{(s)} - \mathbf{x}_{t-1}^{(s)}\big) + \frac{1}{\eta}(\mathbf{P}-\mathbf{I})\big(\mathbf{x}_t^{(s)} - \mathbf{x}_{t-1}^{(s)}\big) + \rho\mathbf{A}^\top\big(\mathbf{A}\mathbf{x}_t^{(s)} + \mathbf{B}\mathbf{y}_t^{(s)} - \mathbf{c} + \mathbf{z}_{t-1}^{(s)}\big)\\ &= \mathbf{h}_{t-1}^{(s)} + \frac{1}{\eta}\mathbf{P}\big(\mathbf{x}_t^{(s)} - \mathbf{x}_{t-1}^{(s)}\big) + \rho\mathbf{A}^\top\big(\mathbf{A}\mathbf{x}_t^{(s)} + \mathbf{B}\mathbf{y}_t^{(s)} - \mathbf{c} + \mathbf{z}_{t-1}^{(s)}\big) = 0.\end{aligned}$$

We complete the proof with (5.9.2) and the proximal operator defined in (5.6.1).

5.9.2 Proof of Lemma 5.6.4

Referring to Line 11 of Algorithm 2, we have

$$\begin{aligned}\mathbb{E}\big[\mathbf{h}_{t-1}^{(s)}\big] &= \mathbb{E}\big[\nabla f_i(\mathbf{x}_{t-1}^{(s)}) - \nabla f_i(\widetilde{\mathbf{x}}^{(s-1)}) + \nabla f(\widetilde{\mathbf{x}}^{(s-1)})\big]\\ &= \frac{1}{n}\sum_{i=1}^n \nabla f_i(\mathbf{x}_{t-1}^{(s)}) - \mathbb{E}\big[\nabla f_i(\widetilde{\mathbf{x}}^{(s-1)})\big] + \mathbb{E}\Big[\mathbb{E}\big[\nabla f_i(\widetilde{\mathbf{x}}^{(s-1)})\big]\Big]\\ &= \nabla f(\mathbf{x}_{t-1}^{(s)}),\end{aligned}$$

where the second equality is obtained by $\nabla f(\widetilde{\mathbf{x}}^{(s-1)}) = (1/n)\sum_{i=1}^n f_i(\widetilde{\mathbf{x}}^{(s-1)}) = \mathbb{E}\big[\nabla f_i(\widetilde{\mathbf{x}}^{(s-1)})\big]$.

5.9.3 Proof of Lemma 5.6.5

Using variance decomposition $\mathbb{E}\Big[\big\|\mathbf{x} - \mathbb{E}[\mathbf{x}]\big\|^2\Big] = \mathbb{E}\big[\|\mathbf{x}\|^2\big] - \big\|\mathbb{E}[\mathbf{x}]\big\|^2$ for all $\mathbf{x}$, we have

$$\begin{aligned}\mathbb{E}\big[\big\|\mathbf{h}_{t-1}^{(s)} - \nabla f(\mathbf{x}_{t-1}^{(s)})\big\|^2\big] &= \mathbb{E}\Big[\Big\|\big[\nabla f_i(\mathbf{x}_{t-1}^{(s)}) - \nabla f_i(\widetilde{\mathbf{x}}^{(s-1)})\big] - \big[\nabla f(\mathbf{x}_{t-1}^{(s)}) - \nabla f(\widetilde{\mathbf{x}}^{(s-1)})\big]\Big\|^2\Big]\\ &= \mathbb{E}\big[\big\|\nabla f_i(\mathbf{x}_{t-1}^{(s)}) - \nabla f_i(\widetilde{\mathbf{x}}^{(s-1)})\big\|^2\big] - \big\|\nabla f(\mathbf{x}_{t-1}^{(s)}) - \nabla f(\widetilde{\mathbf{x}}^{(s-1)})\big\|^2\\ &\le \mathbb{E}\big[\big\|\nabla f_i(\mathbf{x}_{t-1}^{(s)}) - \nabla f_i(\widetilde{\mathbf{x}}^{(s-1)})\big\|^2\big].\end{aligned} \tag{5.9.3}$$

By the property $\|\mathbf{x}+\mathbf{y}\|^2 \le 2\|\mathbf{x}\| + 2\|\mathbf{y}\|^2$ for all $\mathbf{x}$ and $\mathbf{y}$, we obtain

$$\begin{aligned}\mathbb{E}\big[\big\|\nabla f_i(\mathbf{x}_{t-1}^{(s)}) - \nabla f_i(\widetilde{\mathbf{x}}^{(s-1)})\big\|^2\big] &= \mathbb{E}\Big[\Big\|\big[\nabla f_i(\mathbf{x}_{t-1}^{(s)}) - \nabla f_i(\mathbf{x}^*)\big] - \big[\nabla f_i(\widetilde{\mathbf{x}}^{(s-1)}) - \nabla f_i(\mathbf{x}^*)\big]\Big\|^2\Big]\\ &\le 2\mathbb{E}\big[\big\|\nabla f_i(\mathbf{x}_{t-1}^{(s)}) - \nabla f_i(\mathbf{x}^*)\big\|^2\big] + 2\mathbb{E}\big[\big\|\nabla f_i(\widetilde{\mathbf{x}}^{(s-1)}) - \nabla f_i(\mathbf{x}^*)\big\|^2\big].\end{aligned} \tag{5.9.4}$$

With the component gap function $Q_f(\mathbf{x})$ defined in (5.5.2), now we can bound the variance of the gradient estimator $\mathbf{h}_{t-1}^{(s)}$ by combining (5.9.3) and (5.9.4) as

$$\begin{aligned}
&\mathbb{E}\big[\big\|\mathbf{h}_{t-1}^{(s)} - \nabla f(\mathbf{x}_{t-1}^{(s)})\big\|^2\big] \\
&\leq \frac{2}{n}\sum_{i=1}^{n}\big\|\nabla f_i(\mathbf{x}_{t-1}^{(s)}) - \nabla f_i(\mathbf{x}^*)\big\|^2 + \frac{2}{n}\sum_{i=1}^{n}\big\|\nabla f_i(\widetilde{\mathbf{x}}^{(s-1)}) - \nabla f_i(\mathbf{x}^*)\big\|^2 \\
&\leq \frac{4L_M}{n}\sum_{i=1}^{n}\Big[f_i(\mathbf{x}_{t-1}^{(s)}) - f_i(\mathbf{x}^*) - \big\langle\nabla f_i(\mathbf{x}^*), \mathbf{x}_{t-1}^{(s)} - \mathbf{x}^*\big\rangle + f_i(\widetilde{\mathbf{x}}^{(s-1)}) - f_i(\mathbf{x}^*) - \big\langle\nabla f_i(\mathbf{x}^*), \widetilde{\mathbf{x}}^{(s-1)} - \mathbf{x}^*\big\rangle\Big] \\
&= 4L_M\big[Q_f(\mathbf{x}_{t-1}^{(s)}) + Q_f(\widetilde{\mathbf{x}}^{(s-1)})\big],
\end{aligned}$$

where $L_M = \max_i L_i$ and the second inequality is obtained by the property $f_i(\mathbf{x}) + \langle\nabla f_i(\mathbf{x}), \mathbf{y} - \mathbf{x}\rangle + \|\nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y})\|^2/(2L_i) \leq f_i(\mathbf{y})$ for all $\mathbf{x}$ and $\mathbf{y}$ under Assumptions 5.2.1 and 5.2.2 [119, Theorem 2.1.5].

5.9.4 Proof of Lemma 5.6.6

For all $\mathbf{x}$, with the proximal operator defined in (5.6.1) we have

$$\begin{aligned}
&\mathbb{E}\big[\big\langle\mathbf{h}_{t-1}^{(s)} - \nabla f(\mathbf{x}_{t-1}^{(s)}), \mathbf{x} - \mathbf{x}_t^{(s)}\big\rangle\big] \\
&= \mathbb{E}\Big[\Big\langle\mathbf{h}_{t-1}^{(s)} - \nabla f(\mathbf{x}_{t-1}^{(s)}), \mathbf{x} - \operatorname{prox}_\eta\big(\mathbf{x}_{t-1}^{(s)} - \eta\nabla f(\mathbf{x}_{t-1}^{(s)})\big)\Big\rangle\Big] \\
&\qquad + \mathbb{E}\Big[\Big\langle\mathbf{h}_{t-1}^{(s)} - \nabla f(\mathbf{x}_{t-1}^{(s)}), \operatorname{prox}_\eta\big(\mathbf{x}_{t-1}^{(s)} - \eta\nabla f(\mathbf{x}_{t-1}^{(s)})\big) - \mathbf{x}_t^{(s)}\Big\rangle\Big] \\
&= \mathbb{E}\Big[\Big\langle\mathbf{h}_{t-1}^{(s)} - \nabla f(\mathbf{x}_{t-1}^{(s)}), \operatorname{prox}_\eta\big(\mathbf{x}_{t-1}^{(s)} - \eta\nabla f(\mathbf{x}_{t-1}^{(s)})\big) - \mathbf{x}_t^{(s)}\Big\rangle\Big],
\end{aligned} \tag{5.9.5}$$

where the second equality is obtained by using Lemma 5.6.4. By Cauchy-Schwartz inequality, Lemma 5.6.3, and the non-expansiveness of the proximal operator (Remark 5.6.2), we obtain

$$\begin{aligned}
&\Big\langle\mathbf{h}_{t-1}^{(s)} - \nabla f(\mathbf{x}_{t-1}^{(s)}), \operatorname{prox}_\eta\big(\mathbf{x}_{t-1}^{(s)} - \eta\nabla f(\mathbf{x}_{t-1}^{(s)})\big) - \mathbf{x}_t^{(s)}\Big\rangle \\
&\leq \big\|\mathbf{h}_{t-1}^{(s)} - \nabla f(\mathbf{x}_{t-1}^{(s)})\big\| \cdot \Big\|\operatorname{prox}_\eta\big(\mathbf{x}_{t-1}^{(s)} - \eta\nabla f(\mathbf{x}_{t-1}^{(s)})\big) - \operatorname{prox}_\eta\big(\mathbf{x}_{t-1}^{(s)} - \eta\mathbf{h}_{t-1}^{(s)}\big)\Big\| \\
&\leq \big\|\mathbf{h}_{t-1}^{(s)} - \nabla f(\mathbf{x}_{t-1}^{(s)})\big\| \cdot \Big\|\big[\mathbf{x}_{t-1}^{(s)} - \eta\nabla f(\mathbf{x}_{t-1}^{(s)})\big] - \big[\mathbf{x}_{t-1}^{(s)} - \eta\mathbf{h}_{t-1}^{(s)}\big]\Big\| \\
&= \eta\big\|\mathbf{h}_{t-1}^{(s)} - \nabla f(\mathbf{x}_{t-1}^{(s)})\big\|^2.
\end{aligned} \tag{5.9.6}$$

Combining the results of (5.9.5) and (5.9.6) gives

$$\mathbb{E}\big[\big\langle\mathbf{h}_{t-1}^{(s)} - \nabla f(\mathbf{x}_{t-1}^{(s)}), \mathbf{x} - \mathbf{x}_t^{(s)}\big\rangle\big] \leq \eta\mathbb{E}\big[\big\|\mathbf{h}_{t-1}^{(s)} - \nabla f(\mathbf{x}_{t-1}^{(s)})\big\|^2\big].$$

5.9.5 Proof of Lemma 5.6.7

We reproduce the SVR-ADMM-D x-minimization step (Line 12 of Algorithm 2):

$$\mathbf{x}_t^{(s)} = \operatorname*{argmin}_{\mathbf{x}} \langle \mathbf{h}_{t-1}^{(s)}, \mathbf{x} \rangle + \frac{\rho}{2} \|\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{y}_t^{(s)} - \mathbf{c} + \mathbf{z}_{t-1}^{(s)}\|^2 + \frac{1}{2\eta} \|\mathbf{x} - \mathbf{x}_{t-1}^{(s)}\|_{\mathbf{P}}^2. \tag{5.9.7}$$

By taking derivatives of the right-hand side of (5.9.7) with respect to $\mathbf{x}$, the analytical solution for $\mathbf{x}_t^{(s)}$ satisfies (5.7.1).

Reproducing the SVR-ADMM-D dual update step (Line 13 of Algorithm 2), we have

$$\mathbf{z}_t^{(s)} = \mathbf{z}_{t-1}^{(s)} + \mathbf{A}\mathbf{x}_t^{(s)} + \mathbf{B}\mathbf{y}_t^{(s)} - \mathbf{c}. \tag{5.9.8}$$

After combining the results of (5.7.1) and (5.9.8), we can obtain the relationship between consecutive steps of the primal variable $\mathbf{x}$ in the same stage:

$$\mathbf{x}_t^{(s)} - \mathbf{x}_{t-1}^{(s)} = -\eta \mathbf{P}^{-1} \big(\mathbf{h}_{t-1}^{(s)} + \rho \mathbf{A}^\top \mathbf{z}_t^{(s)}\big). \tag{5.9.9}$$

Using the consecutive step relationship in (5.9.9), for all $\mathbf{x}$ we have

$$\begin{aligned}
&\|\mathbf{x} - \mathbf{x}_t^{(s)}\|_{\mathbf{P}}^2 \\
&= \Big\|\mathbf{x} - \mathbf{x}_{t-1}^{(s)} + \eta \mathbf{P}^{-1} \big(\mathbf{h}_{t-1}^{(s)} + \rho \mathbf{A}^\top \mathbf{z}_t^{(s)}\big)\Big\|_{\mathbf{P}}^2 \\
&= \Big\langle \mathbf{x} - \mathbf{x}_{t-1}^{(s)}, \mathbf{P}\big(\mathbf{x} - \mathbf{x}_{t-1}^{(s)}\big)\Big\rangle + \Big\langle \eta \mathbf{P}^{-1} \big(\mathbf{h}_{t-1}^{(s)} + \rho \mathbf{A}^\top \mathbf{z}_t^{(s)}\big), \\
&\qquad \eta \mathbf{P}\mathbf{P}^{-1} \big(\mathbf{h}_{t-1}^{(s)} + \rho \mathbf{A}^\top \mathbf{z}_t^{(s)}\big)\Big\rangle + 2\Big\langle \mathbf{x} - \mathbf{x}_{t-1}^{(s)}, \eta \mathbf{P}\mathbf{P}^{-1} \big(\mathbf{h}_{t-1}^{(s)} + \rho \mathbf{A}^\top \mathbf{z}_t^{(s)}\big)\Big\rangle \\
&= \|\mathbf{x} - \mathbf{x}_{t-1}^{(s)}\|_{\mathbf{P}}^2 + \eta \Big(\eta \|\mathbf{h}_{t-1}^{(s)} + \rho \mathbf{A}^\top \mathbf{z}_t^{(s)}\|_{\mathbf{P}^{-1}}^2 + 2\big\langle \mathbf{x} - \mathbf{x}_{t-1}^{(s)}, \mathbf{h}_{t-1}^{(s)} + \rho \mathbf{A}^\top \mathbf{z}_t^{(s)}\big\rangle\Big).
\end{aligned} \tag{5.9.10}$$

Suppose that $0 < \eta < 1/L$. We can split the first term in the parentheses of the last equality of (5.9.10):

$$\eta \|\mathbf{h}_{t-1}^{(s)} + \rho \mathbf{A}^\top \mathbf{z}_t^{(s)}\|_{\mathbf{P}^{-1}}^2 \leq 2\eta \|\mathbf{h}_{t-1}^{(s)} + \rho \mathbf{A}^\top \mathbf{z}_t^{(s)}\|_{\mathbf{P}^{-1}}^2 - L\eta^2 \|\mathbf{h}_{t-1}^{(s)} + \rho \mathbf{A}^\top \mathbf{z}_t^{(s)}\|_{\mathbf{P}^{-1}}^2. \tag{5.9.11}$$

Hence, by combining the results of (5.9.10) and (5.9.11), we have

$$\begin{aligned}
\|\mathbf{x} - \mathbf{x}_t^{(s)}\|_{\mathbf{P}}^2 - \|\mathbf{x} - \mathbf{x}_{t-1}^{(s)}\|_{\mathbf{P}}^2 \leq \eta \Big(&2\eta \|\mathbf{h}_{t-1}^{(s)} + \rho \mathbf{A}^\top \mathbf{z}_t^{(s)}\|_{\mathbf{P}^{-1}}^2 - L\eta^2 \|\mathbf{h}_{t-1}^{(s)} + \rho \mathbf{A}^\top \mathbf{z}_t^{(s)}\|_{\mathbf{P}^{-1}}^2 \\
&+ 2\big\langle \mathbf{x} - \mathbf{x}_{t-1}^{(s)}, \mathbf{h}_{t-1}^{(s)} + \rho \mathbf{A}^\top \mathbf{z}_t^{(s)}\big\rangle\Big).
\end{aligned} \tag{5.9.12}$$

According to the definition of $\mathbf{P}$-quadratic norm,

$$\big\|\mathbf{h}_{t-1}^{(s)}+\rho\mathbf{A}^\top\mathbf{z}_t^{(s)}\big\|_{\mathbf{P}^{-1}}^2=\Big\langle\mathbf{h}_{t-1}^{(s)}+\rho\mathbf{A}^\top\mathbf{z}_t^{(s)},\mathbf{P}^{-1}\big(\mathbf{h}_{t-1}^{(s)}+\rho\mathbf{A}^\top\mathbf{z}_t^{(s)}\big)\Big\rangle. \tag{5.9.13}$$

Re-arranging terms in (5.9.13), we have

$$\begin{aligned}2\eta\big\|\mathbf{h}_{t-1}^{(s)}+\rho\mathbf{A}^\top\mathbf{z}_t^{(s)}\big\|_{\mathbf{P}^{-1}}^2&=2\Big\langle\mathbf{h}_{t-1}^{(s)}+\rho\mathbf{A}^\top\mathbf{z}_t^{(s)},\eta\mathbf{P}^{-1}\big(\mathbf{h}_{t-1}^{(s)}+\rho\mathbf{A}^\top\mathbf{z}_t^{(s)}\big)\Big\rangle\\&=2\big\langle\mathbf{h}_{t-1}^{(s)}+\rho\mathbf{A}^\top\mathbf{z}_t^{(s)},\mathbf{x}_{t-1}^{(s)}-\mathbf{x}_t^{(s)}\big\rangle,\end{aligned} \tag{5.9.14}$$

where the last equality is obtained by leveraging the relationship between consecutive steps of the primal variable $\mathbf{x}$ in (5.9.9). We can re-arrange terms in (5.9.13) in another form as below.

$$\begin{aligned}-L\eta^2\big\|\mathbf{h}_{t-1}^{(s)}+\rho\mathbf{A}^\top\mathbf{z}_t^{(s)}\big\|_{\mathbf{P}^{-1}}^2&=-L\Big\langle\eta\mathbf{P}^{-1}\big(\mathbf{h}_{t-1}^{(s)}+\rho\mathbf{A}^\top\mathbf{z}_t^{(s)}\big),\eta\mathbf{P}\mathbf{P}^{-1}\big(\mathbf{h}_{t-1}^{(s)}+\rho\mathbf{A}^\top\mathbf{z}_t^{(s)}\big)\Big\rangle\\&=-L\Big\langle\mathbf{x}_{t-1}^{(s)}-\mathbf{x}_t^{(s)},\mathbf{P}\big(\mathbf{x}_{t-1}^{(s)}-\mathbf{x}_t^{(s)}\big)\Big\rangle\\&=-L\big\|\mathbf{x}_t^{(s)}-\mathbf{x}_{t-1}^{(s)}\big\|_{\mathbf{P}}^2,\end{aligned} \tag{5.9.15}$$

where the second equality is obtained by re-using the relationship between consecutive steps of the primal variable $\mathbf{x}$ in (5.9.9).

Replacing the split terms in (5.9.12) by the results of (5.9.14) and (5.9.15), we obtain

$$\begin{aligned}&\big\|\mathbf{x}-\mathbf{x}_t^{(s)}\big\|_{\mathbf{P}}^2-\big\|\mathbf{x}-\mathbf{x}_{t-1}^{(s)}\big\|_{\mathbf{P}}^2\\&\leq\eta\Big(2\big\langle\mathbf{h}_{t-1}^{(s)}+\rho\mathbf{A}^\top\mathbf{z}_t^{(s)},\mathbf{x}_{t-1}^{(s)}-\mathbf{x}_t^{(s)}\big\rangle-L\big\|\mathbf{x}_t^{(s)}-\mathbf{x}_{t-1}^{(s)}\big\|_{\mathbf{P}}^2+2\big\langle\mathbf{x}-\mathbf{x}_{t-1}^{(s)},\mathbf{h}_{t-1}^{(s)}+\rho\mathbf{A}^\top\mathbf{z}_t^{(s)}\big\rangle\Big)\\&=\eta\Big(2\big\langle\mathbf{h}_{t-1}^{(s)}+\rho\mathbf{A}^\top\mathbf{z}_t^{(s)},\mathbf{x}-\mathbf{x}_t^{(s)}\big\rangle-L\big\|\mathbf{x}_t^{(s)}-\mathbf{x}_{t-1}^{(s)}\big\|_{\mathbf{P}}^2\Big)\\&\leq\eta\Big(2\big\langle\mathbf{h}_{t-1}^{(s)}+\rho\mathbf{A}^\top\mathbf{z}_t^{(s)},\mathbf{x}-\mathbf{x}_t^{(s)}\big\rangle-L\big\|\mathbf{x}_t^{(s)}-\mathbf{x}_{t-1}^{(s)}\big\|^2\Big),\end{aligned}$$

where the last inequality is obtained by the input property $\mathbf{P} \succeq \mathbf{I}$. By exploiting the Lipschitz continuity of the gradient and the convexity for the function $f(\mathbf{x})$, we complete the proof with

$$
\begin{aligned}
&\big\|\mathbf{x}-\mathbf{x}_t^{(s)}\big\|_{\mathbf{P}}^2-\big\|\mathbf{x}-\mathbf{x}_{t-1}^{(s)}\big\|_{\mathbf{P}}^2\\
&\leq \eta\Big[2\big\langle \mathbf{h}_{t-1}^{(s)}+\rho\mathbf{A}^\top\mathbf{z}_t^{(s)},\mathbf{x}-\mathbf{x}_t^{(s)}\big\rangle-2f(\mathbf{x}_t^{(s)})+2f(\mathbf{x}_{t-1}^{(s)})+2\big\langle\nabla f(\mathbf{x}_{t-1}^{(s)}),\mathbf{x}_t^{(s)}-\mathbf{x}_{t-1}^{(s)}\big\rangle\Big]\\
&\leq \eta\Big[2\big\langle \mathbf{h}_{t-1}^{(s)}+\rho\mathbf{A}^\top\mathbf{z}_t^{(s)},\mathbf{x}-\mathbf{x}_t^{(s)}\big\rangle-2f(\mathbf{x}_t^{(s)})+2f(\mathbf{x})-2\big\langle\nabla f(\mathbf{x}_{t-1}^{(s)}),\mathbf{x}-\mathbf{x}_{t-1}^{(s)}\big\rangle+2\langle\nabla f(\mathbf{x}_{t-1}^{(s)}),\mathbf{x}_t^{(s)}-\mathbf{x}_{t-1}^{(s)}\big\rangle\Big]\\
&= 2\eta\Big[\big\langle \mathbf{h}_{t-1}^{(s)}+\rho\mathbf{A}^\top\mathbf{z}_t^{(s)}-\nabla f(\mathbf{x}_{t-1}^{(s)}),\mathbf{x}-\mathbf{x}_t^{(s)}\big\rangle+f(\mathbf{x})-f(\mathbf{x}_t^{(s)})\Big],
\end{aligned}
$$

where the first inequality is obtained by the property $f(\mathbf{y})-f(\mathbf{x})-\langle\nabla f(\mathbf{x}),\mathbf{y}-\mathbf{x}\rangle\leq(L/2)\|\mathbf{x}-\mathbf{y}\|^2$ for all $\mathbf{x}$ and $\mathbf{y}$ under Assumptions 5.2.1 and 5.2.2 [119, Theorem 2.1.5], while the second inequality holds under Assumption 5.2.2.

5.9.6 Proof of Lemma 5.6.8

Set $0<\eta<1/(4L_M)$, where $L_M=\max_i L_i$. Recall that $L_M\geq L$, as described in Assumption 5.2.1. We can apply Lemma 5.6.7, and combine its results with those of Lemma 5.6.5 and Lemma 5.6.6:

$$
\begin{aligned}
&\mathbb{E}\big[\big\|\mathbf{x}_t^{(s)}-\mathbf{x}^*\big\|_{\mathbf{P}}^2-\big\|\mathbf{x}_{t-1}^{(s)}-\mathbf{x}^*\big\|_{\mathbf{P}}^2\big]\\
&\leq 2\eta\Big[\mathbb{E}\big[\big\langle \mathbf{h}_{t-1}^{(s)}-\nabla f(\mathbf{x}_{t-1}^{(s)}),\mathbf{x}^*-\mathbf{x}_t^{(s)}\big\rangle\big]+\mathbb{E}\big[\big\langle\rho\mathbf{A}^\top\mathbf{z}_t^{(s)},\mathbf{x}^*-\mathbf{x}_t^{(s)}\big\rangle\big]+f(\mathbf{x}^*)-\mathbb{E}\big[f(\mathbf{x}_t^{(s)})\big]\Big]\\
&\leq 2\eta\Big[\eta\mathbb{E}\big[\big\|\mathbf{h}_{t-1}^{(s)}-\nabla f(\mathbf{x}_{t-1}^{(s)})\big\|^2\big]+\mathbb{E}\big[\big\langle\rho\mathbf{A}^\top\mathbf{z}_t^{(s)},\mathbf{x}^*-\mathbf{x}_t^{(s)}\big\rangle\big]+f(\mathbf{x}^*)-\mathbb{E}\big[f(\mathbf{x}_t^{(s)})\big]\Big]\\
&\leq 2\eta\Big[4L_M\eta\big[Q_f(\mathbf{x}_{t-1}^{(s)})+Q_f(\widetilde{\mathbf{x}}^{(s-1)})\big]+\mathbb{E}\big[\big\langle\rho\mathbf{A}^\top\mathbf{z}_t^{(s)},\mathbf{x}^*-\mathbf{x}_t^{(s)}\big\rangle\big]+f(\mathbf{x}^*)-\mathbb{E}\big[f(\mathbf{x}_t^{(s)})\big]\Big],
\end{aligned}
$$

where the three inequalities use the results of Lemma 5.6.7, Lemma 5.6.6, and Lemma 5.6.5 respectively, and $\mathbf{x}^*$ is the optimal value of the primal variable $\mathbf{x}$. By re-arranging terms, we have

$$
\begin{aligned}
\mathbb{E}\big[f(\mathbf{x}_t^{(s)})\big]-f(\mathbf{x}^*)\leq\frac{1}{2\eta}\mathbb{E}\big[\big\|\mathbf{x}_{t-1}^{(s)}-\mathbf{x}^*\big\|_{\mathbf{P}}^2-\big\|\mathbf{x}_t^{(s)}-\mathbf{x}^*\big\|_{\mathbf{P}}^2\big]+4L_M\eta Q_f(\mathbf{x}_{t-1}^{(s)})\\
+4L_M\eta Q_f(\widetilde{\mathbf{x}}^{(s-1)})+\mathbb{E}\big[\big\langle\rho\mathbf{A}^\top\mathbf{z}_t^{(s)},\mathbf{x}^*-\mathbf{x}_t^{(s)}\big\rangle\big].
\end{aligned}
$$

After appending the term $-\mathbb{E}\big[\big\langle \nabla f(\mathbf{x}^*), \mathbf{x}_t^{(s)} - \mathbf{x}^* \big\rangle\big]$ on both sides, with the component gap function $Q_f(\mathbf{x})$ defined in (5.5.2) we further have

$$
\begin{aligned}
&\mathbb{E}\big[Q_f(\mathbf{x}_t^{(s)})\big] \\
&\leq \frac{1}{2\eta}\mathbb{E}\big[\big\|\mathbf{x}_{t-1}^{(s)} - \mathbf{x}^*\big\|_{\mathbf{P}}^2 - \big\|\mathbf{x}_t^{(s)} - \mathbf{x}^*\big\|_{\mathbf{P}}^2\big] + 4L_M\eta Q_f(\mathbf{x}_{t-1}^{(s)}) + 4L_M\eta Q_f(\widetilde{\mathbf{x}}^{(s-1)}) \\
&\quad + \mathbb{E}\big[\big\langle \rho\mathbf{A}^\top \mathbf{z}_t^{(s)}, \mathbf{x}^* - \mathbf{x}_t^{(s)} \big\rangle\big] - \mathbb{E}\big[\big\langle \nabla f(\mathbf{x}^*), \mathbf{x}_t^{(s)} - \mathbf{x}^* \big\rangle\big] \\
&= \frac{1}{2\eta}\mathbb{E}\big[\big\|\mathbf{x}_{t-1}^{(s)} - \mathbf{x}^*\big\|_{\mathbf{P}}^2 - \big\|\mathbf{x}_t^{(s)} - \mathbf{x}^*\big\|_{\mathbf{P}}^2\big] + 4L_M\eta Q_f(\mathbf{x}_{t-1}^{(s)}) + 4L_M\eta Q_f(\widetilde{\mathbf{x}}^{(s-1)}) \\
&\quad + \mathbb{E}\big[\big\langle \rho\mathbf{A}^\top \mathbf{z}_t^{(s)}, \mathbf{x}^* - \mathbf{x}_t^{(s)} \big\rangle\big] + \mathbb{E}\big[\big\langle \rho\mathbf{A}^\top \mathbf{z}^*, \mathbf{x}_t^{(s)} - \mathbf{x}^* \big\rangle\big] \\
&= \frac{1}{2\eta}\mathbb{E}\big[\big\|\mathbf{x}_{t-1}^{(s)} - \mathbf{x}^*\big\|_{\mathbf{P}}^2 - \big\|\mathbf{x}_t^{(s)} - \mathbf{x}^*\big\|_{\mathbf{P}}^2\big] + 4L_M\eta Q_f(\mathbf{x}_{t-1}^{(s)}) + 4L_M\eta Q_f(\widetilde{\mathbf{x}}^{(s-1)}) \\
&\quad + \mathbb{E}\big[\big\langle \rho\mathbf{A}^\top (\mathbf{z}_t^{(s)} - \mathbf{z}^*), \mathbf{x}^* - \mathbf{x}_t^{(s)} \big\rangle\big],
\end{aligned}
$$

where the first equality is obtained from the optimality condition in (5.3.5) and $\mathbf{z}^*$ is the optimal value of the dual variable $\mathbf{z}$. By summing up the terms on both sides over t from 1 to $m^{(s)}$, we obtain

$$
\begin{aligned}
\sum_{t=1}^{m^{(s)}} \mathbb{E}\big[Q_f(\mathbf{x}_t^{(s)})\big] \leq \frac{1}{2\eta}\mathbb{E}\big[\big\|\mathbf{x}_0^{(s)} - \mathbf{x}^*\big\|_{\mathbf{P}}^2 - \big\|\mathbf{x}_{m^{(s)}}^{(s)} - \mathbf{x}^*\big\|_{\mathbf{P}}^2\big] \\
+ 4L_M\eta\Big[\sum_{t=1}^{m^{(s)}} \mathbb{E}\big[Q_f(\mathbf{x}_t^{(s)})\big] + Q_f(\mathbf{x}_0^{(s)}) - \mathbb{E}\big[Q_f(\mathbf{x}_{m^{(s)}}^{(s)})\big]\Big] \\
+ 4L_M m^{(s)}\eta Q_f(\widetilde{\mathbf{x}}^{(s-1)}) + \sum_{t=1}^{m^{(s)}} \mathbb{E}\big[\big\langle \rho\mathbf{A}^\top (\mathbf{z}_t^{(s)} - \mathbf{z}^*), \mathbf{x}^* - \mathbf{x}_t^{(s)} \big\rangle\big].
\end{aligned}
$$

A further re-arrangement of terms gives the following results:

$$
\begin{aligned}
(1 - 4L_M\eta)\sum_{t=1}^{m^{(s)}} \mathbb{E}\big[Q_f(\mathbf{x}_t^{(s)})\big] \leq \frac{1}{2\eta}\mathbb{E}\big[\big\|\mathbf{x}_0^{(s)} - \mathbf{x}^*\big\|_{\mathbf{P}}^2 - \big\|\mathbf{x}_{m^{(s)}}^{(s)} - \mathbf{x}^*\big\|_{\mathbf{P}}^2\big] + 4L_M\eta Q_f(\mathbf{x}_0^{(s)}) \\
- 4L_M\eta\mathbb{E}\big[Q_f(\mathbf{x}_{m^{(s)}}^{(s)})\big] + 4L_M m^{(s)}\eta Q_f(\widetilde{\mathbf{x}}^{(s-1)}) \\
+ \sum_{t=1}^{m^{(s)}} \mathbb{E}\big[\big\langle \rho\mathbf{A}^\top (\mathbf{z}_t^{(s)} - \mathbf{z}^*), \mathbf{x}^* - \mathbf{x}_t^{(s)} \big\rangle\big]. \qquad (5.9.16)
\end{aligned}
$$

Recall the primal variable update step in Line 15 of Algorithm 2 that $\widetilde{\mathbf{x}}^{(s)} = (1/m^{(s)}) \sum_{t=1}^{m^{(s)}} \mathbf{x}_t^{(s)}$. Based on the definition of the component gap function $Q_f(\mathbf{x})$ in (5.5.2), we obtain

$$\begin{aligned}
\mathbb{E}\big[Q_f(\widetilde{\mathbf{x}}^{(s)})\big] &\leq \mathbb{E}\Big[\frac{1}{m^{(s)}} \sum_{t=1}^{m^{(s)}} f(\mathbf{x}_t^{(s)}) - f(\mathbf{x}^*) - \big\langle \nabla f(\mathbf{x}^*), \widetilde{\mathbf{x}}^{(s)} - \mathbf{x}^* \big\rangle\Big] \\
&= \mathbb{E}\Big[\frac{1}{m^{(s)}} \sum_{t=1}^{m^{(s)}} f(\mathbf{x}_t^{(s)}) - f(\mathbf{x}^*) - \big\langle \nabla f(\mathbf{x}^*), \frac{1}{m^{(s)}} \sum_{t=1}^{m^{(s)}} \mathbf{x}_t^{(s)} - \mathbf{x}^* \big\rangle\Big] \\
&= \frac{1}{m^{(s)}} \sum_{t=1}^{m^{(s)}} \mathbb{E}\big[Q_f(\mathbf{x}_t^{(s)})\big], \qquad (5.9.17)
\end{aligned}$$

where the inequality is obtained by the property of the convex function $f(\mathbf{x})$ that $f(\widetilde{\mathbf{x}}^{(s)}) \leq (1/m^{(s)}) \sum_{t=1}^{m^{(s)}} f(\mathbf{x}_t^{(s)})$ under Assumption 5.2.2. Recall that $0 < \eta < 1/(4L_M)$, where $L_M = \max_i L_i$. Replacing term $\sum_{t=1}^{m^{(s)}} \mathbb{E}\big[Q_f(\mathbf{x}_t^{(s)})\big]$ on the left-hand side of (5.9.16) with the term $m^{(s)}\mathbb{E}\big[Q_f(\widetilde{\mathbf{x}}^{(s)})\big]$ according to (5.9.17), and dividing both sides by $m^{(s)}(1 - 4L_M\eta)$, we have

$$\begin{aligned}
\mathbb{E}\big[Q_f(\widetilde{\mathbf{x}}^{(s)})\big] \leq \frac{1}{2m^{(s)}\eta(1-4L_M\eta)} \Big[& \mathbb{E}\big[\big\|\mathbf{x}_0^{(s)} - \mathbf{x}^*\big\|_{\mathbf{P}}^2 - \big\|\mathbf{x}_{m^{(s)}}^{(s)} - \mathbf{x}^*\big\|_{\mathbf{P}}^2\big] \\
& + 8L_M\eta^2 Q_f(\mathbf{x}_0^{(s)}) - 8L_M\eta^2 \mathbb{E}\big[Q_f(\mathbf{x}_{m^{(s)}}^{(s)})\big] + 8L_M m^{(s)} \eta^2 Q_f(\widetilde{\mathbf{x}}^{(s-1)}) \\
& + 2\eta \sum_{t=1}^{m^{(s)}} \mathbb{E}\big[\big\langle \rho \mathbf{A}^\top (\mathbf{z}_t^{(s)} - \mathbf{z}^*), \mathbf{x}^* - \mathbf{x}_t^{(s)} \big\rangle\big] \Big].
\end{aligned}$$

5.9.7 Proof of Lemma 5.6.9

Recall the primal variable update step in Line 16 of Algorithm 2 that $\widetilde{\mathbf{y}}^{(s)} = (1/m^{(s)}) \sum_{t=1}^{m^{(s)}} \mathbf{y}_t^{(s)}$. With the component gap function $Q_g(\mathbf{y})$ defined in (5.5.2) we have

$$\begin{aligned}
\mathbb{E}\big[Q_g(\widetilde{\mathbf{y}}^{(s)})\big] &\leq \mathbb{E}\Big[\frac{1}{m^{(s)}} \sum_{t=1}^{m^{(s)}} g(\mathbf{y}_t^{(s)}) - g(\mathbf{y}^*) - \big\langle g'(\mathbf{y}^*), \widetilde{\mathbf{y}}^{(s)} - \mathbf{y}^* \big\rangle\Big] \\
&= \mathbb{E}\Big[\frac{1}{m^{(s)}} \sum_{t=1}^{m^{(s)}} g(\mathbf{y}_t^{(s)}) - g(\mathbf{y}^*) - \big\langle g'(\mathbf{y}^*), \frac{1}{m^{(s)}} \sum_{t=1}^{m^{(s)}} \mathbf{y}_t^{(s)} - \mathbf{y}^* \big\rangle\Big] \\
&= \frac{1}{m^{(s)}} \sum_{t=1}^{m^{(s)}} \mathbb{E}\big[Q_g(\mathbf{y}_t^{(s)})\big], \qquad (5.9.18)
\end{aligned}$$

where the inequality is obtained by the property of the convex function $g(\mathbf{y})$ that $g(\widetilde{\mathbf{y}}^{(s)}) \leq (1/m^{(s)}) \sum_{t=1}^{m^{(s)}} g(\mathbf{y}_t^{(s)})$ under Assumption 5.2.2. Note that the first two terms of $Q_g(\mathbf{y}_t^{(s)})$ are $g(\mathbf{y}_t^{(s)}) - g(\mathbf{y}^*)$. These two terms are bounded as follows.

First, we reproduce the SVR-ADMM-D y-minimization step (Line 9 of Algorithm 2):

$$\mathbf{y}_t^{(s)} = \operatorname*{argmin}_{\mathbf{y}} g(\mathbf{y}) + \frac{\rho}{2}\big\|\mathbf{A}\mathbf{x}_{t-1}^{(s)} + \mathbf{B}\mathbf{y} - \mathbf{c} + \mathbf{z}_{t-1}^{(s)}\big\|^2. \tag{5.9.19}$$

By taking derivatives of the right-hand side of (5.9.19) with respect to $\mathbf{y}$, the analytical solution for $\mathbf{y}_t^{(s)}$ satisfies

$$g'(\mathbf{y}_t^{(s)}) + \rho\mathbf{B}^\top\big(\mathbf{A}\mathbf{x}_{t-1}^{(s)} + \mathbf{B}\mathbf{y}_t^{(s)} - \mathbf{c} + \mathbf{z}_{t-1}^{(s)}\big) = 0,$$

which leads to the relation between the first two terms of $Q_g(\mathbf{y}_t^{(s)})$:

$$g(\mathbf{y}_t^{(s)}) - g(\mathbf{y}^*) \le -\big\langle g'(\mathbf{y}_t^{(s)}), \mathbf{y}^* - \mathbf{y}_t^{(s)}\big\rangle = \big\langle \rho\mathbf{B}^\top\big(\mathbf{A}\mathbf{x}_{t-1}^{(s)} + \mathbf{B}\mathbf{y}_t^{(s)} - \mathbf{c} + \mathbf{z}_{t-1}^{(s)}\big), \mathbf{y}^* - \mathbf{y}_t^{(s)}\big\rangle,$$

where the inequality is by the convexity of the function $g(\mathbf{y})$ under Assumption 5.2.2. Recalling the SVR-ADMM-D dual update step (Line 13 of Algorithm 2) that

$$\mathbf{z}_t^{(s)} = \mathbf{z}_{t-1}^{(s)} + \mathbf{A}\mathbf{x}_t^{(s)} + \mathbf{B}\mathbf{y}_t^{(s)} - \mathbf{c}, \tag{5.9.20}$$

we further obtain

$$\begin{aligned} g(\mathbf{y}_t^{(s)}) - g(\mathbf{y}^*) &\le \Big\langle \rho\mathbf{B}^\top\big(\mathbf{A}\mathbf{x}_{t-1}^{(s)} + \mathbf{z}_t^{(s)} - \mathbf{A}\mathbf{x}_t^{(s)}\big), \mathbf{y}^* - \mathbf{y}_t^{(s)}\Big\rangle \\ &= \rho\Big\langle \mathbf{B}^\top\mathbf{A}\big(\mathbf{x}_{t-1}^{(s)} - \mathbf{x}_t^{(s)}\big), \mathbf{y}^* - \mathbf{y}_t^{(s)}\Big\rangle + \big\langle \rho\mathbf{B}^\top\mathbf{z}_t^{(s)}, \mathbf{y}^* - \mathbf{y}_t^{(s)}\big\rangle \\ &= \rho\Big\langle \mathbf{B}^\top\mathbf{A}\big(\mathbf{x}_{t-1}^{(s)} - \mathbf{x}_t^{(s)}\big), \mathbf{y}^* - \mathbf{y}_t^{(s)}\Big\rangle + \big\langle \rho\mathbf{B}^\top(\mathbf{z}_t^{(s)} - \mathbf{z}^*) - g'(\mathbf{y}^*), \mathbf{y}^* - \mathbf{y}_t^{(s)}\big\rangle, \end{aligned} \tag{5.9.21}$$

where the last equality is according to the optimality condition in (5.3.6).

Using the relation between the first two terms of $Q_g(\mathbf{y}_t^{(s)})$ in (5.9.21), together with (5.9.18) we have

$$\begin{aligned} \mathbb{E}\big[Q_g(\widetilde{\mathbf{y}}^{(s)})\big] \le \frac{1}{m^{(s)}}\sum_{t=1}^{m^{(s)}} \mathbb{E}\Big[&\rho\Big\langle \mathbf{B}^\top\mathbf{A}\big(\mathbf{x}_{t-1}^{(s)} - \mathbf{x}_t^{(s)}\big), \mathbf{y}^* - \mathbf{y}_t^{(s)}\Big\rangle \\ &+ \big\langle \rho\mathbf{B}^\top(\mathbf{z}_t^{(s)} - \mathbf{z}^*) - g'(\mathbf{y}^*), \mathbf{y}^* - \mathbf{y}_t^{(s)}\big\rangle - \big\langle g'(\mathbf{y}^*), \mathbf{y}_t^{(s)} - \mathbf{y}^*\big\rangle\Big]. \end{aligned} \tag{5.9.22}$$

The first inner product term on the right-hand side of (5.9.22) can be re-written and simplified as

$$
\begin{aligned}
&\rho\Big\langle \mathbf{B}^\top \mathbf{A}\big(\mathbf{x}_{t-1}^{(s)} - \mathbf{x}_t^{(s)}\big), \mathbf{y}^* - \mathbf{y}_t^{(s)} \Big\rangle \\
&= \frac{\rho}{2}\Big[\big\|\mathbf{A}\mathbf{x}_{t-1}^{(s)} + \mathbf{B}\mathbf{y}^* - \mathbf{c}\big\|^2 + \big\|\mathbf{A}\mathbf{x}_t^{(s)} + \mathbf{B}\mathbf{y}_t^{(s)} - \mathbf{c}\big\|^2 - \big\|\mathbf{A}\mathbf{x}_t^{(s)} + \mathbf{B}\mathbf{y}^* - \mathbf{c}\big\|^2 - \big\|\mathbf{A}\mathbf{x}_{t-1}^{(s)} + \mathbf{B}\mathbf{y}_t^{(s)} - \mathbf{c}\big\|^2 \Big] \\
&\leq \frac{\rho}{2}\Big[\big\|\mathbf{A}\mathbf{x}_{t-1}^{(s)} + \mathbf{B}\mathbf{y}^* - \mathbf{c}\big\|^2 + \big\|\mathbf{A}\mathbf{x}_t^{(s)} + \mathbf{B}\mathbf{y}_t^{(s)} - \mathbf{c}\big\|^2 - \big\|\mathbf{A}\mathbf{x}_t^{(s)} + \mathbf{B}\mathbf{y}^* - \mathbf{c}\big\|^2 \Big].
\end{aligned}
$$

Thus, re-using the SVR-ADMM-D dual update step in (5.9.20) repeatedly, we have

$$
\begin{aligned}
&\rho\Big\langle \mathbf{B}^\top \mathbf{A}\big(\mathbf{x}_{t-1}^{(s)} - \mathbf{x}_t^{(s)}\big), \mathbf{y}^* - \mathbf{y}_t^{(s)} \Big\rangle \\
&\leq \frac{\rho}{2}\Big[\big\|\mathbf{A}\mathbf{x}_{t-1}^{(s)} + \mathbf{B}\mathbf{y}^* - \mathbf{c}\big\|^2 + \big\|\mathbf{z}_t^{(s)} - \mathbf{z}_{t-1}^{(s)}\big\|^2 - \big\|\mathbf{A}\mathbf{x}_t^{(s)} + \mathbf{B}\mathbf{y}^* - \mathbf{c}\big\|^2 \Big] \\
&= \frac{\rho}{2}\Big[\big\|\mathbf{A}\mathbf{x}_{t-1}^{(s)} - \mathbf{A}\mathbf{x}^*\big\|^2 - \big\|\mathbf{A}\mathbf{x}_t^{(s)} - \mathbf{A}\mathbf{x}^*\big\|^2 \Big] + \frac{\rho}{2}\Big[\big\|\mathbf{z}_{t-1}^{(s)} - \mathbf{z}^*\big\|^2 - \big\|\mathbf{z}_t^{(s)} - \mathbf{z}^*\big\|^2 \Big] + \rho\big\langle \mathbf{z}_t^{(s)} - \mathbf{z}^*, \mathbf{z}_t^{(s)} - \mathbf{z}_{t-1}^{(s)} \big\rangle \\
&= \frac{\rho}{2}\Big[\big\|\mathbf{x}_{t-1}^{(s)} - \mathbf{x}^*\big\|_{\mathbf{A}^\top\mathbf{A}}^2 - \big\|\mathbf{x}_t^{(s)} - \mathbf{x}^*\big\|_{\mathbf{A}^\top\mathbf{A}}^2 \Big] \\
&\quad + \frac{\rho}{2}\Big[\big\|\mathbf{z}_{t-1}^{(s)} - \mathbf{z}^*\big\|^2 - \big\|\mathbf{z}_t^{(s)} - \mathbf{z}^*\big\|^2 \Big] + \rho\big\langle \mathbf{z}_t^{(s)} - \mathbf{z}^*, \mathbf{A}\mathbf{x}_t^{(s)} + \mathbf{B}\mathbf{y}_t^{(s)} - \mathbf{c} \big\rangle, \qquad (5.9.23)
\end{aligned}
$$

where the first equality is obtained by the optimality condition in (5.3.4). Using the results of (5.9.23) for the first inner product term on the right-hand side of (5.9.22), we have

$$
\begin{aligned}
\mathbb{E}\big[Q_g(\widetilde{\mathbf{y}}^{(s)})\big] &\leq \frac{1}{m^{(s)}} \sum_{t=1}^{m^{(s)}} \mathbb{E}\Big[\big\langle \rho\mathbf{B}^\top(\mathbf{z}_t^{(s)} - \mathbf{z}^*) - g'(\mathbf{y}^*), \mathbf{y}^* - \mathbf{y}_t^{(s)} \big\rangle - \langle g'(\mathbf{y}^*), \mathbf{y}_t^{(s)} - \mathbf{y}^* \rangle \\
&\quad + \frac{\rho}{2}\Big[\big\|\mathbf{x}_{t-1}^{(s)} - \mathbf{x}^*\big\|_{\mathbf{A}^\top\mathbf{A}}^2 - \big\|\mathbf{x}_t^{(s)} - \mathbf{x}^*\big\|_{\mathbf{A}^\top\mathbf{A}}^2 \Big] + \frac{\rho}{2}\Big[\big\|\mathbf{z}_{t-1}^{(s)} - \mathbf{z}^*\big\|^2 \\
&\quad - \big\|\mathbf{z}_t^{(s)} - \mathbf{z}^*\big\|^2 \Big] + \rho\big\langle \mathbf{z}_t^{(s)} - \mathbf{z}^*, \mathbf{A}\mathbf{x}_t^{(s)} + \mathbf{B}\mathbf{y}_t^{(s)} - \mathbf{c} \big\rangle \Big] \\
&= \frac{1}{2m^{(s)}\eta} \sum_{t=1}^{m^{(s)}} \mathbb{E}\Big[2\eta\big\langle \rho\mathbf{B}^\top(\mathbf{z}_t^{(s)} - \mathbf{z}^*), \mathbf{y}^* - \mathbf{y}_t^{(s)} \big\rangle + 2\eta\rho\big\langle \mathbf{z}_t^{(s)} - \mathbf{z}^*, \\
&\quad \mathbf{A}\mathbf{x}_t^{(s)} + \mathbf{B}\mathbf{y}_t^{(s)} - \mathbf{c} \big\rangle \Big] + \frac{1}{2m^{(s)}\eta} \sum_{t=1}^{m^{(s)}} \mathbb{E}\Big[\eta\rho\Big[\big\|\mathbf{x}_{t-1}^{(s)} - \mathbf{x}^*\big\|_{\mathbf{A}^\top\mathbf{A}}^2 \\
&\quad - \big\|\mathbf{x}_t^{(s)} - \mathbf{x}^*\big\|_{\mathbf{A}^\top\mathbf{A}}^2 \Big] + \eta\rho\Big[\big\|\mathbf{z}_{t-1}^{(s)} - \mathbf{z}^*\big\|^2 - \big\|\mathbf{z}_t^{(s)} - \mathbf{z}^*\big\|^2 \Big] \Big]. \qquad (5.9.24)
\end{aligned}
$$

Re-arranging terms on the right-hand side of (5.9.24), we have

$$
\begin{aligned}
&2\eta\big\langle\rho\mathbf{B}^\top(\mathbf{z}_t^{(s)}-\mathbf{z}^*),\mathbf{y}^*-\mathbf{y}_t^{(s)}\big\rangle+2\eta\rho\big\langle\mathbf{z}_t^{(s)}-\mathbf{z}^*,\mathbf{A}\mathbf{x}_t^{(s)}+\mathbf{B}\mathbf{y}_t^{(s)}-\mathbf{c}\big\rangle\\
&=2\eta\rho\Big[\big\langle\mathbf{z}_t^{(s)}-\mathbf{z}^*,\mathbf{B}\big(\mathbf{y}^*-\mathbf{y}_t^{(s)}\big)\big\rangle+\big\langle\mathbf{z}_t^{(s)}-\mathbf{z}^*,\mathbf{A}\mathbf{x}_t^{(s)}+\mathbf{B}\mathbf{y}_t^{(s)}-\mathbf{c}\big\rangle\Big]\\
&=2\eta\rho\big\langle\mathbf{z}_t^{(s)}-\mathbf{z}^*,\mathbf{A}\mathbf{x}_t^{(s)}-\mathbf{A}\mathbf{x}^*\big\rangle\\
&=-2\eta\Big\langle\rho\mathbf{A}^\top\big(\mathbf{z}_t^{(s)}-\mathbf{z}^*\big),\mathbf{x}^*-\mathbf{x}_t^{(s)}\Big\rangle,
\end{aligned}
\tag{5.9.25}
$$

where the second equality is obtained by the optimality condition in (5.3.4). Replacing the term $2\eta\big\langle\rho\mathbf{B}^\top(\mathbf{z}_t^{(s)}-\mathbf{z}^*),\mathbf{y}^*-\mathbf{y}_t^{(s)}\big\rangle+2\eta\rho\big\langle\mathbf{z}_t^{(s)}-\mathbf{z}^*,\mathbf{A}\mathbf{x}_t^{(s)}+\mathbf{B}\mathbf{y}_t^{(s)}-\mathbf{c}\big\rangle$ on the right-hand side of (5.9.24) by the right-hand-side result of (5.9.25), and with the telescoping sum, we have

$$
\begin{aligned}
\mathbb{E}\big[Q_g(\widetilde{\mathbf{y}}^{(s)})\big]\leq\frac{1}{2m^{(s)}\eta}\Big[&-2\eta\sum_{t=1}^{m^{(s)}}\mathbb{E}\big[\big\langle\rho\mathbf{A}^\top(\mathbf{z}_t^{(s)}-\mathbf{z}^*),\mathbf{x}^*-\mathbf{x}_t^{(s)}\big\rangle\big]\\
&+\eta\rho\mathbb{E}\big[\big\|\mathbf{x}_0^{(s)}-\mathbf{x}^*\big\|_{\mathbf{A}^\top\mathbf{A}}^2-\big\|\mathbf{x}_{m^{(s)}}^{(s)}-\mathbf{x}^*\big\|_{\mathbf{A}^\top\mathbf{A}}^2\big]\\
&+\eta\rho\mathbb{E}\big[\big\|\mathbf{z}_0^{(s)}-\mathbf{z}^*\big\|^2-\big\|\mathbf{z}_{m^{(s)}}^{(s)}-\mathbf{z}^*\big\|^2\big]\Big].
\end{aligned}
$$

5.9.8 Proof of Lemma 5.6.10

Since $0<\eta<1/(4L_M)$, we have $0<1-4L_M\eta<1$. Recall that $Q_g(\mathbf{y})\geq 0$ for all $\mathbf{y}$ as discussed in Remark 5.5.2. By Lemma 5.6.9 we further have

$$
\begin{aligned}
\mathbb{E}\big[Q_g(\widetilde{\mathbf{y}}^{(s)})\big]\leq\frac{1}{2m^{(s)}\eta(1-4L_M\eta)}\Big[&-2\eta\sum_{t=1}^{m^{(s)}}\mathbb{E}\big[\big\langle\rho\mathbf{A}^\top(\mathbf{z}_t^{(s)}-\mathbf{z}^*),\mathbf{x}^*-\mathbf{x}_t^{(s)}\big\rangle\big]\\
&+\eta\rho\mathbb{E}\big[\big\|\mathbf{x}_0^{(s)}-\mathbf{x}^*\big\|_{\mathbf{A}^\top\mathbf{A}}^2-\big\|\mathbf{x}_{m^{(s)}}^{(s)}-\mathbf{x}^*\big\|_{\mathbf{A}^\top\mathbf{A}}^2\big]\\
&+\eta\rho\mathbb{E}\big[\big\|\mathbf{z}_0^{(s)}-\mathbf{z}^*\big\|^2-\big\|\mathbf{z}_{m^{(s)}}^{(s)}-\mathbf{z}^*\big\|^2\big]\Big].
\end{aligned}
\tag{5.9.26}
$$

Recall (5.5.2) that $Q(\widetilde{\mathbf{x}}^{(s)},\widetilde{\mathbf{y}}^{(s)})=Q_f(\widetilde{\mathbf{x}}^{(s)})+Q_g(\widetilde{\mathbf{y}}^{(s)})$. By combining the results for both component gap functions in Lemma 5.6.8 and (5.9.26), we complete the proof.

Chapter 6

Intention Analysis from Human Activities as Motivated by Security

Human activities are human-centric. Now we study an intention analysis problem from medical service providers' electronic health record access activities as motivated by the security perspective. In health care institutions, medical specialty information may be lacking or inaccurate. As a result, false alarms of suspicious accesses to electronic health records might be raised. We think that medical service providers can save their efforts in resolving such false alarms if their actual related specialties can be recognized and assigned to them. In fact, diagnosis histories offer information on which medical specialties may exist in practice, regardless of whether they have official codes. We refer to such specialties that are predicted with high certainty by diagnosis histories as *de facto* diagnosis specialties. Since the false alarms of suspicious accesses to electronic health records may be due to the lacking or inaccurate medical specialty information, we aim to discover *de facto* diagnosis specialties, which reflect medical service providers' genuine and permissible intentions in accessing electronic health records with certain diagnoses. The problem is studied under a general discovery–evaluation framework. Specifically, we employ a semi-supervised learning model analyzing heterogeneous information networks and an unsupervised learning method for discovery. We further employ four supervised learning models for evaluation. We use one year of diagnosis histories from a major medical center, which consists of two data sets: one is fine-grained and the other is general. The semi-supervised learning model discovers a specialty for *Breast Cancer* on the fine-grained data set; while the unsupervised learning method confirms this discovery and suggests another specialty for *Obesity* on the larger general data set. The evaluation results reinforce that these two specialties can be recognized accurately by supervised learning models in comparison with 12 common diagnosis specialties defined by the Health Care Provider Taxonomy Code Set.

6.1 Introduction

Medical specialties provide information about which medical service providers (hereinafter referred to as "providers") have the skills needed to carry out key procedures or make critical judgments. They are useful

for training and staffing, as well as providing confidence to patients that their providers have the expertise required to address their problems.

Health care institutions have many ways to express and take advantage of staff specialties, including organizing them into departments or wards. However, such an organization has its limitations. For instance, at a large and diverse medical center, some specialties may be lacking or inaccurately described (*e.g.*, they are not always entered for new hire documents), employees can change roles, and encoded departments do not always align with specialties. As a result, there could be a gap between the diagnosis histories of certain providers and their specialties. There is thus an opportunity to design and apply data-driven techniques that assist in the management of health care operations, such as staffing (by providing accurate specialty information about current staff), quality control (by verifying that providers practice consistently with their declared specialties), and building patient confidence (by ensuring that patients are treated by specialists) [57].

Health care providers select from the Health Care Provider Taxonomy Code Set (HPTCS) [45] when they apply for their National Provider Identifiers (NPIs) [2]. NPIs are required by the Health Insurance Portability and Accountability Act (HIPAA) of 1996 and are used in health care-related transactions. Health care providers usually choose their taxonomy codes according to the certifications they hold. Ideally, this mechanism would identify each health care provider with the taxonomy codes that most accurately describe their specialties. However, this is not always the case for several reasons. First, the National Plan & Provider Enumeration System does not verify that the taxonomy code selections made by providers in NPI applications are accurate [45]. Second, certain taxonomy codes do not correspond to any nationwide certifications that are approved by a professional board. For example, the specialty for *Men and Masculinity* is a well-recognized area of interest, study, and activity in the field of psychology; however, there is no certification or credential available to identify psychologists who might work in this area [3]. Third, some national certifications are not reflected by the taxonomy code list. Since the taxonomy codes do not correspond to certifications within the field, providers may interpret these codes in inconsistent ways.

In view of the aforementioned limitations of purely relying on NPI taxonomy codes, we propose to leverage real-world diagnosis histories to infer and recognize actual specialties. We refer to such inferred knowledge as *de facto* specialties. *De facto* specialties are medical specialties that exist in practice regardless of the specialty codes (NPI taxonomy codes). To illustrate, imagine that there is a method for recognizing providers' *de facto* specialties based on their actual activities related to diagnosis histories. This enables us to verify the NPI taxonomy codes of the providers in a health care institution. If certain providers' declared

specialties failed to match their activity-based specialties, such as electronic health record (EHR) access, an investigation and possible re-designation of their codes might be warranted.

As the medical profession evolves, the HPTCS needs to be updated to be more comprehensive [15, 40, 143]. Problems and inefficiencies could arise if the specialty codes are not sufficiently expressive to convey providers' specialties. For instance, if there is no official code to express such specialties and no providers declared them, false alarms of suspicious EHR access detection might be raised because such unlisted *de facto* specialties could not be assigned to any providers. Other concerns have been voiced by the American Psychological Association: " *... several national certifications that do exist are not reflected on the specialty code list. Since the specialty codes do not correspond to certifications within the field, psychologists will interpret these codes in different ways. Use of the specialty codes by psychologists therefore will not be uniform and will not provide meaningful information about a psychologist's practice.*" [3]

The focus of our research is on ***de facto* diagnosis specialties** of providers that exist in practice and *are highly predictable by the diagnoses in the EHRs of the patients they treat.* Our goal is to discover *de facto* diagnosis specialties that do not have corresponding codes in the Health Care Provider Taxonomy Code Set. In this study, we use a subset of such codes for both discovery and evaluation. To provide intuition into the problem, let us consider a perfect scenario where *every* NPI code correctly reflects specialties in a data set. If machine learning models are trained on this data set and exhibit decent performance, we believe that such models would reliably discover *de facto* diagnosis specialties in a new data set; the new data set may be provided by another health care institution that needs more reliable *de facto* diagnosis specialty discovery. However, in practice this perfect scenario will not be realized. In this work, we consider a more challenging scenario where we assume that *majority* of the NPI codes correctly reflect specialties in our collected data set.

This study makes three contributions. First, we propose a novel *de facto* diagnosis specialty discovery problem. To solve it, we introduce a discovery–evaluation framework. Specifically, *de facto* diagnosis specialties are proposed and their recognition accuracy is subsequently evaluated in comparison with existing diagnosis specialties listed in the HPTCS. Although we rely on expert opinions to interpret our discovery results, we consider evaluation important because expert opinions may not always be available in practice.

Second, under the discovery–evaluation framework, we employ a semi-supervised learning model (based on heterogeneous information network analysis) on a fine-grained data set and an unsupervised learning method (based on topic modeling) on a larger general data set for discovery. We further employ four supervised learning models for evaluation. Details of the two data sets are described in Section 6.3.

Third, we perform an empirical investigation using one year of diagnosis histories from a major medical center, which consists of two data sets. One is fine-grained and has diagnoses assigned to 41,603 patients that are accessed by 2,504 providers. The other is general and has diagnoses assigned to 291,562 patients that are accessed by 3,269 providers. The semi-supervised model discovers a *de facto* diagnosis specialty for *Breast Cancer* on the fine-grained data set; the unsupervised learning method confirms this discovery and suggests a new *de facto* diagnosis specialty for *Obesity* on the larger general data set. The evaluation results reinforce that these two specialties can be recognized accurately by supervised learning models in comparison with 12 common *de facto* diagnosis specialties defined by HPTCS.

6.2 De Facto Diagnosis Specialty

In Section 6.1, we define *de facto* diagnosis specialties as medical specialties that exist in practice and are highly predictable by the diagnoses inherent in EHRs. Here we illustrate this concept in more detail.

Intuitively, it should be easier to characterize a urologist in terms of medical diagnoses for conditions, for example, of the kidney, ureter, and bladder, as opposed to an anesthesiologist, whose duties are more cross-cutting with respect to diagnoses, concerning essentially all conditions related to surgeries. To orient the reader using a concrete example, let us test this hypothesis with a naïve classifier based on diagnosis codes. To gain intuition into the general idea, consider the following steps. First, we begin with a data set that indicates which EHRs have been accessed by urologists and anesthesiologists, and view each patient as a document whose words are diagnoses in their EHRs. Next, we create a weighting for how many diagnoses of each kind are accessed by each provider, with some adjustment for how common the diagnosis is. This technique is typified by term frequency–inverse document frequency (TF-IDF, with details in Section 6.4.4). We believe such a naïve classifier is the type of model that an administrator might define and apply to infer a specialty from a diagnosis history. The technique proceeds by finding the most relevant diagnoses of each diagnosis specialty (taxonomy code) and the most frequently accessed diagnoses of each provider. Finally, providers are classified according to the specialties with which they share the most commonly accessed diagnoses.

Using the general data set for the empirical study below (details in Section 6.3), we observe that urologists tend to access diagnoses such as "retention of urine" and "urinary tract infection", whereas anesthesiologists tend to access diagnoses such as "other aftercare" and "other screening". When we use the 20 conditions most accessed by either of the two specialties as the features for the naïve classifier, the results are decent for urology, yielding an F_1 score of 70.35% in predicting the urologists[1]. However, the results for anesthesiologists

[1] A higher F_1 score indicates a better performance (more details are provided in Section 6.5.1).

are poorer, yielding F_1 score of 11.30%. If we use a machine learning technique, such as SVM (described in Section 6.4.4), we can achieve substantially better results: finding anesthesiologists with an F_1 score of 48.98%. However, this performance is still weaker than the classifier learned for urologists, which achieves an F_1 score of 97.44%.

De facto specialties that are highly predictable by diagnosis histories are *de facto* diagnosis specialties. Note that there is no ground truth to determine the validity of a discovered *de facto* diagnosis specialty. Ideally, a discovered *de facto* diagnosis specialty can be recognized by classifiers as accurately as the existing listed diagnosis specialties. To illustrate how this is possible, consider an analogy with respect to the classification of documents, an area that has inspired many of the techniques we apply. The providers U can be likened to readers of documents, where A represents an archive of documents in which the words in each document correspond to diagnoses. A function $T(u)$ indicates the collection of documents that a provider u has read. Providers with specialties are groups of readers who (presumably) have a common *de facto* diagnosis specialty and interest in the same group. To solve the *de facto* diagnosis specialty discovery problem we aim to develop a classifier that characterizes this common interest in terms of the documents that they have read, if possible. For instance, if there are a group of readers that are ophthalmologists and they are inordinately interested in documents on disorders of the eyes, then we can use this proclivity to serve as a discriminatory feature.

6.3 Data

Following the aforementioned analogy to the document classification, we use access log data from a hospital and combine it with the diagnosis lists in patient discharge records. That is, for each encounter (visit to the hospital by a patient) we have a set of diagnoses, and for each provider we have a record of whether the provider accessed the chart of that patient during the time of that encounter. If a provider u accessed the patient during that encounter, we include the diagnosis set for that encounter in $T(u)$. We will refer to users (as in chart users) rather than providers for our technical discussion.

We collect data for this study via the Cerner Powerchart EHR system in use at Northwestern Memorial Hospital (NMH). The data contain all user accesses (in the form of audit logs) made over a one-year period, as well as insurance billing code lists, in the form of International Classification of Diseases–ninth revision (ICD-9), for patient encounters during this period. All data were de-identified for this study in accordance with the HIPAA Privacy Rule and carried out under Institutional Review Board approval. Since specialties

Table 6.1: A summary of the attributes for NMH audit logs for the fine-grained and general data sets.

	Fine-Grained	General
Accesses	35,869	4,829,376
Patients	41,603	291,562
Providers	2,504	3,269
Patient encounters	62,390	890,812
Taxonomy codes	161	165

Table 6.2: A summary of the attributes for patient records in NMH audit logs for the fine-grained and general data sets.

	Fine-Grained	General
Provider job titles	167	171
Locations	242	251
Services	101	104
Diagnoses	4,172	13,566
Procedures	740	2,165

are mainly focused on physicians, we filter out users with other positions (*e.g.*, nurses and dieticians) from the data set.

A small portion of the collected data has an explicit mapping between users and diagnoses of the EHRs they accessed. However, majority of the data lacks such an explicit relationship. This is because patients may have multiple diagnoses and their EHRs may be accessed by different users without documentation on which specific diagnoses were associated with the actions of which user. We refer to the former portion as the *fine-grained data set*. As fine-grained data may not always be available, we expand to a more general data set for our study that may be more representative of the challenging scenarios encountered in practice. Hence, we use the entire data after removing all such fine-grained mapping information to form the other data set, which we call it the *general data set*. The attributes of the data sets used in this study are summarized in Table 6.1—6.2.

We use Clinical Classifications Software (CCS) to cluster diagnosis and procedure codes into a manageable number of clinically meaningful categories [42]. This is because ICD-9 codes are not completely indicative of patients' clinical phenotypes [11] and the sheer number of codes (on the order of 10,000) makes it challenging to characterize patterns of diagnoses or procedures. The ICD-9 codes for diagnoses are mapped down to 603 CCS codes and the ICD-9-CM codes for procedures are mapped down to 346 CCS codes. A key characteristic of the data set relevant to this study is that it also contains NPI taxonomy codes for 60% of the providers. About 150 classes of NPI taxonomy codes are listed in the data sets, but most have fewer than 10 user instances. Figure 6.1 shows the frequency distribution of 100 most frequent taxonomy codes in the data set.

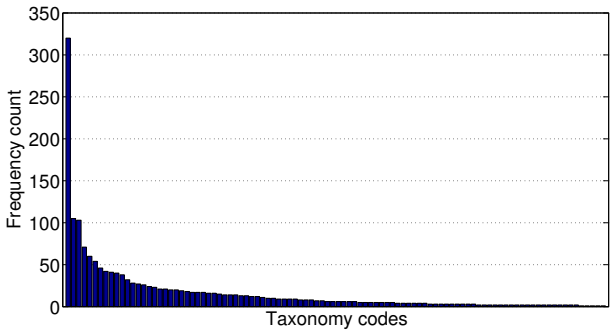


Figure 6.1: The frequency distribution for the 100 most frequent taxonomy codes in the general data set.

To ensure there is a sufficient amount of data to train machine learning models, we filter out NPI taxonomy codes with fewer than 20 user instances [62]. Based on the guidance of several clinicians and hospital administrators, we further identify 12 NPI taxonomy codes as diagnosis specialties: *Obstetrics & Gynecology, Cardiovascular Disease, Neurology, Ophthalmology, Gastroenterology, Dermatology, Orthopaedic Surgery, Neonatal-Perinatal Medicine, Infectious Disease, Pulmonary Disease, Neurological Surgery*, and *Urology*. We refer to this group as the *core NPI taxonomy codes*. As discussed in Section 6.1, we assume that a majority of these codes correctly reflect specialties in the data.

6.4 Methods

In this section, we describe the methods for discovering and evaluating the *de facto* diagnosis specialties.

6.4.1 Discovery–Evaluation

We highlight that there is no ground truth for the *de facto* diagnosis specialty discovery problem. Hence, we solve it under a general discovery–evaluation framework.

Discovery

We invoke machine learning to discover potential *de facto* diagnosis specialties in the data set that lack corresponding codes in the HPTCS. In this study, we first employ a semi-supervised learning model (in the form of PathSelClus [152]) to leverage the mapping between users and their specifically accessed diagnoses of EHRs in the fine-grained data set. Then we consider a more challenging scenario where such fine-grained mapping is not available. In this case, we employ an unsupervised learning model (in the form of Latent Dirichlet Allocation [14]) for discovery in the larger general data set. Since the fine-grained data set is a

subset of the general data set, except for the fine-grained mapping information, the discovery results can be reinforced when they exhibit common findings.

Evaluation

To interpret the discovery results, we rely on expert opinions. However, we acknowledge that in practice such opinions may not be available. Hence, we also make use of supervised learning models to evaluate the recognition accuracy of the discovered specialty by comparing our approach with the existing listed diagnosis specialties, such as the core NPI taxonomy codes described in Section 6.3. Ideally, their recognition accuracy should be similar. In this study, we evaluate such recognition accuracy using four classifiers, namely, decision trees, random forests, PCA-KNN, and SVM.

6.4.2 PathSelClus for Discovery

In general, discovering *de facto* diagnosis specialties from the diagnosis histories of providers may rely upon effective clustering techniques that can divide a pool of providers into groups that have high inter-group distances (distinctiveness), but low intra-group distances (coherence). We anticipate that, new diagnosis specialties may emerge from these clusters. The structure of our data sets can be represented as a typical *heterogeneous information network* [151, 152, 186]. Therefore, we use PathSelClus [152], a state-of-the-art semi-supervised learning model based on heterogeneous information networks for user-guided clustering. For context, we begin with a brief introduction to heterogeneous information networks.

Heterogeneous Information Networks

A heterogeneous information network consists of multiple types of objects and/or multiple types of links. A heterogeneous information network explicitly distinguishes between object types and relationship types in the network, which is quite different from traditional networks. For example, if a relation exists from type A to type B, denoted as ARB, then the inverse relation R^{-1} holds naturally for $BR^{-1}A$. R and its inverse R^{-1} are usually not equal, unless the two types are the same and R is symmetric.

Figure 6.2 depicts our data in the form of a heterogeneous information network and the corresponding schema. It contains 3 types of objects, namely user (U), patient (P) and diagnosis (D). Links exist between users and patients by the relation of "access" and "accessed by"; links exist between patient and diagnosis by the relation of "diagnosed with" and "assigned to".

Link-based clustering in heterogeneous information networks groups objects based on their connections to other objects in the networks. The possible relations derived from a heterogeneous information network

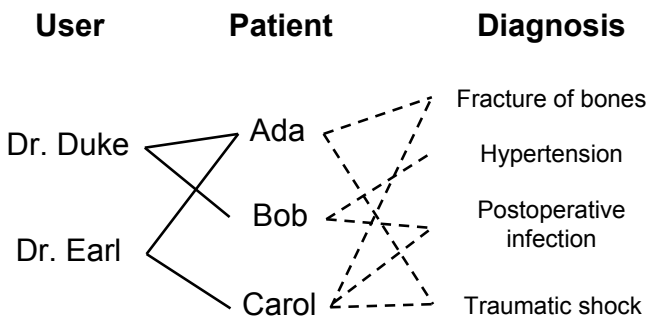


Figure 6.2: A toy example for visualizing the data set in the view of a heterogeneous information network. There are multiple types of nodes, such as users, patients and diagnoses; and multiple types of links between different types of nodes.

between two types of objects in a meta-level is called a *meta-path* [150]. In our case, the *target object type* to cluster is U (users). There are two meta-paths: $U \xrightarrow{access} P \xrightarrow{accessed\ by} U$ and $U \xrightarrow{access} P \xrightarrow{diagnosed\ with} D \xrightarrow{assigned\ to} P \xrightarrow{accessed\ by} U$.

User-Guided Clustering

During clustering, a decision has to be made about the weighted combination of different meta-paths to use. This is where user guidance comes into play. We use the semi-supervised learning model PathSelClus for user-guided clustering. In PathSelClus, user guidance is provided in the form of object seeds in each cluster. For example, to cluster users based on the pattern of the diagnoses of EHRs they access, one can provide several representative users as seeds for each pattern. These seeds provide guidance for clustering the target objects in the heterogeneous information networks and help select the most relevant meta-paths for the clustering task.

PathSelClus is designed to handle unseeded initial clusters because in practice, there may not be sufficient information to seed all the clusters. This is the exact feature that makes it possible to use PathSelClus to discover new diagnosis specialties. Now, let the number of listed diagnosis specialties be N and the number of *de facto* diagnosis specialties we want to discover be δ. We create $N + \delta$ empty clusters at the initiation of PathSelClus and seed N of them with corresponding specialists. The inputs of PathSelClus include all the users regardless of whether they have a taxonomy code.

As an output, each user is assigned to the cluster with the highest assignment likelihood. The δ unseeded clusters should also be filled with users. We can analyze the semantics of the unseeded clusters via the users they contain. We treat a cluster as a taxonomy code and calculate the most relevant diagnoses for each cluster. Then the medical expert labels the clusters, which we use to interpret the discovery results.

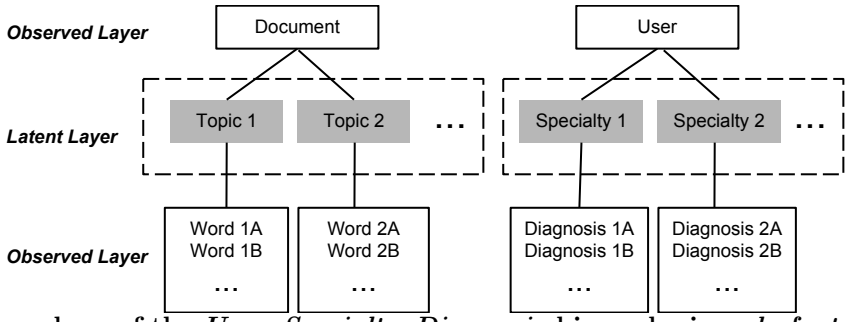


Figure 6.3: An analogy of the *User–Specialty–Diagnosis* hierarchy in a *de facto* diagnosis specialty discovery problem to the *Document–Topic–Word* hierarchy in a topic modeling problem.

Fine-Grained Data Set for PathSelClus

We emphasize that PathSelClus is a network-based learning model and relies on the mapping between users and their specifically accessed diagnoses in the fine-grained data set. In Section 6.5.2, we empirically compare and analyze PathSelClus in more detail on both the fine-grained data set and the general data set.

6.4.3 Latent Dirichlet Allocation (LDA) for Discovery

In practice, fine-grained data sets may not be available for PathSelClus. Hence, we also employ an unsupervised learning method [14], which is based on topic modeling.

General Data Set for LDA

In Latent Dirichlet Allocation (LDA) [14], topics act as summaries of the different themes pervasive in the corpus and documents are characterized with respect to these topics. The intuition behind our employment of LDA is diagnosis topics with coherent themes in a hospital. By treating each provider as a document in which the provider's associated diagnoses are the words and applying LDA to model all these documents, we can obtain an allocation of diagnosis topics for each provider. This analogy is illustrated in Figure 6.3. We can further cluster the providers using their topic allocations by the topic simplex[2] that they are closest to.

LDA does not leverage network information and does not require a fine-grained mapping between users and their accessed diagnoses. Instead, LDA models specialties with respect to different diagnosis themes. In this study, all the data sets to which LDA is applied refer to the larger general data set.

[2]This can be visualized by plotting the providers by their topic distributions.

Representation of Users

Diagnoses in our data set are provided with respect to patients, but not users. Therefore, we associate users with diagnoses via the patients they access. We consider two approaches for accomplishing this task.

User-document approach: For any user u_i, find the set of patients P_i whose EHR is accessed by u_i. Then, for each patient $p_j \in P_i$, let D_j be the set of diagnoses associated with p_j. We add diagnoses to D_j that occurred during the encounter of u_i and p_j to a set of diagnoses that represent u_i. The diagnosis topics and their allocations for users are discovered directly by applying LDA.

Patient-document approach: In this alternative approach, we start by applying LDA on the patient dimension to obtain a topic distribution in diagnoses for patients rather than users. Let T_{p_j} denote the topic distribution in diagnoses of patient p_j. Let T_{u_i} be the topic distribution of user u_i and let P_i be the set of patients whose EHRs are accessed by u_i. Then, the topic distribution for user u_i is

$$T_{u_i} = \frac{1}{|P_i|} \times \sum_{p_j \in P_i} T_{p_j}.$$

Both approaches were tested on the general data set. Table 6.3 shows one sample topic summary for both approaches. It is notable that the topic obtained from the user-document approach exhibits no clear theme, whereas the topic obtained from the patient-document approach has a consistent theme related to Urology. This is due to the fact that, in the user-document approach, each document contains the union of the diagnoses of all the accessed patients, whereas in the patient-document approach only the diagnoses of a single patient are in the document. The hodgepodge of many patients' diagnoses is likely to contain diverse and inconsistent themes, thus rendering the topics generated by the user-document approach not easily interpretable. Since discovering *de facto* diagnosis specialties requires experts to interpret such topics, we use the patient-document approach.

Choice of Topic Number

An important parameter for LDA is the number of topics k. There is no consensus on how to determine the best value of k. The sign of a good topic number is that the resulting topic summaries are semantically meaningful. The general rule for picking k is the perplexity measure [14]. This is an estimate of the expected number of equally likely words. Minimizing perplexity corresponds to maximizing the captured topic variance. Based on the perplexity measure, k is set to 30 in this study.

Table 6.3: A comparison of two sample *de facto* diagnosis specialties obtained by two different LDA approaches on the general data set. They are represented by 10 most probable diagnoses according to LDA. The user-document approach obtains more semantically random diagnoses, whereas the patient-document approach obtains a specialty with diagnoses consistent with a Urology theme.

Other hypertensive complications
Hypotension
Cancer of ovary
Coma, stupor, and brain damage
Hyposmolality
Ascites
Hematuria
Acute myocardial infarction
Backache, unspecified
Other connective tissue disease

Calculus of kidney
Elevated prostate specific antigen
Hematuria
Impotence of organic origin
Incomplete bladder emptying
Bladder neck obstruction
Urinary frequency
Hydronephrosis
Unspecified retention of urine
Other testicular hypofunction

Clustering Users

After applying LDA, each user is assigned to an allocation in the specialty topic simplex. A higher frequency in a specialty indicates that the user is more likely to access patients with diagnoses popular in that specialty. Therefore, if we cluster users by *de facto* diagnosis specialties, it is reasonable to cluster users by the closest specialties. This is because this specialty has the highest proportion in the specialty topic simplex:

$$C_{u_i} = \underset{t \in T}{\operatorname{argmax}} P(u_i, t),$$

where C_{u_i} denotes the specialty cluster assignment for the user u_i and T denotes the set of specialty topics, and $P(u_i, t)$ denotes the proportion of the topic t for the user u_i.

6.4.4 Classifiers for Evaluation

In PathSelClus, a *de facto* diagnosis specialty is represented by the most accessed diagnoses by all users in the same cluster that have such a specialty. In LDA, a *de facto* diagnosis specialty is represented by the most probable diagnoses as an output of the model. To interpret the discovered *de facto* diagnosis specialties, we rely on physicians (authors) with medical expertise. The experts reviewed the diagnosis summaries of the specialty and labeled each with one or a few medical themes that are pervasive in the specialty. After labeling, we compare the labeled specialties with the HPTCS to see if there are specialties that have pervasive themes but are not listed in the code set. If such specialties exist, they are considered to be potential newly discovered *de facto* diagnosis specialties. Since there is no ground truth for the discovery results, we use supervised learning models to evaluate the recognition accuracy of the discovered *de facto* diagnosis specialty. We briefly describe the four classifiers used in this study.

Decision Trees

A decision tree (J48) is constructed in a top-down recursive divide-and-conquer manner. To start, all the training examples are at the root. Examples are partitioned recursively based on selected attributes. Test attributes are selected on the basis of a heuristic or statistical measure. A decision tree is a popular nonlinear classifier because it is convertible to classification rules that can be reviewed and interpreted by experts.

Random Forests

To aggregate decision trees, we can use random forests. To do so, for $b \in \{1, \ldots, B\}$, we draw samples from the training data and grow a big tree T_b with some restrictions: at each split, randomly select m features from the p features and pick the best split among them. The recommended value (used in this study) for m is $\sqrt{p}$. Then the forests are represented as a collection of trees $\{T_b\}_{b=1}^{B}$. To classify a testing instance, we conduct majority voting among $T_1(x), ..., T_B(x)$.

KNN-PCA

K-Nearest Neighbors (KNN) is an instance-based learning method. It stores training examples and delays the processing until a new instance must be classified. All instances correspond to points in the n-dimensional space. The nearest neighbors are defined in terms of Euclidean distance. KNN returns the most common label among the K training examples nearest to the new testing instance. KNN is sensitive to the "curse of dimension" such that the distance between neighbors could be dominated by irrelevant attributes when the dimension of space goes higher. To mitigate this problem, we use principal component analysis (PCA) by selecting a small number of the principal components to perform dimension reduction.

SVM

A support vector machine (SVM) is a classification method for both linear and nonlinear data. It uses a nonlinear mapping to transform the original training data set into a higher dimension. We used a Gaussian kernel in this study. The SVM searches for the optimal linear separating hyperplain in this new space by using support vectors that lie closest to the decision boundary. In particular, SVM is effective on a high-dimensional data set because the complexity of the trained classifier is characterized by the number of support vectors rather than the dimension of the data set.

Classification

To apply these classifiers to our data, we map each user u_i in the set of users U to a TF-IDF weighted diagnosis vectors $v_i' = \{\mathsf{tfidf}_{\mathsf{d}_1}, ..., \mathsf{tfidf}_{\mathsf{d}_k}\}$ according to:

$$\mathsf{tfidf}_{d_j} = \log\left(\frac{v_i(d_j)}{a_i} + 1\right) \times \log\left(\frac{|U|+1}{r_{d_j}}\right),$$

where d_j is the diagnosis with the globally unique index j, and each user u_i has a vector $v_i = \{c_1, ..., c_k\}$ where c_j denotes the number of times that the user has accessed patients with d_j. Let a_i be the total count of all diagnoses in v_i, and let r_{d_j} be the number of users that have accessed patients with d_j. This vector, along with each user u_i's primary taxonomy code, serves as the input to these classifiers, with a length of 603. For KNN-PCA, we perform dimension reduction via PCA to the vectors before applying KNN. We do not use procedure codes because they are less expressive than diagnosis codes [100].

6.5 Experiment

This section describes the experiment setting and provides an analysis of the *de facto* specialty discovery results.

6.5.1 Setup and Evaluation Measures

We use Weka [58] for decision trees (J48), random forests, and SVM with the default parameter values. In PCA-KNN, the number of nearest neighbors K is set to 9 with 50 principal components, based on a cross-validation tuning process [34].

In the evaluation stage, we use precision, recall, and F_1 score to assess performance. For a specialty s, the true positive count $TP(s)$ is the number of users with the specialty s that are correctly classified. The false positive count $FP(s)$ is the number of users with a specialty other than s that are classified as s. The false negative $FN(s)$ count is the number of users with the specialty s that are wrongly classified. The precision P for a specialty s is computed as $\frac{TP(s)}{TP(s)+FP(s)}$ and the recall R is $\frac{TP(s)}{TP(s)+FN(s)}$. The precision of a classifier is the weighted average of precision for each specialty; the weight for a specialty s is the ratio of the number of users with s to the total number of users. The recall of a classifier is defined similarly. The F_1 score is the harmonic mean of the precision (P) and recall (R): $F_1 = \frac{2PR}{P+R}$. We use 5×2 cross-validation for evaluation with classifiers. In each of the 5 rounds, observations are split into two equal-sized sets A and

Table 6.4: Three inconsistent *de facto* diagnosis specialties are obtained by PathSelClus when the number of unseeded clusters δ is set to 3 on the general data set. They are represented by the top 10 most accessed diagnoses by all the users that are in each cluster respectively. None shows a consistent theme with respect to a specialty.

Other bacterial infections
Other non-traumatic joint disorders
Convulsions
Other upper respiratory disease
Phlebitis and thrombophlebitis
Malaise and fatigue
Other skin disorders
Fever of unknown origin
Cardiomyopathy
Substance-related disorders

Chronic kidney disease
Essential hypertension
Other cardiac dysrhythmias
Abdominal pain
Phlebitis and thrombophlebitis
Other fluid and electrolyte disorders
Anemia; unspecified
Pleurisy; pleural effusion
Acute renal failure
Hyperpotassemia

Abdominal pain
Other and unspecified lower respiratory disease
Nonspecific chest pain
Urinary tract infection; site not specified
Diabetes mellitus without complication
Essential hypertension
Other nervous system symptoms and disorders
Pneumonia; organism unspecified
Phlebitis and thrombophlebitis
Other and unspecified circulatory disease

B. Then a classifier is trained on A and tested on B and *vice versa*. After 5 rounds, the average of the 10 results is reported.

6.5.2 Results for PathSelClus

In Section 6.4.2, we mentioned PathSelClus relies on the mapping between users and their specifically accessed diagnoses of EHRs in the fine-grained data set. Table 6.4 shows inconsistent *de facto* diagnosis specialties by PathSelClus when the number of unseeded clusters δ is set to 3. None exhibits a consistent theme with respect to a specialty and it remains the same when δ is set to other values.

One reason why PathSelClus leads to inconsistent themes is that the general data set does not contain the aforementioned fine-grained mapping information. As a consequence, all of the diagnoses that belong to patients can be mapped to users that access such patients. We observe that a patient can have multiple encounters, such as delivering a baby and returning several months later due to a infectious disease. Therefore, in the general data set, clustering users based on all the diagnoses of their accessed patients may not be accurate (as shown in Table 6.4).

On the fine-grained data set, PathSelClus discovers a specialty for *Breast Cancer* that does not have a corresponding code in HPTCS, as shown in Table 6.5 ($\delta = 3$). Setting δ between 1 to 4 generates this

Table 6.5: The *de facto* diagnosis specialty Breast Cancer is discovered by PathSelClus. It is represented by the top 10 most accessed diagnoses by all the users that are associated with the Breast Cancer specialty.

Lump or mass in breast
Diffuse cystic mastopathy
Galactorrhea not associated with childbirth
Benign neoplasm of breast
Unspecified breast disorder
Abnormal mammogram, unspecified
Malignant neoplasm of upper-inner quadrant of female breast
Benign neoplasm of lymph nodes
Personal history of malignant neoplasm of breast
Other sign and symptom in breast

discovery although a larger value of δ makes the discovery less clear. In the fine-grained data set, 35 users are found to be associated with the Breast Cancer specialty.

Table 6.7 summarizes the average accuracy of multi-class classification on the fine-grained data set under 5×2 cross-validation. Users with the *de facto* Breast Cancer specialty discovered by PathSelClus are in one class; users with core NPI taxonomy codes as discussed in Section 6.3 are in the 12 distinct core classes. The F_1 score of the discovered *de facto* Breast Cancer specialty is significantly higher than that of mean of 12 core classes under all the four classifiers (paired t-test with $p < 0.05$).

6.5.3 Results for LDA

With a larger general data set, LDA confirms the discovery of Breast Cancer by PathSelClus and suggests another *de facto* diagnosis specialty for *Obesity* as shown in Table 6.6. The Breast Cancer and Obesity specialties are found to be associated with 68 and 20 users, respectively.

Tables 6.8 and 6.9 summarize the average accuracy of multi-class classification on the general data set under 5×2 cross-validation for the two discovered specialties. The F_1 score of the discovered *de facto* Breast Cancer specialty by LDA is also significantly higher than that of mean of 12 core classes under all the four classifiers, confirming the finding from PathSelClus (paired t-test with $p < 0.05$). The result for Obesity is similar except that PCA-KNN is not statistically significantly better than the other classifiers.

6.6 Related Work

The discovery of *de facto* diagnosis specialties is critical to managing health care institutions and allocating resources to clinicians. This work shows that such discovery is possible and that existing vocabularies may be insufficient or incomplete. To date, there has been little investigation into automated learning for the *de*

Table 6.6: *De facto* diagnosis specialties Breast Cancer and Obesity are discovered by LDA. They are represented by 10 most probable diagnoses respectively as an output of LDA.

Personal history of malignant neoplasm of breast	Obesity, unspecified
Lump or mass in breast	Morbid obesity
Abnormal mammogram, unspecified	Obstructive sleep apnea
Other specified aftercare following surgery	Unspecified sleep apnea
Other sign and symptom in breast	Hypersomnia with sleep apnea, unspecified
Carcinoma in situ of breast	Paralysis agitans
Family history of malignant neoplasm of breast	Hip joint replacement by other means
Other specified disorder of breast	Edema
Benign neoplasm of breast	Other dyspnea and respiratory abnormality
Acquired absence of breast and nipple	Body Mass Index 4

facto diagnosis specialty discovery; however, we wish to note that the approaches introduced in this work are related to those that have been developed for health care role prediction and access control management. Here, we take a moment to review relevant work in such areas.

A driver behind inferring medical specialties is the analysis of audit logs for security and privacy purposes [21, 23, 126]. This is feasible because EHRs and their audit logs encode valuable interactions between users and patients [146]. Users have roles in the health care institutions. If these roles are not respected by the online activities of the users, there may be an evidence of a security or privacy violation. An early study on this theme examined the idea of examining accesses to patient records to determine the position of an employee [188]. This work used a Naïve Bayes classifier and had generally poor performance on many positions, often because such positions could not easily be characterized in terms of the chosen attributes. Moreover, Experience Based Access Management envisioned such studies as part of a general effort to understand roles by exploiting information about institutional activities through the study of audit logs [56]. Another study in this direction sought to infer new roles from ways in which employees acted in their positions by iteratively revising existing positions based on experiences [187].

The problem of determining which departments are responsible for treating a given diagnosis was addressed by studies on Explanation-Based Auditing System (EBAS) [44, 43]. They are similar to our problem of identifying an employee's specialty. In these studies the auditing system utilizes the access patterns of departments to determine diagnosis responsibility information in two ways: by analyzing (i) how frequent a department accesses patients with the diagnosis, and (ii) how focused the department is at treating the given diagnosis. For instance, EBAS could use this approach to determine that the Oncology Department is responsible for chemotherapy patients, while the Central Staffing Nursing Department is not. The random topic access model (RTAM) [57] went beyond approaches based on conditional probabilities to work with topic models that characterize the common activities of employees in certain positions in the hospital. The evaluation of our work can be seen as merging ideas from EBAS and RTAM to explore when a *de facto*

diagnosis specialty can be described with a classifier. An advantage of our work comparing with the other recent work on inappropriate EHR access detection [108, 110, 123] is that our work outputs *de facto* diagnosis specialty information even for those that lack codes from the HPTCS. It has been known that the *de facto* diagnosis specialty information is useful in convincing patients into trusting a provider for using their EHRs [22, 161, 84].

6.7 Conclusion

Medical specialties are important but may be lacking or inaccurate in part because there is no official code to express them. We first proposed a novel and challenging *de facto* diagnosis specialty discovery problem under a general discovery–evaluation framework. Under this framework, we then employed a semi-supervised learning model on a fine-grained data set and an unsupervised learning model on a larger general data set for discovery; we further employed four supervised learning models for evaluation. Finally, we experimented on one year of diagnosis histories from a major medical center. The semi-supervised learning model discovered a *de facto* diagnosis specialty for Breast Cancer on the fine-grained data set; the unsupervised learning model confirmed this discovery and suggested a new *de facto* diagnosis specialty for Obesity on the larger general data set. The evaluation results reinforced that these two specialties can be recognized accurately by classifiers in comparison with 12 common diagnosis specialties defined by the Health Care Provider Taxonomy Code Set.

Table 6.7: Average accuracy of multi-class classification on the fine-grained data set under 5×2 cross-validation (in percent). Users with the *de facto* Breast Cancer specialty discovered by PathSelClus are in one class; users with core NPI taxonomy codes are in 12 distinct core classes. The boldfaced result with the superscript † denotes that, the F_1 score of the discovered *de facto* Breast Cancer specialty is significantly higher than that of mean of 12 core classes (paired t-test with $p < 0.05$).

Specialty	Decision Trees			Random Forests		
	P	**R**	**F_1**	**P**	**R**	**F_1**
Breast Cancer	86.67	57.14	**68.87**†	89.13	64.16	**74.61**†
Mean of 12 Core Classes	67.37	58.07	62.38	72.08	67.36	69.64
Urology	70.59	60.00	64.86	68.42	65.00	66.67
Neurology	71.05	57.45	63.53	71.05	57.45	63.53
Pulmonary Disease	100.00	54.17	70.27	93.33	58.33	71.79
Orthopaedic Surgery	93.33	48.28	63.64	93.33	48.28	63.64
Neonatal-Perinatal Medicine	87.50	25.00	38.89	89.43	89.29	85.36
Gastroenterology	67.86	50.00	57.58	69.23	47.37	56.25
Obstetrics & Gynecology	42.23	97.25	58.89	49.03	94.50	64.56
Neurological Surgery	100.00	35.00	51.85	100.00	35.00	51.85
Ophthalmology	73.91	40.48	52.31	90.04	71.43	79.66
Cardiovascular Disease	63.93	61.90	62.90	62.12	65.08	63.57
Infectious Disease	79.17	73.08	76.00	79.17	73.08	76.00
Dermatology	78.95	39.47	52.63	78.95	39.47	52.63

Specialty	PCA-KNN			SVM		
	P	**R**	**F_1**	**P**	**R**	**F_1**
Breast Cancer	77.00	79.09	**78.03**†	92.50	93.11	**92.80**†
Mean of 12 Core Classes	72.30	74.02	73.15	89.30	86.72	87.99
Urology	81.82	90.00	85.71	100.00	95.00	97.44
Neurology	65.57	85.11	74.07	81.48	93.62	87.13
Pulmonary Disease	71.43	83.33	76.92	95.83	95.83	95.83
Orthopaedic Surgery	69.70	79.31	74.19	100.00	89.66	94.55
Neonatal-Perinatal Medicine	92.59	89.29	90.91	96.15	89.29	92.59
Gastroenterology	69.23	94.74	80.00	95.00	100.00	97.44
Obstetrics & Gynecology	87.18	93.58	90.27	98.99	89.91	94.23
Neurological Surgery	33.33	5.00	8.70	100.00	35.00	51.85
Ophthalmology	80.56	69.05	74.36	54.67	97.62	70.09
Cardiovascular Disease	71.95	93.65	81.38	96.83	96.83	96.83
Infectious Disease	63.64	53.85	58.33	89.29	96.15	92.59
Dermatology	71.43	52.63	60.61	100.00	68.42	81.25

P: Precision; R: Recall; F_1: F_1 Score

Table 6.8: Average accuracy of multi-class classification on the general data set under 5×2 cross-validation (in percent). Users with the *de facto* Breast Cancer specialty discovered by LDA are in one class; users with core NPI taxonomy codes are in the 12 distinct core classes. The boldfaced result with the superscript † denotes that, the F_1 score of the discovered *de facto* Breast Cancer specialty is significantly higher than that of mean of 12 core classes (paired t-test with $p < 0.05$).

Specialty	Decision Trees			Random Forests		
	P	**R**	**F_1**	**P**	**R**	**F_1**
Breast Cancer	95.12	57.35	**71.56**†	91.11	60.29	**72.57**†
Mean of 12 Core Classes	66.42	53.21	59.08	71.28	63.18	66.99
Urology	75.00	45.00	56.25	75.00	45.00	56.25
Neurology	65.52	40.43	50.00	64.52	42.55	51.28
Pulmonary Disease	87.50	58.33	70.00	87.50	58.33	70.00
Orthopaedic Surgery	76.92	34.48	47.62	89.43	79.31	84.07
Neonatal-Perinatal Medicine	100.00	14.29	25.00	100.00	82.29	90.28
Gastroenterology	65.38	44.74	53.12	65.38	44.74	53.12
Obstetrics & Gynecology	55.80	70.64	62.35	57.03	66.97	61.60
Neurological Surgery	100.00	35.00	51.85	88.89	40.00	55.17
Ophthalmology	23.12	95.24	37.21	69.36	95.24	80.27
Cardiovascular Disease	64.29	57.14	60.50	64.15	53.97	58.62
Infectious Disease	76.19	61.54	68.09	73.91	65.38	69.39
Dermatology	75.00	31.58	44.44	82.11	68.42	74.64

Specialty	PCA-KNN			SVM		
	P	**R**	**F_1**	**P**	**R**	**F_1**
Breast Cancer	82.58	80.88	**81.69**†	96.92	92.65	**94.74**†
Mean of 12 Core Classes	75.45	76.21	75.83	90.84	88.93	89.88
Urology	78.26	90.00	83.72	100.00	95.00	97.44
Neurology	72.73	85.11	78.43	80.36	95.74	87.38
Pulmonary Disease	70.37	79.17	74.51	95.65	91.67	93.62
Orthopaedic Surgery	68.57	82.76	75.00	100.00	93.10	96.43
Neonatal-Perinatal Medicine	92.59	89.29	90.91	96.15	89.29	92.59
Gastroenterology	75.00	94.74	83.72	97.44	100.00	98.70
Obstetrics & Gynecology	90.83	90.83	90.83	99.02	92.66	95.73
Neurological Surgery	50.00	10.00	16.67	100.00	20.00	33.33
Ophthalmology	86.11	73.81	79.49	100.00	88.10	93.67
Cardiovascular Disease	76.62	93.65	84.29	96.77	95.24	96.00
Infectious Disease	52.00	50.00	50.98	96.00	92.31	94.12
Dermatology	79.17	50.00	61.29	55.22	97.37	70.48

P: Precision; R: Recall; F_1: F_1 Score

Table 6.9: Average accuracy of multi-class classification on the general data set under 5×2 cross-validation (in percent). Users with the *de facto* Obesity specialty discovered by LDA are in one class; users with core NPI taxonomy codes are in 12 distinct core classes. The boldfaced result with the superscript † denotes that, the F_1 score of the discovered *de facto* Obesity specialty is significantly higher than that of mean of 12 core classes (paired t-test with $p < 0.05$).

Specialty	Decision Trees			Random Forests		
	P	**R**	**F_1**	**P**	**R**	**F_1**
Obesity	100.00	40.41	**57.56**†	83.22	56.98	**67.64**†
Mean of 12 Core Classes	63.18	45.62	52.98	75.62	53.51	62.68
Urology	100.00	50.00	66.67	100.00	55.00	70.97
Neurology	85.71	38.30	52.94	70.71	57.45	63.39
Pulmonary Disease	100.00	45.83	62.86	100.00	45.83	62.86
Orthopaedic Surgery	100.00	3.45	6.67	82.15	37.93	51.90
Neonatal-Perinatal Medicine	100.00	39.29	56.41	100.00	39.29	56.41
Gastroenterology	82.35	36.84	50.91	82.35	36.84	50.91
Obstetrics & Gynecology	30.59	99.08	46.75	38.12	93.58	54.17
Neurological Surgery	100.00	40.00	57.14	100.00	50.00	66.67
Ophthalmology	100.00	4.76	9.09	100.00	4.76	9.09
Cardiovascular Disease	76.92	63.49	69.57	76.92	63.49	69.57
Infectious Disease	78.95	57.69	66.67	78.95	57.69	66.67
Dermatology	100.00	2.63	5.13	87.21	39.47	54.34

Specialty	PCA-KNN			SVM		
	P	**R**	**F_1**	**P**	**R**	**F_1**
Obesity	75.01	82.12	78.40	92.85	94.01	**93.43**†
Mean of 12 Core Classes	77.12	80.36	78.70	90.23	89.19	89.70
Urology	86.36	95.00	90.48	100.00	95.00	97.44
Neurology	63.49	85.11	72.73	82.14	97.87	89.32
Pulmonary Disease	71.43	83.33	76.92	100.00	87.50	93.33
Orthopaedic Surgery	62.16	79.31	69.70	96.43	93.10	94.74
Neonatal-Perinatal Medicine	89.29	89.29	89.29	100.00	89.29	94.34
Gastroenterology	76.09	92.11	83.33	94.87	97.37	96.10
Obstetrics & Gynecology	94.50	94.50	94.50	100.00	94.50	97.17
Neurological Surgery	33.33	5.00	8.70	100.00	40.00	57.14
Ophthalmology	88.24	71.43	78.95	86.96	95.24	90.91
Cardiovascular Disease	75.64	93.65	83.69	95.31	96.83	96.06
Infectious Disease	66.67	53.85	59.57	88.89	92.31	90.57
Dermatology	66.67	52.63	58.82	67.27	97.37	79.57

P: Precision; R: Recall; F_1: F_1 Score

Chapter 7

Privacy Risk in Anonymized Big Data Traces of Human Activities

In the end, this chapter studies studies the privacy risk in anonymized big data traces of human activities that are released for external intention analysis research. As an example, *t.qq.com* released its anonymized users' profile, social interaction, and recommendation log data in KDD Cup 2012 to call for recommendation algorithms. The goal is to improve the prediction accuracy for users' online networking intentions on *t.qq.com*. Specifically, the online networking intention prediction task involves predicting whether or not a user will follow an item (person, organization, or group) that has been recommended to the user. Since the entities (users and so on) and edges (links among entities) are of multiple types, the released social network is a *heterogeneous information network*. Prior work has shown how privacy can be compromised in homogeneous information networks by the use of specific types of graph patterns. We show how the extra information derived from heterogeneity can be used to relax these assumptions. To characterize and demonstrate this added threat, we formally define privacy risk in an anonymized heterogeneous information network to identify the vulnerability in the possible way such data are released, and further present a new de-anonymization attack that exploits the vulnerability. Our attack successfully de-anonymized most individuals involved in the data—for an anonymized 1,000-user *t.qq.com* network of density 0.01, the attack precision is over 90% with a 2.3-million-user auxiliary network.

7.1 Introduction

The world is getting more inter-connected. Tons of social network data are generated through people's interactions, and different entities are linked across multiple relations, forming a gigantic information-rich, inter-related and multi-typed *heterogeneous information network* [59]. Is there any risk in the current efforts to avoid privacy intrusion upon the anonymized copy of a heterogeneous information network? We start with a motivating example.

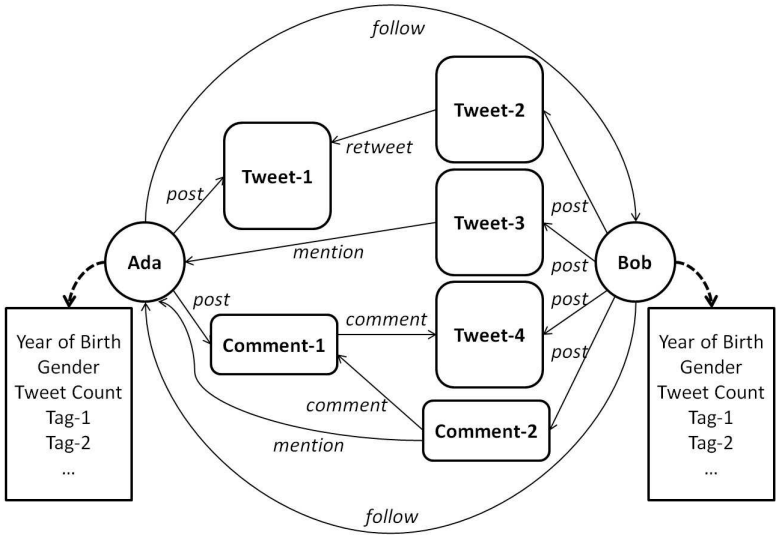


Figure 7.1: The heterogeneous information network in t.qq

7.1.1 Motivating Example

Various data sets containing *micro-data*, that is, information about specific individuals, have been released for different research purposes or industry applications [115]. Some data sets contain individual profiles, preferences, or transactions, which many people consider sensitive or *private*. In the recent KDD Cup 2012, *t.qq.com* (a popular microblogging site, hereinafter referred to as *t.qq*) released its 2.3 million users' profile, social interaction, and recommendation preference log data to call for more efficient recommendation algorithms [1]. In a microblogging site like *t.qq* as depicted in Figure 7.1, entities (nodes) correspond to *users*, *tweets* or *comments*, and edges correspond to different types of links (*post*, *mention*, *retweet*, *comment*, and *follow*) among them[1]. Since both nodes and links are of multiple types, such a social network is essentially a heterogeneous information network [149]. Besides identifying information such as *user ID* which has been anonymized by randomly assigned strings, some other attributes are also replaced with meaningless IDs, such as *user tags*.

In the released anonymized *target data set*, consider an adversary that is interested in breaching privacy of some selected target users based on their preferences. The preference can be inferred from the target users' recommendation preference (acceptance/rejection) log included in the target data set. This information is sensitive and not accessible on the *t.qq* site (the rejection log cannot be inferred from the site). Suppose the adversary obtains the non-anonymized *auxiliary data set* from *t.qq* exactly containing the users from the same time-synchronized target data set. To *de-anonymize* the users of interests in the target data

[1]The terms *edge* and *link* are used interchangeably in this work, while the term *entity* is preferred over *node* here to reflect more realistic scenarios where each node contains multiple attributes rather than a single identifier in the settings of a heterogeneous information network.

set, the adversary has to match the meaningless user IDs in the target data set with the real user names in the auxiliary data set. Given the rich information available in the heterogeneous information network as demonstrated in Figure 7.1, suppose the adversary locks his target on an anonymized user (say, *A3H*) in the target data set who accepted the "follow Citibank" recommendation but rejected all other bank recommendations. The adversary may search in the auxiliary data set by specifying *A3H*'s entity profile (*A3H*'s year of birth, hereinafter referred to as yob: 1980, gender: male, *etc.*) combined with *A3H*'s multiple social links (mention, retweet, comment, follow) and profile information of its neighbor entity to whom the target user connects via these links—*A3H* gave 15 comments to an anonymized female user *F8P* born in 1985 and retweeted an anonymized male user *M7R* 10 times that is born in 1970. If Ada in the non-anonymized auxiliary data set is the only one that satisfies the matching—Ada has both the same profile information as *A3H* and Ada has the same social interactions with the other users of the same gender and age as those of *F8P* and *M7R* correspondingly; thus, the adversary successfully de-anonymizes *A3H* by establishing a *unique matching* between it in the target data set and the real user Ada in the auxiliary data set. Now the adversary knows Ada probably has a Citibank account or is interested in applying for it. The leak of such private information may allow scammers to spam Ada with phishing URLs camouflaged with the Citibank online-banking interface. In fact, 8% of some sampled 25 million URLs posted to microblogging sites point to phishing, malware, and scams [54].

Therefore, there is *privacy risk* in an anonymized heterogeneous information network if such unique matchings can be easily established. Users in a network of high privacy risk that can be easily de-anonymized may be vulnerable to external threats. In this work, we experimentally substantiate adversaries can exploit the privacy risk to de-anonymize over 90% users in a 1,000-user *t.qq* network of density 0.01 from a 2,320,895-user auxiliary network.

7.1.2 Limitations of k-Anonymity

To formalize privacy risk observed in Section 7.1.1, directly using the existing metric seems possible at first thought. A data set is said to be k-anonymous if on the minimal set of attributes in the table that can be joined with external information to de-anonymize individual records, each record is indistinguishable from at least $k-1$ other records within the same data set [156]. The larger the value of k, the better the privacy is preserved.

Consider target data set T_{1000} that satisfies 1000-anonymity and another target data set T_2 that satisfies 2-anonymity, together with their original non-anonymized counterparts. Imagine a new tuple t^* is created and inserted into both T_{1000} and T_2. After anonymization processes still no any other tuple in either data

set has the same value of t^*, and the new data sets are T^*_{1000} and T^*_2 respectively. Both T^*_{1000} and T^*_2 are now 1-anonymity simply because of the injection of t^*—both T^*_{1000} and T^*_2 are same vulnerable in terms of the same k-anonymity. Suppose a selective adversary is not interested in de-anonymizing t^*, then the remaining T^*_{1000} of 1000-anonymity seems much less vulnerable than the remaining T^*_2 of 2-anonymity, which may be misled by the same 1-anonymity.

Due to limitations of k-anonymity in differentiating individuals in the same target data set, it is not suitable to formalize privacy risk in a more general scenario where adversaries may not be equally interested in de-anonymizing all users. We define privacy risk in a more general sense, and prove it can be very high in the anonymized heterogeneous information network.

7.1.3 New Settings, New Threats

Social media are getting popular with more and more functionalities. As shown in Section 7.1.1, *t.qq* allows its over 500 million users to connect with one another in different ways such as follow, mention, retweet, and comment. The growing multi-typed heterogeneous information networks out of the growing social media functionalities may render the existing homogeneous information network anonymization algorithms no more effective.

Existing de-anonymization attacks on social networks made several assumptions, such as both target and auxiliary graphs are large-scale so random graphs or non-trivial cliques can be re-identified from both graphs [7, 116]. It should be highlighted that, in the new settings of a heterogeneous information network, if new attacks are feasible while relaxing these assumptions, such attacks must be addressed in the proposal of all relevant anonymization algorithms.

7.1.4 Our Contributions

In this work we make three unique contributions. First, we propose a definition of privacy risk tuned to the concerns of heterogenous information networks. In particular, this definition considers a more general situation where adversaries may not be equally interested in compromising all users' privacy. We show that the privacy risk can be high in an anonymized heterogeneous information network, and can be exploited in practice.

Second, we present a de-anonymization algorithm against heterogeneous information networks which exploits the identified privacy risk without requiring creating new accounts or relying on easily-detectable graph structures in a large-scale network. While central in illuminating the privacy issue for a heteroge-

neous information network, we also expect our algorithm to be applied to de-anonymizing a homogeneous information network (with slight performance degradation).

Our third contribution is a practical evaluation of the KDD Cup 2012 *t.qq* anonymized data set, which contains 2.3 million users and over 60 million multiple types of social links among them. To demonstrate the effectiveness of the de-anonymization algorithm, we apply the state-of-the-art graph anonymization algorithms to the *t.qq* data set, which were claimed effective by their designers for defending graph structural attacks. The experiments show that our algorithm is able to beat the investigated graph anonymization algorithms in the settings of a heterogeneous information network even without knowledge of the specific anonymization technique in use. It undermines the notion of "security by obscurity" for privacy preservation: ignorance of the anonymization does not prevent an adversary from de-anonymizing successfully.

7.2 Related Work

Simply replacing sensitive information with random strings cannot guarantee privacy and how to release data for different research purposes or industry applications without leaking any privacy information has been an interesting problem.

7.2.1 Relational Data Anonymization

A major category of privacy attacks on relational data is to de-anonymize individuals by joining a released table containing sensitive information with some external tables modeling the auxiliary data set of attackers. To mitigate this type of attacks, k-anonymity was proposed [156]. Further enhanced techniques include l-diversity [105] and t-closeness [88].

Narayanan and Shmatikov proposed de-anonymization attacks against high-dimensional micro-data and showed success in Netflix Prize data set [115]. They pointed out micro-data are characterized by high dimensionality and sparsity. A recent study by Narayanan *et al.* further demonstrated the feasibility of internet-scale author identification via linguistic stylometry [114]. However, all the aforementioned studies assume that an adversary utilizes *attribute information* of micro-data and can deal with relational data only.

7.2.2 Graph Structural Attacks

In a large-scale social network, it is hard to observe non-trivial random subgraphs or cliques [122]. Hence they easily stand out if they exist. Backstrom *et al.* discussed active attacks where adversaries create users and establish connections randomly among them and attach such random subgraphs ("sybil nodes") into

the target nodes in the auxiliary graph data [7]. Since such random subgraphs can be easily detected from the anonymized counterpart of the original data, the target nodes connected to the sybil nodes are then de-anonymized by consulting the original auxiliary graph. Narayanan and Shmatikov pointed out the main drawback of this active attack is that, creating accounts, links among themselves and links to target nodes, is not feasible on a large-scale [116]. They designed an attack propagating the de-anonymization process via neighbor structure from the initial precisely-matched "seed nodes". Hence success of this attack heavily depends on if such seed nodes can be detected precisely; thus, seed nodes must stand out easily both in the target and auxiliary data set. So non-trivial cliques are chosen [116]. Since there is no guarantee that the released anonymized network is always large, this attack is not always successful because non-trivial cliques cannot always be detectable.

7.2.3 Graph Data Anonymization

For graph-based social network data, the degree of nodes in a graph can reveal the identities of individuals. Liu and Terzi studied a specific graph-anonymization problem and called a graph k-degree anonymous if for every node v, there exist at least $k-1$ other nodes in the graph with the same degree [99]. This definition of anonymity prevents de-anonymization of individuals by adversaries with a background knowledge of the degree of certain nodes.

Zhou and Pei identified a structural *neighborhood attack* and tackled it by proposing k-neighborhood anonymization [194]. They assumed an adversary may know the neighbors of the target nodes and their inter-connections. The privacy preservation goal is to protect neighborhood attacks which use neighbor structure matching to de-anonymize nodes. For a social network, suppose an adversary knows the neighbor structure for a node. If such neighbor structure has at least k isomorphic copies in the anonymized social network, then the node can be de-anonymized in the target data set with confidence at most $1/k$ [195]. Due to its heavy isomorphism testing computation, a limitation of this attack is only distance-1 neighbors can be evaluated effectively.

Zou *et at.* assumed an attacking model where an adversary can know any subgraph that contains the targeted individual and proposed k-automorphic anonymity that the graph must has $k-1$ non-trivial automorphism and no node is mapped to itself under the $k-1$ non-trivial automorphism [198]. Wu *et al.* proposed a similar k-symmetry [170].

Cheng *et al.* identified that k-automorphism approach is insufficient for protecting link privacy and proposed the k-security anonymity [29]. In their approach, an anonymized graph satisfies k-security if for any two target individuals and any subgraphs containing either individual, the adversary cannot determine

either whether a node that is linked to either target individual (NodeInfo Security) or whether both target individuals are linked by a path of a certain length (LinkInfo Security), with probability higher than $1/k$.

Although these recent graph data anonymization algorithms can be applied to social network data against graph structural attacks in Section 7.2.2, their applicability has not been demonstrated in the more challenging settings of a heterogeneous information network. Our evaluation in Section 7.6 shows that these graph data anonymization algorithms are not effective to preserve privacy of an anonymized heterogeneous information network.

7.3 Heterogeneous Information Network Settings

In this section, we formalize the general anonymized heterogeneous information network settings that are frequently discussed in the remaining of the chapter and illustrate them with the motivating example discussed in Section 7.1.1.

Definition 7.3.1 *The **information network** is a directed graph $G = (V, E)$ with an entity type mapping function $\tau : V \to \mathcal{E}$ and a link type mapping function $\phi : E \to \mathcal{L}$, where each entity $v \in V$ belongs to one particular entity type $\tau(v) \in \mathcal{E}$, and each edge $e \in E$ belongs to a particular link type $\phi(e) \in \mathcal{L}$. If two edges belong to the same link type, they must share the same starting and ending entity types.*

Definition 7.3.2 *The **heterogeneous information network** is an information network where $|\mathcal{E}| > 1$ or $|\mathcal{L}| > 1$.*

A sample heterogeneous information network for the *t.qq* data set is depicted in Figure 7.1. Given a complicated heterogeneous information network, it is necessary to provide its meta level (*i.e.*, schema-level) description for better understanding the network, and *network schema* is to describe the meta structure of a network.

Definition 7.3.3 *The **network schema**, denoted as $T_G = (\mathcal{E}, \mathcal{L})$, is a meta template for a heterogeneous information network $G = (V, E)$ with the entity type mapping $\tau : V \to \mathcal{E}$ and the link mapping $\phi : E \to \mathcal{L}$, which is a directed graph defined over entity types $\mathcal{E}$, with edges as links from $\mathcal{L}$.*

Figure 7.2 shows the network schema for the heterogeneous information network in Figure 7.1. In practice data publishers may not release information about all the entities and links in the original network schema while links among the same entity type (also the target entity type of adversaries' interests) are generally available either directly or indirectly via summarization over different entity types [1]. In view of this,

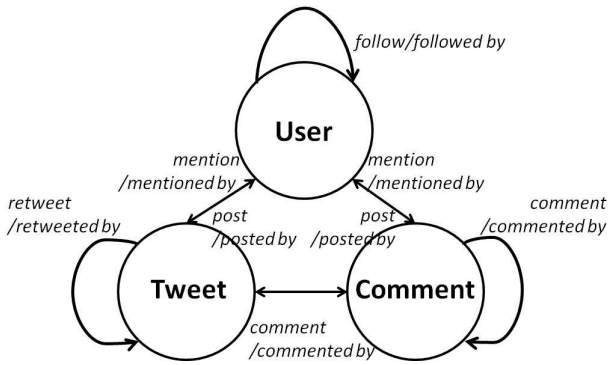


Figure 7.2: The corresponding network schema for the heterogeneous information network in Figure 7.1

although we believe providing richer information about multiple types of entities could further facilitate de-anonymization, in this work, we consider a more challenging and practical scenario where data publishers only provide limited information about how the same type of entity (*i.e.*, *target entity type* $\mathcal{E}^*$) can be linked via different types of links or over different types of entities. Thus, a simplified network schema is needed such that it reflects only the relationships over the target entity type.

Definition 7.3.4 *The* ***target meta paths (target network schema links)*** $\mathcal{P}(\mathcal{E}^*)$*, are paths defined on the graph of network schema* $T_G = (\mathcal{E}, \mathcal{L})$*, denoted by* $\mathcal{E}^* \xrightarrow{\mathcal{L}_1} \mathcal{E}_1 \xrightarrow{\mathcal{L}_2} ... \xrightarrow{\mathcal{L}_n} \mathcal{E}^*$.

Definition 7.3.5 *The* ***target network schema*** $T_G^* = (\mathcal{E}^*, \mathcal{L}^*)$ *is projected from* $T_G = (\mathcal{E}, \mathcal{L})$ *where* $\mathcal{L}^*$ *are* reproduced *or* short-circuited *from target meta paths* $\mathcal{P}(\mathcal{E}^*)$ *and target entity type* $\mathcal{E}^*$.

To illustrate, we take the released target *t.qq* data set as an example. This anonymized data set contains the following files and attributes (anonymized attributes are marked with underlines):

- *recommendation preference data*: user ID($\mathcal{A}$), recommended item ID($\mathcal{R}$), result (whether $\mathcal{A}$ likes $\mathcal{R}$)
- *user profile data*: user ID, yob, gender, tweet count (no. of tweets), tag IDs
- *user mention data*: user ID($\mathcal{A}$), user ID($\mathcal{B}$), the number of times $\mathcal{A}$ mentioned $\mathcal{B}$ either in $\mathcal{A}$'s tweets or comments (mention strength)
- *user retweet data*: user ID($\mathcal{A}$), user ID($\mathcal{B}$), the number of times $\mathcal{A}$ retweeted $\mathcal{B}$'s tweets (retweet strength)
- *user comment data*: user ID($\mathcal{A}$), user ID($\mathcal{B}$), the number of times $\mathcal{A}$ commented $\mathcal{B}$ either in $\mathcal{B}$'s tweets or comments (comment strength)
- *user follow data*: user ID(follower), user ID(followee)

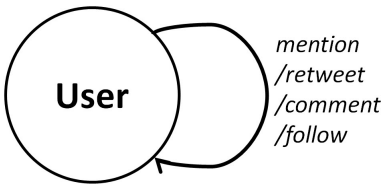


Figure 7.3: The target network schema for Figure 7.2

In the above data set, besides user entities' profile information, users' multiple social interactions are also available. Thus, the adversary can decide to project the original network schema in Figure 7.2 to only reflect relationships among his target user entity. Navigating the original network schema based on the above user mention, retweet, comment, and follow data, these target meta paths connecting users across different types of entities are possible:

- *user mention path*: $User \xrightarrow{post} Tweet \xrightarrow{mention} User$ or $User \xrightarrow{post} Comment \xrightarrow{mention} User$ (short-circuited feature: mention strength)
- *user retweet path*: $User \xrightarrow{post} Tweet \xrightarrow{retweet} Tweet \xrightarrow{posted\ by} User$ (short-circuited feature: retweet strength)
- *user comment path*: $User \xrightarrow{post} Comment \xrightarrow{comment} Tweet \xrightarrow{posted\ by} User$ or $User \xrightarrow{post} Comment \xrightarrow{comment} Comment \xrightarrow{posted\ by} User$ (short-circuited feature: comment strength)
- *user follow path*: $User \xrightarrow{follow} User$

The target meta paths allow the adversary to produce a new network schema by projecting the original network schema to a simplified one to only reflect particular few relationships over the target entity type. Specifically, the user mention, retweet and comment paths can be *short-circuited* to produce new links over users respectively while the user following path can be *reproduced* in the projection. It is also emphasized that, the target meta paths are able to greatly enrich the features (attributes) of the target entity by utilizing different *distances* of *neighbors* from the target entity along the specified meta paths. Specifically, target meta paths that are short-circuited across different types of entities and different types of links, may preserve the link heterogeneity information of the network by generating new *short-circuited feature (attribute)* and further enrich the features of the target entity. For instance, the short-circuited feature *mention strength* can be newly generated from the user mention path.

The target network schema for Figure 7.2 is shown in Figure 7.3. Since target meta paths may span across multiple types of entities, entity heterogeneity information is still preserved, although not fully, in target network schema only containing the target entity type.

Therefore, the de-anonymization problem in the settings of a heterogeneous information network can be formulated as follows. Detailed illustrations are provided in Section 7.5.

Definition 7.3.6 *The **de-anonymization problem in heterogeneous information network** is utilizing the background knowledge of the public graph $G = (V, E)$, the private graph $G' = (V', E')$, and the target network schema T_G^* to de-anonymize a target entity $v' \in V'$ by establishing matches between v' and a candidate set $C \subseteq V$ where the anonymized v''s counterpart $v \in C$. If $|C| = 1$ and the only element $v \in C$ is the correct counterpart of v', the de-anonymization is successful.*

7.4 Privacy Risk Analysis

Intuitively, privacy risk in a heterogeneous information network is the ease of formulating unique attribute-metapath-combined values as formalized in Section 7.3. Formal analysis is derived from the definition of privacy risk in general anonymized data sets.

7.4.1 Attribute-Metapath-Combined Values of Target Entities

Data publishers anonymize data through generalization, suppression, adding, deleting, switching edges or nodes [155][195]. Naturally, such modifications cause information loss and for a certain privacy preservation goal they should be minimized to ensure the anonymized data still satisfy the need for how they are expected to be used, *i.e.*, the need for *utility*. Generally, a certain level of utility has to be preserved for the anonymized *t.qq* data set in order to design effective and reliable recommendation algorithms; thus, an adversary is expected to be able to compromise some sacrificed privacy due to the natural tradeoff between utility and privacy preservation [195]. In the *t.qq* data set case, the utility is preserved in the sense that, some attribute values of user entities and most of the social interactions among different user entities are preserved (non-anonymized) as in the available target data set descriptions in Section 7.3 (*e.g.*, non-anonymized attributes are not underlined).

Based on the target network schema in Figure 7.3, Figure 7.4 describes an example of how user entities are directly inter-connected via part of different types of links in the *t.qq* data set. Here m, r, c, f stands for *mention*, *retweet*, *comment*, *follow* links in the target network schema shown in Figure 7.3.

As mentioned in Section 7.3, target meta paths that are short-circuited across different types of entities and different types of links preserve the link heterogeneity information of the information network and further enrich the features of the target entity. It should be noted that, following the user mention path identified in Section 7.3, $5m$ in Figure 7.4 from *A1X* to *U2V* indicates a new numerical feature (attribute)

short-circuited from the user mention path—the mention strength from *A1X* to *U2V* in the target data set of value 5 either through the tweet entity or comment entity. Thus, multiple meta-paths inject richer heterogeneity information for target entities in the settings of a heterogeneity information network.

If target user entities in the target data set can form unique *attribute-metapath-combined values* across the entire network, these users can be de-anonymized from the auxiliary data set by establishing unique matches and the data set is not secure. To analyze the privacy risk of a heterogeneous information network, which can be intuitively considered similar to the ease of formulating unique attribute-metapath-combined values, one way is to expand the attribute dimensions of micro-data by navigating from user entities to their neighbors, neighbors' neighbors, and so on, via their multiple types of target meta paths.

With the assumption made in Section 7.1.1 that the target and auxiliary data sets are time-synchronized counterparts, take *A1X* in Figure 7.4 as an example. Without utilizing meta paths and only utilizing profile attribute information, the features of *A1X* are:

- Max. Distance-0: *yob, gender, ...*

After utilizing his immediate distance-1 neighbors along target meta paths, the features of *A1X* are expanded to (here "5-time-mentionee" means a mentionee mentioned 5 times by the target entity, *i.e.*, mention strength = 5):

- Max. Distance-1: *yob, gender, ..., 5-time-mentionee (U2V)'s yob, 5-time-mentionee's gender, ..., 15-time-mentionee (P3M)'s yob, 15-time-mentionee's gender, ..., 10-time-retweetee (E4G)'s yob, 10-time-retweetee's gender, ...*

Further utilizing his distance-2 neighbors (neighbors of distance-2 along target meta paths from *A1X*), the features of *A1X* are further expanded to:

- Max. Distance-2: *yob, gender, ..., 5-time-mentionee's yob, 5-time-mentionee's gender, ..., 15-time-mentionee's yob, 15-time-mentionee's gender, ..., 10-time-retweetee's yob, 10-time-retweetee's gender, ..., 10-time-retweetee's followee (B8R)'s yob, 10-time-retweetee's followee's gender, 10-time-retweetee's 1-time-mentionee (Y9Z)'s yob, 10-time-retweetee's 1-time-mentionee's gender, ...*

Consistent with the idea by Narayanan and Shmatikov that large dimensions of micro-data give rise to risks of privacy [115], the expansion of dimensions by propagating via multiple types of target meta paths seems to increase the possibility for a user entity to form a unique attribute-metapath-combined value under all the expanded features across the entire data set, which can be considered as privacy risk. In the remaining of this section, we formally prove this intuition from the observations.

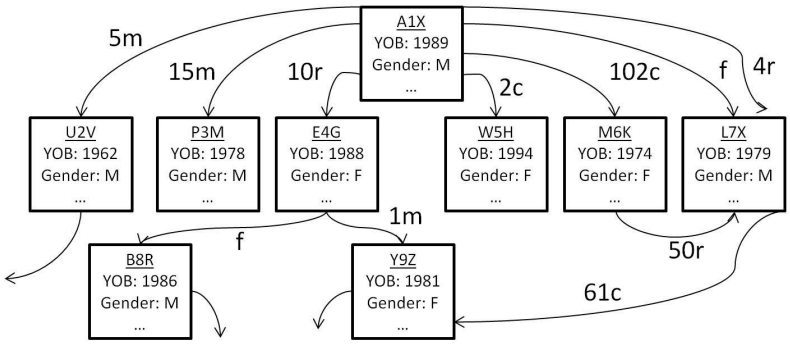


Figure 7.4: The neighbors of the target entity *A1X* are generated along target meta paths

7.4.2 Privacy Risk in General Anonymized Data Sets

Privacy Risk indicates risk that privacy of a given data set can be compromised—the higher privacy risk, the lower security and *vice versa*. Hence it might be tempting to directly adopt the notion of widely-used k-anonymity and simply reverse its value to obtain the measure of privacy risk. Here we state that, k-anonymity is not able to differentiate users from one another in terms of their different levels of security or privacy risk.

As discussed in Section 7.1.2, k-anonymity may be misleading in more general situations where adversaries may not be equally interested in compromising all users' privacy. To address its limitations, when quantifying risk of any user in any data set, we consider factors that influence privacy risk both socially and mathematically.

In real life, it is highly possible that an adversary is not equally interested in compromising everyone's privacy in a data set. As illustrated in Section 7.1.1, an adversary may be more motivated to de-anonymize an anonymized user who probably has a Citibank account. We denote the loss function of tuple t_i by $l(t_i)$, with values between 0 and 1. $l(t_i)$ can be considered as the potential loss of a user whose privacy is compromised given that this user does care about his loss of privacy. Therefore, in a social network, $l(t_i)$ is a certain user's privacy need because such need is positively correlated with the cost of privacy breach; hence, it is the *social factor* of a user's privacy risk.

Similar to the concept of k-anonymity, we make the same assumption that the target data set is an anonymized copy of the same auxiliary data set. In any given data set T, if there are $k(t_i) - 1$ other tuples of the same value of tuple t_i, the probability that each of these $k(t_i)$ tuples, say t_i, can be de-anonymized by random guessing with probability no higher than $1/k(t_i)$. Therefore, the higher value of $1/k(t_i)$, the higher possibility that the privacy of user t_i can be compromised—hence the higher privacy risk of the user t_i. The fraction $1/k(t_i)$ is the *mathematical factor*. Mathematical factor can be considered positively correlated with

the attack incentive as well: given the same social factor, the adversary is more motivated to de-anonymize the user with a higher mathematical factor because the potential attack precision is higher.

Combining both social and mathematical factors, we define the privacy risk of a tuple in a data set as follows.

Definition 7.4.1 *We define the* ***privacy risk*** $\mathfrak{R}(t_i)$ *of tuple* t_i *in data set* T *as follows:*

$$\mathfrak{R}(t_i) = \frac{l(t_i)}{k(t_i)},$$

where $k(t_i)$ *is the number of tuples in* T *with the same value of tuple* t_i*, and* $l(t_i)$ *is the loss function of tuple* t_i*.*

Averaging the risk $\mathfrak{R}(t_i)$ for each tuple t_i in data set T, the risk $\mathfrak{R}(T)$ for data set T is defined as follows.

Definition 7.4.2 *The* ***privacy risk*** $\mathfrak{R}(T)$ *of data set* T *is*

$$\mathfrak{R}(T) = \frac{\sum_{i=1}^{N} \mathfrak{R}(t_i)}{N},$$

where size N *is the number of tuples* t_i *in* T*.*

It is noted that the privacy risk value $\mathfrak{R}(T) \in [0, 1]$. Denoting by $\mathbb{C}(T)$ the *cardinality* of T—the number of distinct values, or distinct combined values under different attributes, describing each tuple t_i in T, we give the following lemma.

Lemma 7.4.3 *Given data set* T *with the cardinality* $\mathbb{C}(T)$*, for each tuple* t_i *in* T*, assuming the loss function is independent of* $1/k(t_i)$ *with mean value* μ*, the expected privacy risk*

$$\mathbb{E}(\mathfrak{R}(T)) = \frac{\mu \mathbb{C}(T)}{N}.$$

Proof. By Definition 7.4.1 and 7.4.2,

$$\mathfrak{R}(T) = \frac{\sum_{i=1}^{N} l(t_i)/k(t_i)}{N}.$$

Hence, we have

$$\begin{aligned}\mathbb{E}(\mathfrak{R}(T)) =& \frac{\sum_{i=1}^{N} \mathbb{E}(1/k(t_i))\mathbb{E}(l(t_i))}{N} \\ =& \frac{\sum_{i=1}^{N} \mu\mathbb{E}(1/k(t_i))}{N} \\ =& \frac{\mu\mathbb{E}(\sum_{i=1}^{N} 1/k(t_i))}{N} \\ =& \frac{\mu\mathbb{E}(\mathbb{C}(T))}{N} \\ =& \frac{\mu\mathbb{C}(T)}{N}.\end{aligned}$$

■

Lemma 7.4.3 provides an estimation of data set privacy risk in a relatively general sense. For instance, if the loss function for each tuple is a random number between 0 and 1 and independent of $1/k(t_i)$, the expected privacy risk of the data set is $\mathbb{C}(T)/(2N)$. Although it may be interesting to quantify the social factor in other ways, to guarantee the highest possible privacy need from all users has been considered, in the remaining analysis we focus on the mathematical factor and set the value of every loss function $l(t_i)$ to 1. Adversaries may still have varying attack incentives in terms of different mathematical factors as discussed earlier in this section.

Theorem 7.4.4 *The privacy risk $\mathfrak{R}(T)$ of data set T is*

$$\mathfrak{R}(T) = \frac{\mathbb{C}(T)}{N}, \quad \left(\mathfrak{R}(T) \in \left[\frac{1}{N}, 1\right]\right),$$

where in T, N is the number of tuples, and cardinality $\mathbb{C}(T)$ is the number of distinct (combined) attribute values describing tuples.

Proof. The proof can be completed by applying Lemma 7.4.3 and mathematical derivation with $l(t_i) = 1$. $\mathfrak{R}(T)$ is lowest when all the tuples are of the same value; in contrast, if every t_i has a unique value in T, $\mathfrak{R}(T) = 1$. ■

Back to the example of T_{1000} and T_2 in Section 7.1.2, suppose they are both of the same size 1000: T_{1000} has 1000 tuples of the same value while T_2 has 500 same-value tuple pairs and values from different pairs are distinct. By Definition 7.4.2, $\mathfrak{R}(T_{1000}) = 0.001$ and $\mathfrak{R}(T_2) = 0.5$ and the result is consistent with k-anonymity in terms of relative privacy risk. After inserting the unique tuple t^*, $\mathfrak{R}(T^*_{1000}) = 2/1001$ and $\mathfrak{R}(T^*_2) = 501/1001$, reasonably indicating T^*_{1000} is in general still much less vulnerable than T^*_2. It addresses

the identified limitations of k-anonymity when adversaries may not select some users to de-anonymize in the target data set.

7.4.3 Privacy Risk in Anonymized Heterogeneous Information Networks

Section 7.4.1 informally shows entity attribute dimensions grow fast when neighbors are utilized. It is highlighted that, rather than the exact value of privacy risk, it is the growth of privacy risk with respect to max. distances of utilized neighbors n that we focus on. Hence, given any anonymized data set, the number of tuples N is fixed as a constant. So Theorem 7.4.4 implies that privacy risk $\mathfrak{R}(T)$ is of the same order of growth as that of the cardinality $\mathbb{C}(T)$.

Theorem 7.4.5 *For power-law distribution of the user out-degree, the lower and upper bounds for the expected heterogeneous information network cardinality grows faster than double exponentially with respect to the max. distance of utilized neighbors.*

Proof. Given a network schema $T_G^* = (\mathcal{E}^*, \mathcal{L}^*)$ projected from its original schema $T_G = (\mathcal{E}, \mathcal{L})$ and the network entity size N is ideally large enough and all possible distinct values describing $\mathcal{E}^*$ appear in T_G^*. Let $\mathcal{A}(\mathcal{E}^*)_j$ and $\mathcal{A}(L_i^*)_j$ denote the j-th attribute of the entity type $\mathcal{E}^*$ and the link type L_i^*. We assume independence among entity attributes and link types with attributes along target meta paths. To focus on the analysis of key factors that may affect the bounds of network cardinality, we also assume an entity has at most in-degree 1, the link among each pair of entities is of all types and the out-degree k of each entity follows the power-law distribution $P_K(k) = ck^{-\alpha}$, which are commonly adopted in social network analysis with $\alpha \in [2, 3]$ [122][194].

To analyze the number of distinct attribute-metapath-combined values describing $\mathcal{E}^*$, or the cardinality $\mathbb{C}(T_G^*)$, of the network schema T_G^*, we begin with the network cardinality $\mathbb{C}(T_G^*)$ without utilizing any neighbors (distance-0); it is equal to the *entity cardinality* $\mathbb{C}(\mathcal{E}^*)$, which is the actual observed number of distinct combined attribute values describing entities:

$$\mathbb{C}(T_G^*)_0 = \mathbb{C}(\mathcal{E}^*).$$

Theoretically, $\mathbb{C}(\mathcal{E}^*)$ can be as high as the product of each entity attribute's cardinality:

$$\mathbb{C}(\mathcal{E}^*) \leq \prod_{j=1}^{|\mathcal{A}(\mathcal{E}^*)|} \mathbb{C}(\mathcal{A}(\mathcal{E}^*)_j).$$

After utilizing the distance-1 neighbors from the entity, let $\mathbb{C}(L_i^*)$ denote the *homogeneous link cardinality*, which is the actual observed number of distinct combined attribute values describing the link L_i^*. Likewise, the maximum value of L_i^* is the product of each attribute cardinality of the link type L_i^*:

$$\mathbb{C}(L_i^*) \leq \prod_{j=1}^{|\mathcal{A}(L_i^*)|} \mathbb{C}(\mathcal{A}(L_i^*)_j).$$

Since entities are connected to one another via different target meta paths, *heterogeneous link cardinality* is no greater than the product of each homogeneous link cardinality:

$$\mathbb{C}(\mathcal{L}^*) \leq \prod_{i=1}^{|\mathcal{L}^*|} \mathbb{C}(L_i^*).$$

Thus, the number of distinct values that an entity can have when distance-1 neighbors are utilized is:

$$\mathbb{C}(T_G^*)_1 = \mathbb{C}(T_G^*)_0 \cdot (\mathbb{C}(\mathcal{E}^*)\mathbb{C}(\mathcal{L}^*))^k.$$

By utilizing neighbors of next distance iteratively, generally when max. distance of utilized neighbors from target entities $n > 0$,

$$\mathbb{C}(T_G^*)_n = \mathbb{C}(T_G^*)_{n-1} \cdot (\mathbb{C}(\mathcal{E}^*)\mathbb{C}(\mathcal{L}^*)^n)^{k^n}. \tag{7.4.1}$$

Based on the distribution function of power law for the out-degree $P_K(k) = ck^{-\alpha}$, we estimate the expected value $\mathbb{E}[\mathbb{C}(T_G^*)_n]$ of (7.4.1) as follows:

$$\begin{aligned}
\mathbb{E}[\mathbb{C}(T_G^*)_n] &= \mathbb{C}(T_G^*)_{n-1} \cdot \mathbb{E}[(\mathbb{C}(\mathcal{E}^*)\mathbb{C}(\mathcal{L}^*)^n)^{k^n}] \\
&\geq \mathbb{C}(\mathcal{E}^*) \cdot \mathbb{E}[(\mathbb{C}(\mathcal{E}^*)\mathbb{C}(\mathcal{L}^*)^n)^{k^n}] \\
&= \mathbb{E}[\mathbb{C}(\mathcal{E}^*) \cdot (\mathbb{C}(\mathcal{E}^*)\mathbb{C}(\mathcal{L}^*)^n)^{k^n}] \\
&= \sum_{k=1}^{N} P_K(k) \cdot \mathbb{C}(\mathcal{E}^*) \cdot (\mathbb{C}(\mathcal{E}^*)\mathbb{C}(\mathcal{L}^*)^n)^{k^n} \\
&> \sum_{k=2}^{N} ck^{-\alpha} \cdot \mathbb{C}(\mathcal{E}^*) \cdot (\mathbb{C}(\mathcal{E}^*)\mathbb{C}(\mathcal{L}^*)^n)^{k^n} \\
&\geq \sum_{k=2}^{N} ck^{-\alpha} \cdot (\mathbb{C}(\mathcal{E}^*)\mathbb{C}(\mathcal{L}^*)^n)^{k^n}.
\end{aligned}$$

Let $f = ck^{-\alpha} \cdot (\mathbb{C}(\mathcal{E}^*)\mathbb{C}(\mathcal{L}^*)^n)^{k^n}, \quad k \in \mathbb{R}, \;\; 2 \leq k \leq N,$

$$\begin{aligned}\frac{\partial f}{\partial k} &= \frac{c(\mathbb{C}(\mathcal{E}^*)\mathbb{C}(\mathcal{L}^*)^n)^{k^n}(nk^n ln(\mathbb{C}(\mathcal{E}^*)\mathbb{C}(\mathcal{L}^*)^n) - \alpha)}{k^{\alpha+1}} \\ &> 0 \qquad (nk^n ln(\mathbb{C}(\mathcal{E}^*)\mathbb{C}(\mathcal{L}^*)^n) > \alpha).\end{aligned}$$

Hence,

$$\mathbb{E}[\mathbb{C}(T_G^*)_n] > 2^{-\alpha}(N-1)c \cdot (\mathbb{C}(\mathcal{E}^*)\mathbb{C}(\mathcal{L}^*)^n)^{2^n}.$$

Since the vertex size N is given, the lower bound of the expected network cardinality is

$$\Omega\{\mathbb{E}[\mathbb{C}(T_G^*)_n]\} = (\mathbb{C}(\mathcal{E}^*)\mathbb{C}(\mathcal{L}^*)^n)^{2^n}. \tag{7.4.2}$$

To establish the upper bound of the expected network cardinality, since $k \leq N$ and we assume N is large, solving the recursion of (7.4.1) we have

$$\begin{aligned}\mathbb{C}(T_G^*)_n &\leq \mathbb{C}(\mathcal{E}^*)^{\frac{N^{n+1}-1}{N-1}}\mathbb{C}(\mathcal{L}^*)^{\frac{N^{n+1}((N-1)n+1)-N}{(N-1)^2}} \\ &\approx (\mathbb{C}(\mathcal{E}^*)\mathbb{C}(\mathcal{L}^*)^n)^{N^n}.\end{aligned}$$

Hence the upper bound of the expected network cardinality is the same as that of the network cardinality when all k is set to N:

$$O\{\mathbb{E}[\mathbb{C}(T_G^*)_n]\} = (\mathbb{C}(\mathcal{E}^*)\mathbb{C}(\mathcal{L}^*)^n)^{N^n}. \tag{7.4.3}$$

(7.4.2) and (7.4.3) complete the proof. ■

Recalling the positive linear relationship between privacy risk and cardinality from Theorem 7.4.4, we obtain the following corollary.

Corollary 7.4.6 *For power-law distribution of the user out-degree, the lower and upper bounds for the expected privacy risk of a heterogeneous information network grows faster than double exponentially with respect to the max. distance of utilized neighbors.*

Corollary 7.4.6 substantiates the privacy risk growth in a heterogeneous information network as observed in Section 7.4.1. It should be emphasized that, it is the heterogeneity of information network links, which is in the mathematical form of $\mathbb{C}(\mathcal{L}^*)^n$, that makes both bounds even a higher order than double exponential growth.

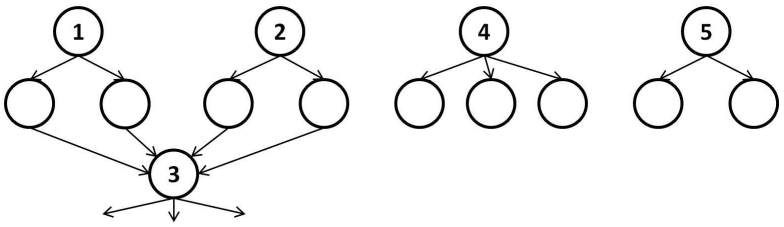


Figure 7.5: The bottleneck scenarios

7.4.4 Limitations of the Analysis

While it may be tempting to conclude that, as long as the max. distance of utilized neighbors grows infinitely, the dimensions for each entity will grow more than double exponentially until the privacy risk $\mathfrak{R}(t)$ becomes 1; it should be pointed out that it is not feasible in practice.

First, the assumption that N is large and all possible distinct values describing $\mathcal{E}^*$ appear in T^*_G may not hold. Then the observed cardinality depends on how to sample from a pool of all possible distinct values. The extreme case is that such "sampling" is so biased that each entity is assigned a value from a very small subset of the pool. However, such a "sampling" bias hardly happens because both $\mathbb{C}(\mathcal{E}^*)$ and $\mathbb{C}(\mathcal{L}^*)$ are actual observed cardinalities which are generally of reasonable sizes in practice.

Second, the assumption that in-degree is at most 1 may not hold and a large-scale information network in practice often has small average diameters [166]. For instance, in Figure 7.5, if user v'_1 and user v'_2 have the same attribute-metapath-combination value after utilizing their distance-1 neighbors, further utilizing their longer-distance neighbors will not make them unique from each other since they will share the same neighbors of distances longer than 1. In addition, the existence of leaf nodes which do not have outgoing edges also prevents utilizing longer-distance of entity neighbors, such as user v'_4 and v'_5 in Figure 7.5. However, in Section 7.6 we show in practice this concern can be addressed because a slight increase of n renders the actual cardinality very close to N.

We show the empirical findings in Table 7.1 and Figure 7.7 that $\mathfrak{R}(t)$ grows very fast when $n \in \{0, 1\}$ and after $n > 1$, $\mathfrak{R}(t)$ grows towards 1 asymptotically until the bottleneck scenarios keep $\mathfrak{R}(t)$ from growing. Nonetheless, the growth order of bounds is consistent with the actual growth during $n \in \{0, 1\}$ so $\mathfrak{R}(t)$ can soon get very close to 1.

7.4.5 Practical Implications to Reduce Privacy Risk

To reduce privacy risk, following the two bounds established in (7.4.2) and (7.4.3), either the entity cardinality $\mathbb{C}(\mathcal{E}^*)$ or link cardinality $\mathbb{C}(\mathcal{L}^*)$ has to be reduced. Since preventing users from sharing their profile

information may restrain the growth of online communities, practical efforts should focus on reducing $\mathbb{C}(\mathcal{L}^*)$ which makes both bounds grow more than double exponentially. Instead of making heterogenous types of links fully accessible from the public, online forums may only allow premium users to access all or partial types of relationships, so $\mathbb{C}(\mathcal{L}^*)$ decreases.

7.5 De-Anonymization Algorithm

To exploit the privacy risk in a heterogeneous information network as identified in Section 7.4, a de-anonymization algorithm is presented with a threat model.

7.5.1 Threat Model

In the privacy risk analysis, we assume the auxiliary data set is exactly the non-anonymized counterpart of the target data set. Although this assumption may hold in real attack scenarios, we consider a more challenging scenario where there is a time gap between the time data publishers release the target data set and the time adversaries start to collect the auxiliary data set from the web. Since a social network generally grows over time, we assume the later collected auxiliary data set contain all the target users and links among them. Other or newly formed users and links can be included in the auxiliary data set as well.

We emphasize that de-anonymizing with the auxiliary data set larger than the target data set is a non-trivial and more challenging task than both data sets are of the same size, especially when allowing certain attribute values and links to grow. First, when the auxiliary data set becomes a superset of the target data set without increasing the cardinality of each tuple from the target data set, the actual risk should be lower because each tuple t_i in the target data set has potentially more matches with users in the auxiliary data set. Second, allowing certain attribute or link growth gives rise to potentially more candidate users in the auxiliary data set that may match a certain target user. For instance, for a user in the target data set that posted 3 tweets and only followed 5 users, any user in the auxiliary data set with more than 3 tweets and more than 5 followees could be a candidate match if we consider number of tweets and number of followers grow over time. Section 7.6 demonstrates that the proved privacy risk can still be exploited even when the task is more challenging.

7.5.2 Algorithm

In Algorithm 3 we formulate a general de-anonymization algorithm **DeHIN** to prey upon the risk of a heterogeneous information network as identified in Section 7.4.

Algorithm 3 De-anonymizing entity v' in a Heterogeneous Information Network: DeHIN (G, G', T^*_G, v', n)

Input: $G = (V, E)$: auxiliary graph, $G' = (V', E')$: target graph, $T^*_G = (\mathcal{E}^*, \mathcal{L}^*)$: target network schema, $v' \in G'$: target entity, n: max. distance of utilized neighbors
Output: C: candidate set from the auxiliary data set matching v'

```
begin
    C ←set ∅
    foreach v ∈ V
        if entity_attribute_match(v', v, E*)
            if n > 0
                if link_match(n, v', v, G, G', T*_G)
                    C ←add v
            else
                C ←add v
    return C
```

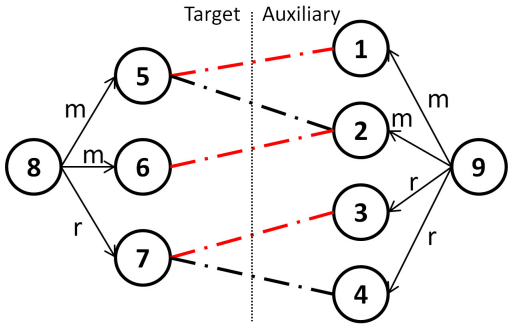


Figure 7.6: Comparing neighbors via multiple types of target network schema links from target and auxiliary data sets

The attribute values of the target entity and the entity from the auxiliary data set is compared by function *entity_attribute_match*. This function can be configured by users depending on different scenarios. We consider the auxiliary data set grows from the target data set in the threat model. So some attribute values may grow over time, such as number of tweets.

The recursive Algorithm 4 is to assist DeHIN to compare the distance-n neighbors from a target entity and an entity in the auxiliary data set whose attributes are matched with those of the target. Likewise, function *link_attribute_match* compares the attribute values of target meta paths (links in the target network schema), if any, and is configurable. The challenge lies in how to compare the neighbors of two entities, after their own entity and link attribute values are matched. Consider the case depicted in Figure 7.6, the target entity v'_8 is matched with entity v_9 in the auxiliary data set for function *entity_attribute_match*, and the target's neighbor v'_5 is matched with v_1 and v_2 (entity v_9's neighbors) via the same type of link for the same function, v'_6 matched with v_2, v'_7 matched with v_3 and v_4. For a growing network, v_9 in the auxiliary data set may be the "grown" target: v_9 itself matches v'_8 in profile attributes, v_9's neighbors v_1 and v_2 in fact are the

Algorithm 4 Comparing neighbors of entities v' and v via heterogeneous links: $link\_match(n, v', v, G, G', T^*_G)$

Input: n: max. distance of utilized neighbors, $v' \in G'$: target entity, v: the entity in auxiliary graph under comparison, $G = (V, E)$: auxiliary graph, $G' = (V', E')$: target graph, $T^*_G = (\mathcal{E}^*, \mathcal{L}^*)$: target network schema

Output: is_match: a boolean value

begin

- $is\_match \xleftarrow{set} true$
- $G_B \xleftarrow{set} \emptyset$ (The bipartite graph modeling neighborhood matching)
- $\mathcal{N}_b(v', L^*_i) \xleftarrow{set} v'$'s neighbors via the link type L^*_i
- $\mathcal{N}_b(v, L^*_i) \xleftarrow{set} v$'s neighbors via the link type L^*_i
- **foreach** *link type* $L^*_i \in \mathcal{L}^*$
  - **foreach** *neighbor* $b'_i \in \mathcal{N}_b(v', L^*_i)$
    - $\emptyset \leftarrow C(b'_i)$; ($C(b'_i)$: candidate set for b'_i)
    - **foreach** *neighbor* $b_i \in \mathcal{N}_b(v, L^*_i)$
      - **if** $link\_attribute\_match(b'_i, b_i)$
        - **if** $entity\_attribute\_match(b'_i, b_i)$
          - **if** $n = 1$
            - $C(b'_i) \xleftarrow{add} b_i$
          - **else**
            - **if** $link\_match(n-1, v', v, G, G', T^*_G)$
              - $C(b'_i) \xleftarrow{add} b_i$
    - $G_B \xleftarrow{add} C(b'_i)$
  - **if** $max\_bipartite\_match(G_B) \neq |\mathcal{N}_b(v', L^*_i)|$
    - $is\_match \xleftarrow{set} false$
- return is_match

non-anonymized v'_5 and v'_6, who are the neighbors of the target via the same type of link. Although v'_7 may be either v_3 or v_4 since they are matched via the same type of link, we can consider the remaining neighbor of v_9, either v_4 or v_3, to be the newly developed relationships during the time gap of the target and auxiliary data sets. Therefore, it is a maximum bipartite matching problem in graph theory (the candidate set for v'_5, $C(v'_5) = \{v_1, v_2\}$, $C(v'_6) = \{v_2\}$, $C(v'_7) = \{v_3, v_4\}$), and the most efficient Hopcroft-Karp algorithm is employed to decide whether such a maximum bipartite matching exists [64]. As long as a maximum bipartite matching exists (*e.g.*, v'_5, v'_6 and v'_7 match v_1, v_2 and v_3 respectively; or v'_5, v'_6 and v'_7 match v_1, v_2 and v_4 respectively), v_9 is considered as a candidate of v'_8. Finally DeHIN returns a *candidate set* containing all entities from the auxiliary data set that may be the target entity. If the size of the correct candidate set is 1, a unique matching is found and the target entity is successfully de-anonymized.

It should be pointed out that, DeHIN is suitable for the general information network and is also applicable to a homogeneous information network, when it is considered as a special case of the general information network whose number of entity type and link type are 1. Besides, DeHIN does not employ isomorphism

testing algorithms due to its high computational cost although we believe it can further enhance the accuracy. In the next section, we show DeHIN is effective in the settings of a heterogeneous information network even without incorporating isomorphism tests.

7.6 Evaluation

In this section, we evaluate the privacy risk and DeHIN performance on *t.qq* data set. Then we show DeHIN is able to beat the investigated graph anonymization algorithms in the settings of a heterogeneous information network, while further sacrificing utility is able to defend the attack. It is also shown that DeHIN undermines the notion of "security by obscurity" for privacy preservation.

7.6.1 Case Study of t.qq Data Set

Following the motivating example in Section 7.1.1, we first evaluate the privacy risk as formalized in Section 7.4. Details of the anonymized KDD Cup 2012 *t.qq* data set is depicted in Section 7.1.1 and Section 7.3. 500 target graphs of 1,000 vertices are sampled from *t.qq* data set where vertices are randomly sampled and all the edges among them are preserved. Although a power-law out-degree distribution is assumed in the analysis (Section 7.4), since increasing privacy risk requires more edges to utilize different distances of neighbors from a target user, the privacy risk may vary when in reality heterogeneous information networks are of different densities:

$$density = \frac{|E|}{m\,|V|^2 + (|\mathcal{L}| - m)\,|V|\,(|V| - 1)} \tag{7.6.1}$$

In (7.6.1), $|E|$ and $|V|$ are the number of edges and vertices in the network. $|\mathcal{L}|$ indicates the total number of link types in the network and m denotes the number of link types which allow nodes to self-link. The denominator of (7.6.1) represents the maximum possible number of edges in the network and the value of density is always between 0 and 1.

57 of the sampled target graphs have density 0.01. The average cardinality of gender, yob, number of tweets, and number of tags for these 57 samples are 3, 87, 643, and 11 respectively. Considering the relatively small size of the target data set, to better observe the growth of risk and variation in terms of different amounts of link types, only the number of tags is used in computing the entity cardinality $\mathbb{C}(\mathcal{E}^*)$. Results in Table 7.1 and Figure 7.7 (Figure 7.7 averages the privacy risk utilizing the same amount of link types) show that privacy risk calculated by Theorem 7.4.4 increases as the utilized heterogeneity information grows, which is the amount of target network schema link types. The drastic growth from distance 0 to 1 is consistent with the established order of growth in (7.4.2) and (7.4.3), then risk grows asymptotically towards

Table 7.1: Privacy risk of the anonymized t.qq data set (density: 0.01, size: 1000) increases as the amount of utilized target network schema link types increases (in percentage)

Types of Links \ Max. Distance	1	2	3
f	84.4	93.8	93.8
m	85.4	93.6	93.8
c	87.6	93.6	93.9
r	90.2	94.2	94.3
f-m	96.0	98.5	98.6
f-c	95.6	98.5	98.5
f-r	96.8	98.5	98.5
m-c	89.9	94.0	94.2
m-r	91.2	94.4	94.5
c-r	91.8	94.4	94.5
f-m-c	96.5	98.5	98.6
f-m-r	96.9	98.6	98.6
f-c-r	96.8	98.6	98.6
m-c-r	92.3	94.5	94.6
f-m-c-r	96.9	98.6	98.6

*f: follow; m: mention; r: retweet; c: comment
**Max. Distance* n: max. distance of utilized neighbors to target entities
*$n = 0$: only target entities' profiles are utilized and risk is always 1.1%

1 until it remains unchanged. Recall Section 7.4.5, the results also justify the practical efforts of reducing accessible link types is able to reduce $\mathbb{C}(\mathcal{L}^*)$ and hence privacy risk. When no link information is accessible, $n = 0$ and privacy risk is reduced efficiently given that the entity cardinality is not large as compared with the entity size.

To evaluate the performance of DeHIN proposed in Section 7.5 on *t.qq* data set, the entire anonymized *t.qq* data set is used as the auxiliary data set while the target data set is the sampled 500 target graphs and none of them contains cliques of size over 3. We will show DeHIN works effectively without the need to create any "sybil nodes" or to rely on easily-detectable graph structures in a large-scale network as required in the existing attacks [7, 116]. The anonymized user IDs (randomly assigned strings) in both target and auxiliary data sets are not used for attribute value matching. After DeHIN employs the remaining attribute and link information described in the motivating example (*user profile, mention, retweet, comment, follow data*) to establish the unique matching between the target user in the target data set and a user in the auxiliary data set, the anonymized user IDs will serve as the ground truth to decide if the unique matching is correct.

Since a social network generally grows over time, we intentionally consider attributes such as *tweet count*, *mention strength*, *retweet strength*, *comment strength* may grow between the time gap of the auxiliary and target data sets. Therefore, the attribute matching functions are configured to allow any user entity in the auxiliary data set with values of these attributes greater than or equal to those of the target user to

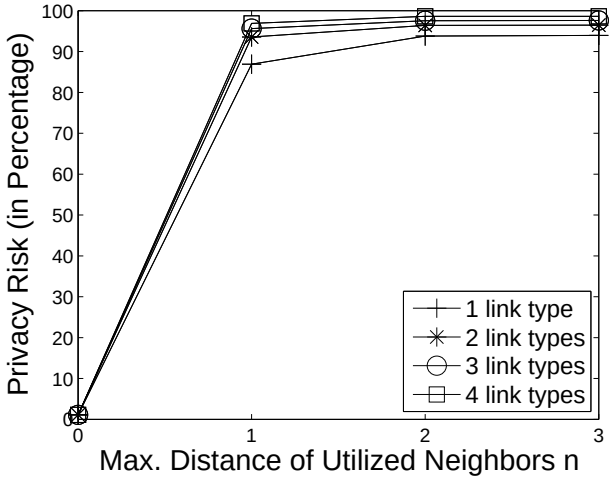


Figure 7.7: Privacy risk increases with more link types

be a candidate. Likewise, we also intentionally consider links may be newly formed in the auxiliary data set for link matching. These considerations make the de-anonymization scenario more practical and more challenging since they will potentially introduce more candidates comparing with the exact attribute or link value matching.

The entire auxiliary data set contains 2,320,895 user entities. With random guessing, the adversary may de-anonymize a user from the target data set with probability no higher than $\frac{1}{2,320,895}$. If the candidate size can be reduced to 100 including the target, the random guessing may be correct with a drastically increased chance of $\frac{1}{100}$. If the candidate size is exactly 1 and such a unique matching is correct, the de-anonymization is successful. Hence, we define two metrics for the experiments:

$$Precision = \frac{\sum_{i=1}^{|V'|} s(v'_i)}{|V'|},$$

$$Reduction\ Rate = \frac{1}{|V'|}\sum_{i=1}^{|V'|}(1 - \frac{|C(v'_i)|}{|V|}),$$

where $|V'|$ and $|V|$ are the size of the target and auxiliary data set, $s = 1$ if $v'_i \in V'$ is successfully de-anonymized, otherwise $s = 0$, and $|C(v'_i)|$ is the size of candidate set for the target v'_i.

The performance of DeHIN on target data sets of different densities is shown in Table 7.2. Clearly, the general performance improves as the density of the target data set increases because higher density indicates DeHIN may be able to utilize more neighbors to expand the dimensions of each target user to achieve unique matchings. It reveals an important problem that, if a group of people have rich social connections, they may have higher social values and may cause adversaries' attention; however, their privacy can be compromised

Table 7.2: Performance of DeHIN on t.qq anonymized data set (in percentage)

Density	Max. Distance 0		Max. Distance 1		Max. Distance 2		Max. Distance 3	
	Precision	Reduction Rate	Precision	Reduction Rate	Precision	Reduction Rate	Precision	Reduction Rate
0.001	4.1	99.836	12.6	99.848	12.6	99.848	12.6	99.848
0.002	5.1	99.925	22	99.947	22.7	99.948	22.7	99.948
0.003	6.5	99.917	32.8	99.944	33.5	99.945	33.5	99.945
0.004	4.3	99.907	39.4	99.941	40.8	99.942	40.9	99.942
0.005	4.3	99.927	48.7	99.969	49.8	99.969	49.9	99.969
0.006	7	99.920	59.4	99.979	61.6	99.980	61.7	99.980
0.007	5.1	99.908	65.6	99.977	68.8	99.978	68.9	99.978
0.008	5.3	99.921	76.6	99.989	78.8	99.989	79	99.989
0.009	6.4	99.914	86.2	99.997	88.6	99.997	88.8	99.997
0.01	5.4	99.892	92.5	99.989	95.6	99.990	95.7	99.990

**Max. Distance n*: max. distance of utilized neighbors to target entities; when $n = 0$, only target entities' profile attributes are utilized.

Table 7.3: Performance of DeHIN on t.qq anonymized data set (density: 0.01) improves as the amount of utilized target network schema link types increases (in percentage)

Types of Links	Max. Distance 1		Max. Distance 2		Max. Distance 3	
	Precision	Reduction Rate	Precision	Reduction Rate	Precision	Reduction Rate
f	68.1	99.982	77.6	99.983	77.7	99.983
m	80.9	99.976	87.8	99.976	88	99.976
c	82.8	99.975	88.7	99.976	88.8	99.976
r	81.1	99.976	88.7	99.976	88.9	99.976
f-m	89.3	99.989	94.2	99.990	94.2	99.990
f-c	90.1	99.989	94.6	99.990	94.6	99.990
f-r	89.2	99.989	94.9	99.990	95	99.990
m-c	84.7	99.976	89.6	99.976	89.7	99.976
m-r	83.2	99.976	89.5	99.977	89.7	99.977
c-r	85.2	99.976	90.3	99.976	90.5	99.976
f-m-c	91.6	99.989	94.8	99.990	94.8	99.990
f-m-r	90.6	99.989	95.1	99.990	95.2	99.990
f-c-r	91.5	99.989	95.4	99.990	95.5	99.990
m-c-r	86.5	99.977	91	99.977	91.2	99.977
f-m-c-r	92.5	99.989	95.6	99.990	95.7	99.990

*f: follow; m: mention; r: retweet; c: comment

**Max. Distance n*: max. distance of utilized neighbors to target entities; when $n = 0$, only target entities' profile attributes are utilized.

*$n = 0$: only target entities' profiles are utilized: precision and reduction rate are always 5.4% and 99.892%.

more easily. Generally, the reduction rate looks promising as compared with the original candidate size of 2.3 million; so even when precision is relatively low on a low-density network, high reduction rate makes manual investigation of matched candidates possibly practical. For a certain density level, precision increases drastically when distance-1 neighbors are utilized, particularly for a higher-density network where there may be more neighbors. Due to the bottleneck scenarios discussed in Section 7.4.3 and Figure 7.5, the performance improves much more slowly or remains unchanged when DeHIN utilizes neighbors of longer distances.

Table 7.4: Performance of DeHIN on t.qq data set of complete graph anonymity (in percentage)

Density	Max. Distance 0		Max. Distance 1		Max. Distance 2		Max. Distance 3	
	Precision	Reduction Rate	Precision	Reduction Rate	Precision	Reduction Rate	Precision	Reduction Rate
0.001	4.1	99.836	11.5	99.847	11.9	99.847	11.9	99.847
0.002	5.1	99.925	19.7	99.941	20.9	99.941	20.9	99.941
0.003	6.5	99.917	29.8	99.938	31.6	99.938	31.6	99.938
0.004	4.3	99.907	35.8	99.936	38.3	99.936	38.4	99.936
0.005	4.3	99.927	44.1	99.963	47.1	99.963	47.1	99.963
0.006	7	99.921	54.3	99.973	57.8	99.973	57.9	99.973
0.007	5.1	99.908	59.5	99.971	64.2	99.971	64.2	99.971
0.008	5.3	99.921	70.3	99.978	74.8	99.978	74.8	99.978
0.009	6.4	99.914	78.1	99.985	83.4	99.986	83.5	99.986
0.01	5.4	99.892	84.4	99.976	89.8	99.976	89.8	99.976

**Max. Distance n*: max. distance of utilized neighbors to target entities; when $n = 0$, only target entities' profile attributes are utilized.

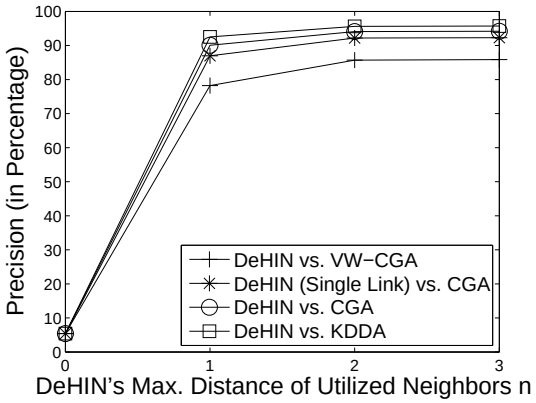


Figure 7.8: DeHIN Precision Improves with More Link Types

To evaluate whether the heterogeneity of an information network improves the performance, we selectively employ different types of links in DeHIN and gradually increase the number of links in de-anonymizing the target data set with potentially a higher social value (density = 0.01). The results in Table 7.3 and Figure 7.8 (Figure 7.8 averages the precision of DeHIN utilizing the same amount of link types) justifies that the performance improves as the utilized heterogeneity information grows, which is the amount of target network schema link types. Moreover, the observed growth trend is consistent to that of privacy risk in Figure 7.7.

7.6.2 Beating Complete Graph Anonymity

The utility of *t.qq* data set has to be preserved to a certain level to ensure effective recommendation algorithms can be designed. We now lower their utility and apply the state-of-the-art graph anonymization algorithms

in Section 7.2.3 on *t.qq* data set. Since adding edges to link all the users will make the entire network safer from all the structural attacks as identified in the work of k-degree, k-neighborhood, k-symmetry, k-automorphism, and k-security, to ensure the best case of defence, we formulate *complete graphs* under different types of links. *Complete Graph Anonymity* can be considered as one of the best case for the investigated graph anonymization algorithms. For instance, when the graph becomes a complete graph after fake links are added, the k turns to be the largest possible value, which is the number of vertices in the graph, for anonymization like k-degree, k-neighborhood, *etc,* as surveyed in Section 7.2.3. To be consistent with these original algorithms that do not consider short-circuited features and to preserve certain utility, we set short-circuited attribute values to be the same random number and keep the existing short-circuited attribute values.

To address the enhanced anonymity, DeHIN is now re-configured to remove all the links with the majority short-circuited attribute value in the entire network before taking effect. Since a social network is generally of density lower than 0.5, it can almost be ensured that all the newly added fake links will be removed from the target data set. However, this step will mistakenly remove the real links that have the same short-circuited attribute values as the fake links from the target data set and $\mathbb{C}(\mathcal{L}^*)$ decreases in (7.4.2) and (7.4.3); thus the performance of DeHIN degrades slightly as shown in Table 7.4 and Figure 7.9(a)—Figure 7.9(j). In Figure 7.9(a)—Figure 7.9(j), complete graph anonymity is able to lower the attack precision effectively when DeHIN only utilizes a single homogeneous link. However, DeHIN still poses great threats to complete graph anonymity, when heterogeneous links are fully utilized.

7.6.3 Defending DeHIN by Sacrificing Utility

To enhance preserved privacy against DeHIN, we have to further lower the utility of the target data set by assigning randomly generated varying weights to the short-circuited attributes of each newly added fake links. It can be observed from Figure 7.9(a)—Figure 7.9(j) that this *Varying Weight Complete Graph Anonymity* renders DeHIN ineffective when utilizing neighbors because most faked links are still preserved in the target data set and n is clear to 0 in (7.4.2) and (7.4.3). However, varying weight values in the fake links cause much higher information loss than assigning the same values; thus the anonymized data utility is sacrificed much more.

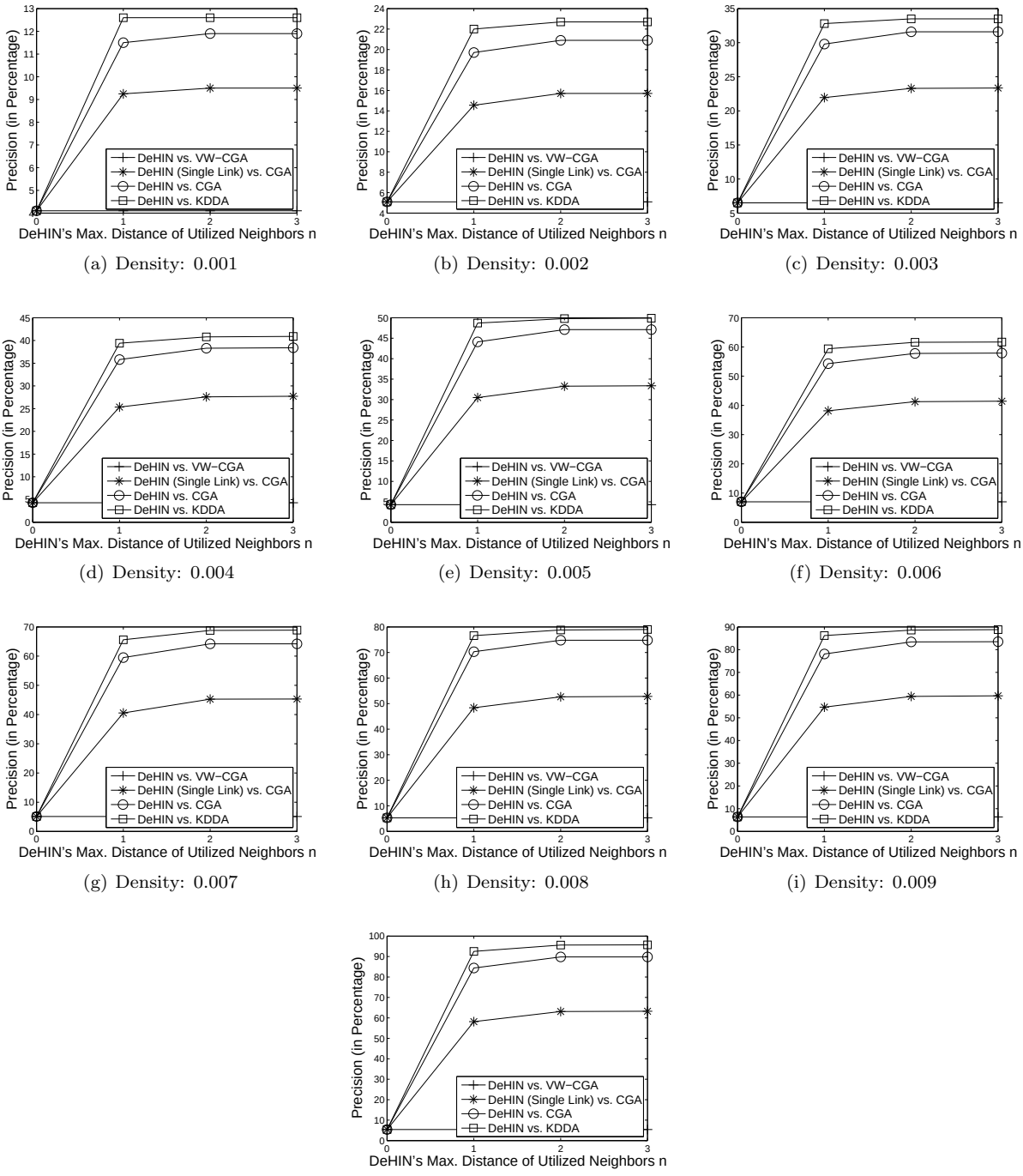
7.6.4 "Security by Obscurity"?

While DeHIN can be launched successfully against certain anonymization (*e.g.*, DeHIN v.s. KDD Cup Original anonymization), it may be (slightly) less effective against other anonymizations (*e.g.*, complete

graph anonymity) even when it is re-configured as in Section 7.6.2. Researchers might be tempted to suggest that, because the adversary might not know what anonymity is employed, he might not be able to launch an attack. Here, we hope to dispel this notion. Suppose an adversary always uses the re-configured DeHIN in Section 7.6.2, the performance on the original *t.qq* anonymization will be exactly the same as that of complete graph anonymity because likewise only the real edges of the same majority attribute values will be affected during de-anonymization. Since DeHIN still poses great threats, this is an extremely important indication that privacy preservation requires more attention from researchers.

7.7 Conclusion

Heterogeneous information networks abound in real life but privacy preservation in such new settings has not received the due attention. In this work, we defined and identified privacy risk in anonymized heterogeneous information networks and presented a new de-anonymization attack that preys upon their risk. We further experimentally substantiated the presence of privacy risk and successfully tested the attack in the KDD Cup 2012 *t.qq* data set. One might find surprising the ease with which the devised attack can beat the investigated anonymization algorithms. While we have selected a small number of anonymization for this initial study, we have no reason to believe that other anonymization will prove impervious to this attack. Hence, our results make a compelling argument that privacy must be a central goal for sensitive heterogeneous information network publishers.



(a) Density: 0.001 (b) Density: 0.002 (c) Density: 0.003

(d) Density: 0.004 (e) Density: 0.005 (f) Density: 0.006

(g) Density: 0.007 (h) Density: 0.008 (i) Density: 0.009

(j) Density: 0.01

Figure 7.9: Precision of DeHIN against different anonymized heterogeneous information networks of different densities (CGA: Complete Graph Anonymity; VW-CGA: Varying Weight Complete Graph Anonymity; KDDA: KDD Cup 2012 t.qq Original Anonymization)

Chapter 8

Summary

There are multifaceted problems in analyzing intentions from big data traces of human activities, and such problems may span a range of machine learning, optimization, and security and privacy.

From the machine learning perspective, we demonstrated that analyzing intentions from industry-scale human activity big data can effectively improve the accuracy of computational models. Specifically, we considered query auto-completion as a case study. After identifying the hitherto-undiscovered adaptive query auto-completion problem and mobile query auto-completion problem, we developed two computational query auto-completion models with intention analysis from large-scale human activities on search interface interactions and on mobile app usage respectively.

From the optimization perspective, we considered generalized machine learning problem settings that hold in the studied query auto-completion problems. We focused on developing lightweight stochastic algorithms as solvers to the large-scale convex optimization problems with theoretical guarantees. For optimizing strongly convex objectives, we designed an accelerated stochastic block coordinate descent method with optimal sampling that uses variance reduction; for optimizing non-strongly convex objectives, we designed a stochastic variance reduced alternating direction method of multipliers with the doubling-trick.

From the security and privacy perspective, we considered the heterogeneous information network settings. To reduce false alarms of medical service providers' suspicious accesses to electronic health records, we discovered potential *de facto* diagnosis specialties that reflect providers' genuine and permissible intentions of accessing records with certain diagnoses. A proposed discovery method exploited the heterogeneous information networks represented by the health record access activities. Besides, we examined the privacy risk in anonymized heterogeneous information networks representing large-scale human activities in social networking. The data were released for improving the prediction accuracy of online networking intentions of the data publishers' online users. We provided a negative result that makes a compelling argument: privacy must be a central goal for sensitive human activity data publishers, especially in the heterogeneous information network setting.

In summary, this dissertation provides evidence to support the following statement: analyzing intentions from big data traces of human activities can improve the accuracy of computational models, such as for query auto-completion; can be faster with an appropriate algorithm design, such as with variance reduction techniques; and can inform security and privacy, such as in the heterogeneous information network setting.

Bibliography

[1] http://www.kddcup2012.org/c/kddcup2012-track1.

[2] National provider identifier. `http://nppes.cms.hhs.gov/NPPES/Welcome.do`.

[3] The npi taxonomy codes for psychology: Apa practice organization offers guidance, advocates for change. `http://www.apapracticecentral.org/reimbursement/npi/select-code.aspx`.

[4] E. Adar, D. S. Weld, B. N. Bershad, and S. S. Gribble. Why we search: visualizing and predicting user behavior. In *Proceedings of the international conference on World Wide Web (WWW)*, pages 161–170. ACM, 2007.

[5] D. Agarwal, B. Long, J. Traupman, D. Xin, and L. Zhang. Laser: A scalable response prediction platform for online advertising. In *Proceedings of the ACM international conference on Web search and data mining (WSDM)*, pages 173–182. ACM, 2014.

[6] Z. Allen-Zhu and Y. Yuan. Improved svrg for non-strongly-convex or sum-of-non-convex objectives. In *Proceedings of the International Conference on International Conference on Machine Learning (ICML)*, pages 1080–1089. JMLR.org, 2016.

[7] L. Backstrom, C. Dwork, and J. Kleinberg. Wherefore art thou r3579x?: anonymized social networks, hidden patterns, and structural steganography. In *Proceedings of the international conference on World Wide Web (WWW)*, pages 181–190. ACM, 2007.

[8] R. Baeza-Yates, G. Dupret, and J. Velasco. A study of mobile search queries in japan. In *Proceedings of the International World Wide Web Conference (WWW)*, 2007.

[9] R. Baeza-Yates, D. Jiang, F. Silvestri, and B. Harrison. Predicting the next app that you are going to use. In *Proceedings of the ACM International Conference on Web Search and Data Mining (WSDM)*, pages 285–294. ACM, 2015.

[10] Z. Bar-Yossef and N. Kraus. Context-sensitive query auto-completion. In *Proceedings of the International Conference on World Wide Web (WWW)*, pages 107–116. ACM, 2011.

[11] C. Benesch, D. Witter, A. Wilder, P. Duncan, G. Samsa, and D. Matchar. Inaccuracy of the international classification of diseases (icd-9-cm) in identifying the diagnosis of ischemic cerebrovascular disease. *Neurology*, 49, 1997.

[12] U. Bhandari, K. Sugiyama, A. Datta, and R. Jindal. Serendipitous recommendation for mobile apps using item-item similarity graph. In *Asia Information Retrieval Symposium*, pages 440–451. Springer, 2013.

[13] C. M. Bishop. *Pattern recognition and machine learning*, volume 1. 2006.

[14] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022, 2003.

[15] Y. Bonomo. Addiction medicine: a new medical specialty in a new age of medicine. *Internal Medicine Journal*, 40(8):543–544, 2010.

[16] L. Bottou. Stochastic gradient descent tricks. In *Neural Networks: Tricks of the Trade*, pages 421–436. Springer, 2012.

[17] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, 2011.

[18] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge University Press, 2004.

[19] P. Breheny and J. Huang. Coordinate descent algorithms for nonconvex penalized regression, with applications to biological feature selection. *The annals of applied statistics*, 5(1):232, 2011.

[20] F. Cai, S. Liang, and M. De Rijke. Time-sensitive personalized query auto-completion. In *Proceedings of the ACM International Conference on Conference on Information and Knowledge Management (CIKM)*, pages 1599–1608. ACM, 2014.

[21] K. Caine and R. Hanania. Patients want granular privacy control over health information in electronic medical records. *Journal of the American Medical Informatics Association*, 20(1):7–15, 2013.

[22] K. Caine and W. M. Tierney. Point and counterpoint: Patient control of access to data in their electronic health records. *Journal of general internal medicine*, 30(1):38–41, 2015.

[23] C. Campos-Castillo and D. L. Anthony. The double-edged sword of electronic health records: implications for patient disclosure. *Journal of the American Medical Informatics Association*, 22(e1):e130–e140, 2015.

[24] C.-C. Chang and C.-J. Lin. Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27, 2011.

[25] K.-W. Chang, C.-J. Hsieh, and C.-J. Lin. Coordinate descent method for large-scale l2-loss linear support vector machines. *The Journal of Machine Learning Research*, 9:1369–1398, 2008.

[26] S. Chaudhuri and R. Kaushik. Extending autocompletion to tolerate errors. In *Proceedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD)*, pages 707–718. ACM, 2009.

[27] G. H. Chen and R. Rockafellar. Convergence rates in forward–backward splitting. *SIAM Journal on Optimization*, 7(2):421–444, 1997.

[28] J. Chen and Q. Gu. Accelerated stochastic block coordinate gradient descent for sparsity constrained nonconvex optimization. In *Conference on Uncertainty in Artificial Intelligence*, 2016.

[29] J. Cheng, A. W.-c. Fu, and J. Liu. K-isomorphism: privacy preserving network publication against structural attacks. In *Proceedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD)*, pages 459–470. ACM, 2010.

[30] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to algorithms*, volume 2. 2001.

[31] E. Costa-Montenegro, A. B. Barragáns-Martínez, and M. Rey-López. Which app? A recommender system of applications in markets: Implementation of the service for monitoring users' interaction. *Expert systems with applications*, 39(10), 2012.

[32] D. Csiba, Z. Qu, and P. Richtárik. Stochastic dual coordinate ascent with adaptive probabilities. *arXiv:1502.08053*, 2015.

[33] Y. Cui and K. Liang. A probabilistic top-n algorithm for mobile applications recommendation. In *IEEE International Conference on Broadband Network & Multimedia Technology (IC-BNMT)*, pages 129–133. IEEE, 2013.

[34] P. Cunningham and S. J. Delany. k-nearest neighbour classifiers. *Multi Classifier System*, pages 1–17, 2007.

[35] C. D. Dang and G. Lan. Stochastic block mirror descent methods for nonsmooth and stochastic optimization. *SIAM Journal on Optimization*, 25(2):856–881, 2015.

[36] C. Davidsson and S. Moritz. Utilizing implicit feedback and context to recommend mobile applications from first use. In *Proceedings of the Workshop on Context-awareness in Retrieval and Recommendation*, pages 19–22. ACM, 2011.

[37] A. Defazio, F. Bach, and S. Lacoste-Julien. Saga: A fast incremental gradient method with support for non-strongly convex composite objectives. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1646–1654, 2014.

[38] A. J. Defazio, T. S. Caetano, and J. Domke. Finito: A faster, permutable incremental gradient method for big data problems. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 1125–1133, 2014.

[39] S. Demetriou, W. Merrill, W. Yang, A. Zhang, and C. A. Gunter. Free for all! assessing user data exposure to advertising libraries on android. In *Proceedings of the Network and Distributed System Security Symposium (NDSS)*, 2016.

[40] D. E. Detmer, B. S. Munger, and C. U. Lehmann. Clinical informatics board certification: history, current status, and predicted impact on the clinical informatics workforce. *Applied Clinical Informatics*, 1(1):11, 2010.

[41] H. Duan and B.-J. P. Hsu. Online spelling correction for query completion. In *Proceedings of the International Conference on World Wide Web (WWW)*, pages 117–126. ACM, 2011.

[42] A. Elixhauser and E. McCarthy. *Clinical classifications for health policy research, version 2: hospital inpatient statistics*. Number 96. US Department of Health and Human Services, Public Health Service, Agency for Health Care Policy and Research, 1996.

[43] D. Fabbri and K. LeFevre. Explanation-based auditing. *Proceedings of the VLDB Endowment*, 5(1):1–12, 2011.

[44] D. Fabbri and K. LeFevre. Explaining accesses to electronic medical records using diagnosis information. *Journal of the American Medical Informatics Association*, 20(1):52–60, 2013.

[45] Centers. for Medicare & Medicaid Services. Taxonomy code. `http://www.cms.gov/Medicare/Provider-Enrollment-and-Certification/MedicareProviderSupEnroll/Taxonomy.html`.

[46] J. Friedman, T. Hastie, H. Höfling, R. Tibshirani, et al. Pathwise coordinate optimization. *The Annals of Applied Statistics*, 1(2):302–332, 2007.

[47] H. Fu, A. Zhang, and X. Xie. De-anonymizing social graphs via node similarity. In *Proceedings of the International Conference on World Wide Web (WWW)*, pages 263–264, 2014.

[48] H. Fu, A. Zhang, and X. Xie. Effective social graph deanonymization based on graph structure and descriptive information. *ACM Transactions on Intelligent Systems and Technology (ACM TIST)*, 6(4):49, 2015.

[49] S. Fu, B. Pi, M. Desmarais, Y. Zhou, W. Wang, and S. Han. Query recommendation and its usefulness evaluation on mobile search engine. In *IEEE International Conference on Systems, Man and Cybernetics (SMC)*, 2009.

[50] W. J. Fu. Penalized regressions: the bridge versus the lasso. *Journal of computational and graphical statistics*, 7(3):397–416, 1998.

[51] D. Gabay and B. Mercier. A dual algorithm for the solution of nonlinear variational problems via finite element approximation. *Computers & Mathematics with Applications*, 2(1):17–40, 1976.

[52] D. Goldfarb, S. Ma, and K. Scheinberg. Fast alternating linearization methods for minimizing the sum of two convex functions. *Mathematical Programming*, 141(1-2):349–382, 2013.

[53] T. Goldstein and S. Osher. The split bregman method for l1-regularized problems. *SIAM Journal on Imaging Sciences*, 2(2):323–343, 2009.

[54] C. Grier, K. Thomas, V. Paxson, and M. Zhang. @ spam: the underground on 140 characters or less. In *Proceedings of the ACM Conference on Computer and Communications Security (CCS)*, pages 27–37. ACM, 2010.

[55] D. Guan, S. Zhang, and H. Yang. Utilizing query change for session search. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 453–462. ACM, 2013.

[56] C. A. Gunter, D. Liebovitz, and B. Malin. Experience-based access management: A life-cycle framework for identity and access management systems. *IEEE Security & Privacy*, 9(5):48, 2011.

[57] S. Gupta, C. Hanson, C. Gunter, M. Frank, D. Liebovitz, B. Malin, et al. Modeling and detecting anomalous topic access. In *IEEE International Conference on Intelligence and Security Informatics (ISI)*, pages 100–105. IEEE, 2013.

[58] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The weka data mining software: an update. *ACM SIGKDD Explorations Newsletter*, 11, 2009.

[59] J. Han, Y. Sun, X. Yan, and P. Yu. Mining knowledge from data: An information network analysis approach. In *IEEE International Conference on Data Engineering (ICDE)*, pages 1214–1217. IEEE, 2012.

[60] T. Hastie, R. Tibshirani, and J. Friedman. *The elements of statistical learning*, volume 2. 2009.

[61] K. Hofmann, B. Mitra, F. Radlinski, and M. Shokouhi. An eye-tracking study of user interactions with query auto completion. In *Proceedings of the ACM International Conference on Information and Knowledge Management (CIKM)*, pages 549–558. ACM, 2014.

[62] R. Hogg and E. Tanis. *Probability and Statistical Inference*. Pearson Prentice Hall, 2006.

[63] Y. Hong, Q.-q. Cai, S. Hua, J.-m. Yao, and Q.-m. Zhu. Negative feedback: the forsaken nature available for re-ranking. In *Proceedings of the International Conference on Computational Linguistics (COLING): Posters*, pages 436–444. Association for Computational Linguistics, 2010.

[64] J. Hopcroft and R. Karp. An n^5/2 algorithm for maximum matchings in bipartite graphs. *SIAM Journal on Computing*, 2(4):225–231, 1973.

[65] C.-J. Hsieh, K.-W. Chang, C.-J. Lin, S. S. Keerthi, and S. Sundararajan. A dual coordinate descent method for large-scale linear svm. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 408–415. ACM, 2008.

[66] B.-J. P. Hsu and G. Ottaviano. Space-efficient data structures for top-k completion. In *Proceedings of the International Conference on World Wide Web (WWW)*, pages 583–594. ACM, 2013.

[67] R. Islam, R. Islam, and T. Mazumder. Mobile application and its global impact. *International Journal of Engineering & Technology (IJEST)*, 10(6), 2010.

[68] S. Ji, G. Li, C. Li, and J. Feng. Efficient interactive fuzzy keyword search. In *Proceedings of the International Conference on World Wide Web (WWW)*, pages 371–380. ACM, 2009.

[69] J.-Y. Jiang, Y.-Y. Ke, P.-Y. Chien, and P.-J. Cheng. Learning user reformulation behavior for query auto-completion. In *Proceedings of the International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR)*, pages 445–454. ACM, 2014.

[70] R. Johnson and T. Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in Neural Information Processing Systems (NIPS)*, 2013.

[71] S. R. Kairam, M. R. Morris, J. Teevan, D. J. Liebling, and S. T. Dumais. Towards supporting search over trending events with social media. *The International AAAI Conference on Web and Social Media (ICWSM)*, 13:43, 2013.

[72] M. Kamvar, M. Kellar, R. Patel, and Y. Xu. Computers and iphones and mobile phones, oh my!: a logs-based comparison of search users on different devices. In *Proceedings of the 18th International Conference on World Wide Web (WWW)*, pages 801–810. ACM, 2009.

[73] M. Karimzadehgan and C. Zhai. Improving retrieval accuracy of difficult queries through generalizing negative document language models. In *Proceedings of the ACM International Conference on Conference on Information and Knowledge Management (CIKM)*, pages 27–36. ACM, 2011.

[74] E. Kharitonov, C. Macdonald, P. Serdyukov, and I. Ounis. User model-based metrics for offline query suggestion evaluation. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 633–642. ACM, 2013.

[75] S. Kim, K.-A. Sohn, and E. P. Xing. A multivariate regression approach to association analysis of a quantitative trait network. *Bioinformatics*, 25(12):i204–i212, 2009.

[76] S. Kogan, D. Levin, B. R. Routledge, J. S. Sagi, and N. A. Smith. Predicting risk from financial reports with regression. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 272–280. Association for Computational Linguistics, 2009.

[77] R. Kohavi. Scaling up the accuracy of naive-bayes classifiers: A decision-tree hybrid. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, volume 96, pages 202–207. Citeseer, 1996.

[78] J. Konečnỳ, J. Liu, P. Richtárik, and M. Takáč. ms2gd: Mini-batch semi-stochastic gradient descent in the proximal setting. *arXiv:1410.4744*, 2014.

[79] J. Konečnỳ, Z. Qu, and P. Richtárik. Semi-stochastic coordinate descent. *arXiv:1412.6293*, 2014.

[80] J. Konečnỳ and P. Richtárik. Semi-stochastic gradient descent methods. *arXiv:1312.1666*, 2013.

[81] W. Kong, R. Li, J. Luo, A. Zhang, Y. Chang, and J. Allan. Predicting search intent based on pre-search context. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 503–512. ACM, 2015.

[82] Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8), 2009.

[83] A. Kulkarni, J. Teevan, K. M. Svore, and S. T. Dumais. Understanding temporal query dynamics. In *Proceedings of the ACM International Conference on Web Search and Data Mining (WSDM)*, pages 167–176. ACM, 2011.

[84] K.-M. Kuo, C.-C. Ma, J. W. Alexander, et al. How do patients respond to violation of their information privacy? *Health Information Management Journal*, 43(2):23, 2014.

[85] K. Lang. Newsweeder: Learning to filter netnews. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 331–339, 1995.

[86] D. D. Lewis, Y. Yang, T. G. Rose, and F. Li. Rcv1: A new benchmark collection for text categorization research. *The Journal of Machine Learning Research*, 5:361–397, 2004.

[87] L. Li, H. Deng, A. Dong, Y. Chang, H. Zha, and R. Baeza-Yates. Analyzing user's sequential behavior in query auto-completion via markov processes. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 123–132. ACM, 2015.

[88] N. Li, T. Li, and S. Venkatasubramanian. t-closeness: Privacy beyond k-anonymity and l-diversity. In *IEEE International Conference on Data Engineering (ICDE)*, pages 106–115. IEEE, 2007.

[89] Y. Li, A. Dong, H. Wang, H. Deng, Y. Chang, and C. Zhai. A two-dimensional click model for query auto-completion. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 455–464. ACM, 2014.

[90] Y. Li and S. Osher. Coordinate descent optimization for 1 minimization with application to compressed sensing; a greedy algorithm. *Inverse Problems and Imaging*, 3(3):487–503, 2009.

[91] Y. Li, X. Tao, A. Algarni, and S.-T. Wu. Mining specific and general features in both positive and negative relevance feedback. In *The Text Retrieval Conference (TREC)*, 2009.

[92] Z.-X. Liao, S.-C. Li, W.-C. Peng, S. Y. Philip, and T.-C. Liu. On the feature discovery for app usage prediction in smartphones. In *IEEE International Conference on Data Mining (ICDM)*, pages 1127–1132. IEEE, 2013.

[93] M. Lichman. UCI machine learning repository. `http://archive.ics.uci.edu/ml`, 2013.

[94] J. Lin, K. Sugiyama, M.-Y. Kan, and T.-S. Chua. New and improved: modeling versions to improve app recommendation. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 647–656. ACM, 2014.

[95] Q. Lin, Z. Lu, and L. Xiao. An accelerated proximal coordinate gradient method and its application to regularized empirical risk minimization. *arXiv preprint arXiv:1407.1296*, 2014.

[96] P.-L. Lions and B. Mercier. Splitting algorithms for the sum of two nonlinear operators. *SIAM Journal on Numerical Analysis*, 16(6):964–979, 1979.

[97] B. Liu, D. Kong, L. Cen, N. Z. Gong, H. Jin, and H. Xiong. Personalized mobile app recommendation: Reconciling app functionality and user privacy preference. In *Proceedings of the ACM International Conference on Web Search and Data Mining (WSDM)*, pages 315–324. ACM, 2015.

[98] H. Liu, M. Palatucci, and J. Zhang. Blockwise coordinate descent procedures for the multi-task lasso, with applications to neural semantic basis discovery. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 649–656. ACM, 2009.

[99] K. Liu and E. Terzi. Towards identity anonymization on graphs. In *Proceedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD)*, pages 93–106. ACM, 2008.

[100] X. Lu. Diagnosis based specialist identification in the hospital. *Thesis, University of Illinois at Urbana-Champaign*, 2014.

[101] X. Lu, A. Zhang, C. A. Gunter, D. Fabbri, D. Liebovitz, and B. Malin. Discovering de facto diagnosis specialties. In *Proceedings of the ACM Conference on Bioinformatics, Computational Biology and Health Informatics (ACM BCB)*, pages 7–16. ACM, 2015.

[102] Z. Lu and L. Xiao. On the complexity analysis of randomized block-coordinate descent methods. *Mathematical Programming*, 152(1-2):615–642, 2015.

[103] J. Luo, S. Zhang, and H. Yang. Win-win search: Dual-agent stochastic game in session search. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 587–596. ACM, 2014.

[104] Y. Ma and H. Lin. A multiple relevance feedback strategy with positive and negative models. *PloS ONE*, 9(8), 2014.

[105] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam. l-diversity: Privacy beyond k-anonymity. *ACM Transactions on Knowledge Discovery from Data*, 1(1):3, 2007.

[106] J. Mairal. Incremental majorization-minimization optimization with application to large-scale machine learning. *arXiv:1402.4419*, 2014.

[107] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to information retrieval*, volume 1. 2008.

[108] P. Martin, A. D. Rubin, and R. Bhatti. Enforcing minimum necessary access in healthcare through integrated audit and access control. In *Proceedings of the ACM Conference on Bioinformatics, Computational Biology and Health Informatics (ACM BCB)*, page 946. ACM, 2013.

[109] R. Mazumder, J. H. Friedman, and T. Hastie. Sparsenet: Coordinate descent with nonconvex penalties. *Journal of the American Statistical Association*, 106(495):1125–1138, 2011.

[110] A. K. Menon, X. Jiang, J. Kim, J. Vaidya, and L. Ohno-Machado. Detecting inappropriate access to electronic health records using collaborative filtering. *Machine Learning*, 95(1):87–101, 2014.

[111] B. Mitra, M. Shokouhi, F. Radlinski, and K. Hofmann. On user interactions with query auto-completion. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 1055–1058. ACM, 2014.

[112] T. Miyanishi and T. Sakai. Time-aware structured query suggestion. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 809–812. ACM, 2013.

[113] K. P. Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

[114] A. Narayanan, H. Paskov, N. Gong, J. Bethencourt, E. Stefanov, E. Shin, and D. Song. On the feasibility of internet-scale author identification. In *IEEE Symposium on Security and Privacy (S & P)*, pages 300–314. IEEE, 2012.

[115] A. Narayanan and V. Shmatikov. Robust de-anonymization of large sparse datasets. In *IEEE Symposium on Security and Privacy (S & P)*, pages 111–125. IEEE, 2008.

[116] A. Narayanan and V. Shmatikov. De-anonymizing social networks. In *IEEE Symposium on Security and Privacy (S & P)*, pages 173–187. IEEE, 2009.

[117] A. Nedić. *Optimization*. Technical Report, UIUC, 2011.

[118] D. Needell, R. Ward, and N. Srebro. Stochastic gradient descent, weighted sampling, and the randomized kaczmarz algorithm. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1017–1025, 2014.

[119] Y. Nesterov. *Introductory lectures on convex optimization: A Basic Course*, volume 87. Springer Science & Business Media, 2004.

[120] Y. Nesterov. *Gradient methods for minimizing composite objective function*. Technical report, Center for Operations Research and Econometrics, 2007.

[121] Y. Nesterov. Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM Journal on Optimization*, 22(2):341–362, 2012.

[122] M. Newman. *Networks: an introduction*. 2009.

[123] A. V. Nimkar and S. K. Ghosh. An access control model for cloud-based emr federation. *International Journal of Trust Management in Computing and Communications*, 2(4):330–352, 2014.

[124] H. Ouyang, N. He, L. Tran, and A. Gray. Stochastic alternating direction method of multipliers. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 80–88, 2013.

[125] Y. Ouyang, Y. Chen, G. Lan, and E. Pasiliao Jr. An accelerated linearized alternating direction method of multipliers. *SIAM Journal on Imaging Sciences*, 8(1):644–681, 2015.

[126] U. Premarathne, F. Han, H. Liu, and I. Khalil. Impact of privacy issues on user behavioural acceptance of personalized mhealth services. In *Mobile Health*, pages 1089–1109. Springer, 2015.

[127] D. Prokhorov. Ijcnn 2001 neural network competition. *Slide presentation in IJCNN*, 1:97, 2001.

[128] Z. Qin and D. Goldfarb. Structured sparsity via alternating direction methods. *The Journal of Machine Learning Research*, 13(1):1435–1468, 2012.

[129] Z. Qu and P. Richtárik. Coordinate descent with arbitrary sampling i: Algorithms and complexity. *arXiv preprint arXiv:1412.8060*, 2014.

[130] Z. Qu, P. Richtárik, and T. Zhang. Randomized dual coordinate ascent with arbitrary sampling. *arXiv:1411.5873*, 2014.

[131] S. Reddi, A. Hefny, C. Downey, A. Dubey, and S. Sra. Large-scale randomized-coordinate descent methods with non-separable linear constraints. *arXiv preprint arXiv:1409.2617*, 2014.

[132] P. Richtárik and M. Takáč. Iteration complexity of randomized block-coordinate descent methods for minimizing a composite function. *Mathematical Programming*, 144(1-2):1–38, 2014.

[133] J. J. Rocchio. Relevance feedback in information retrieval. *The SMART Retrieval System Experiments in Automatic Document Processing*, 1971.

[134] N. L. Roux, M. Schmidt, and F. R. Bach. A stochastic gradient method with an exponential convergence rate for finite training sets. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2663–2671, 2012.

[135] M. Schmidt, R. Babanezhad, M. O. Ahemd, A. Defazio, A. Clifton, and A. Sarkar. Non-uniform stochastic average gradient method for training conditional random fields. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2015.

[136] M. Schmidt, N. L. Roux, and F. Bach. Minimizing finite sums with the stochastic average gradient. *arXiv:1309.2388*, 2013.

[137] S. Shalev-Shwartz and A. Tewari. Stochastic methods for l 1-regularized loss minimization. *The Journal of Machine Learning Research*, 12:1865–1892, 2011.

[138] S. Shalev-Shwartz and T. Zhang. Proximal stochastic dual coordinate ascent. *arXiv:1211.2717*, 2012.

[139] S. Shalev-Shwartz and T. Zhang. Accelerated mini-batch stochastic dual coordinate ascent. In *Advances in Neural Information Processing Systems (NIPS)*, pages 378–385, 2013.

[140] S. Shalev-Shwartz and T. Zhang. Stochastic dual coordinate ascent methods for regularized loss. *The Journal of Machine Learning Research*, 14(1):567–599, 2013.

[141] M. Shokouhi. Learning to personalize query auto-completion. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 103–112. ACM, 2013.

[142] M. Shokouhi and K. Radinsky. Time-sensitive query auto-completion. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 601–610. ACM, 2012.

[143] T. R. Shulimzon. Interventional pulmonology: a new medical specialty. *The Israel Medical Association journal*, 16(6):379–384, 2014.

[144] Y. Singer and J. C. Duchi. Efficient learning using forward-backward splitting. In *Advances in Neural Information Processing Systems*, pages 495–503, 2009.

[145] Y. Song, H. Ma, H. Wang, and K. Wang. Exploring and exploiting user search behavior on mobile and tablet devices to improve search relevance. In *Proceedings of the 22nd International Conference on World Wide Web*, pages 1201–1212. ACM, 2013.

[146] N. D. Soulakis, M. B. Carson, Y. J. Lee, D. H. Schneider, C. T. Skeehan, and D. M. Scholtens. Visualizing collaborative electronic health record usage for hospitalized patients with heart failure. *Journal of the American Medical Informatics Association*, 22(2):299–311, 2015.

[147] S. Sra, S. Nowozin, and S. J. Wright. *Optimization for machine learning*. Mit Press, 2012.

[148] T. Strohmer and R. Vershynin. A randomized kaczmarz algorithm with exponential convergence. *Journal of Fourier Analysis and Applications*, 15(2):262–278, 2009.

[149] Y. Sun and J. Han. Mining heterogeneous information networks: Principles and methodologies. *Synthesis Lectures on Data Mining and Knowledge Discovery*, 3(2), 2012.

[150] Y. Sun, J. Han, X. Yan, P. S. Yu, and T. Wu. Pathsim: Meta path-based top-k similarity search in heterogeneous information networks. *Proceedings of the VLDB Endowment (VLDB)*, 4(11):992–1003, 2011.

[151] Y. Sun, J. Han, P. Zhao, Z. Yin, H. Cheng, and T. Wu. Rankclus: integrating clustering with ranking for heterogeneous information network analysis. In *Proceedings of the International Conference on Extending Database Technology (EDBT)*, pages 565–576. ACM, 2009.

[152] Y. Sun, B. Norick, J. Han, X. Yan, P. S. Yu, and X. Yu. Pathselclus: Integrating meta-path selection with user-guided object clustering in heterogeneous information networks. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 7(3):11, 2013.

[153] T. Suzuki. Dual averaging and proximal gradient descent for online alternating direction multiplier method. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 392–400, 2013.

[154] T. Suzuki. Stochastic dual coordinate ascent with alternating direction method of multipliers. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 736–744, 2014.

[155] L. Sweeney. Achieving k-anonymity privacy protection using generalization and suppression. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(05):571–588, 2002.

[156] L. Sweeney. k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(05):557–570, 2002.

[157] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[158] R. Tibshirani, J. Bien, J. Friedman, T. Hastie, N. Simon, J. Taylor, and R. J. Tibshirani. Strong rules for discarding predictors in lasso-type problems. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 74(2):245–266, 2012.

[159] R. Tibshirani, M. Saunders, S. Rosset, J. Zhu, and K. Knight. Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(1):91–108, 2005.

[160] R. J. Tibshirani, J. E. Taylor, E. J. Candes, and T. Hastie. *The solution path of the generalized lasso.* Stanford University, 2011.

[161] W. M. Tierney, S. A. Alpert, A. Byrket, K. Caine, J. C. Leventhal, E. M. Meslin, and P. H. Schwartz. Provider responses to patients controlling access to their electronic health records: a prospective cohort study in primary care. *Journal of general internal medicine*, 30(1):31–37, 2015.

[162] H. Wang and A. Banerjee. Online alternating direction method. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 1119–1126, 2012.

[163] H. Wang and A. Banerjee. Randomized block coordinate descent for online and stochastic optimization. *arXiv:1407.0107*, 2014.

[164] X. Wang, H. Fang, and C. Zhai. Improve retrieval accuracy for difficult queries using negative feedback. In *Proceedings of the ACM Conference on Conference on Information and Knowledge Management (CIKM)*, pages 991–994. ACM, 2007.

[165] X. Wang, H. Fang, and C. Zhai. A study of methods for negative relevance feedback. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 219–226. ACM, 2008.

[166] S. Wasserman and K. Faust. *Social network analysis: Methods and applications*, volume 8. 1994.

[167] S. Wenxuan and Y. Airu. Interoperability-enriched app recommendation. In *IEEE International Conference on Data Mining Workshop (ICDMW)*, pages 1242–1245. IEEE, 2014.

[168] S. Whiting and J. M. Jose. Recent and robust query auto-completion. In *Proceedings of the International Conference on World Wide Web (WWW)*, pages 971–982. ACM, 2014.

[169] T. T. Wu and K. Lange. Coordinate descent algorithms for lasso penalized regression. *The Annals of Applied Statistics*, pages 224–244, 2008.

[170] W. Wu, Y. Xiao, W. Wang, Z. He, and Z. Wang. K-symmetry model for identity anonymization in social networks. In *Proceedings of the International Conference on Extending Database Technology*, pages 111–122. ACM, 2010.

[171] C. Xiao, J. Qin, W. Wang, Y. Ishikawa, K. Tsuda, and K. Sadakane. Efficient error-tolerant query autocompletion. *Proceedings of the Very Large Data Base Endowment (VLDB)*, 6(6), 2013.

[172] L. Xiao. Dual averaging method for regularized stochastic learning and online optimization. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2116–2124, 2009.

[173] L. Xiao and T. Zhang. A proximal stochastic gradient method with progressive variance reduction. *SIAM Journal on Optimization*, 24(4):2057–2075, 2014.

[174] Y. Xu and W. Yin. Block stochastic gradient iteration for convex and nonconvex optimization. *SIAM Journal on Optimization*, 25(3):1686–1716, 2015.

[175] C. Yang, T. Wang, G. Yin, H. Wang, M. Wu, and M. Xiao. Personalized mobile application discovery. In *Proceedings of the International Workshop on Crowd-based Software Development Methods and Technologies*, pages 49–54. ACM, 2014.

[176] H. Yang, M. Sloan, and J. Wang. Dynamic information retrieval modeling. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 1290–1290. ACM, 2014.

[177] J. Yang and Y. Zhang. Alternating direction algorithms for \ell_1-problems in compressive sensing. *SIAM Journal on Scientific Computing*, 33(1):250–278, 2011.

[178] S. Yang, H. Yu, W. Deng, and X. Lai. Mobile application recommendations based on complex information. In *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*, pages 415–424. Springer, 2015.

[179] S.-H. Yang, B. Long, A. J. Smola, H. Zha, and Z. Zheng. Collaborative competitive filtering: learning recommender using context of user choice. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 295–304. ACM, 2011.

[180] S. Yao, S. Hu, Y. Zhao, A. Zhang, and T. Abdelzaher. Deepsense: A unified deep learning framework for time-series mobile sensing data processing. In *Proceedings of the International Conference on World Wide Web (WWW)*, pages 351–360, 2017.

[181] P. Yin, P. Luo, W.-C. Lee, and M. Wang. App recommendation: a contest between satisfaction and temptation. In *Proceedings of the ACM International Conference on Web Search and Data Mining (WSDM)*, pages 395–404. ACM, 2013.

[182] A. Zhang, L. Garcia-Pueyo, J. B. Wendt, M. Najork, and A. Broder. Email category prediction. In *Proceedings of the International Conference on World Wide Web (WWW)*, pages 495–503. International World Wide Web Conferences Steering Committee, 2017.

[183] A. Zhang, A. Goyal, R. Baeza-Yates, Y. Chang, J. Han, C. A. Gunter, and H. Deng. Towards mobile query auto-completion: An efficient mobile application-aware approach. In *Proceedings of the 25th International Conference on World Wide Web (WWW)*, pages 579–590, 2016.

[184] A. Zhang, A. Goyal, W. Kong, H. Deng, A. Dong, Y. Chang, C. A. Gunter, and J. Han. adaqac: Adaptive query auto-completion via implicit negative feedback. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 143–152. ACM, 2015.

[185] A. Zhang and Q. Gu. Accelerated stochastic block coordinate descent with optimal sampling. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 2035–2044. ACM, 2016.

[186] A. Zhang, X. Xie, K. C.-C. Chang, C. A. Gunter, J. Han, and X. Wang. Privacy risk in anonymized heterogeneous information networks. In *Proceedings of the International Conference on Extending Database Technology (EDBT)*, pages 595–606, 2014.

[187] W. Zhang, Y. Chen, C. Gunter, D. Liebovitz, and B. Malin. Evolving role definitions through permission invocation patterns. In *Proceedings of the ACM Symposium on Access Control Models and Technologies*, pages 37–48. ACM, 2013.

[188] W. Zhang, C. A. Gunter, D. Liebovitz, J. Tian, and B. Malin. Role prediction using electronic medical record system audits. In *AMIA Annual Symposium Proceedings*, volume 2011, page 858. American Medical Informatics Association, 2011.

[189] W.-j. Zhang and J.-y. Wang. The study of methods for language model based positive and negative relevance feedback in information retrieval. In *International Symposium on Information Science and Engineering (ISISE)*, pages 39–43. IEEE, 2012.

[190] P. Zhao and T. Zhang. Stochastic optimization with importance sampling. *arXiv preprint arXiv:1401.2753*, 2014.

[191] T. Zhao, M. Yu, Y. Wang, R. Arora, and H. Liu. Accelerated mini-batch randomized block coordinate descent method. In *Advances in Neural Information Processing Systems (NIPS)*, pages 3329–3337, 2014.

[192] S. Zheng and J. T. Kwok. Fast-and-light stochastic admm. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 2407–2413, 2016.

[193] L. W. Zhong and J. T. Kwok. Fast stochastic alternating direction method of multipliers. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 46–54, 2014.

[194] B. Zhou and J. Pei. Preserving privacy in social networks against neighborhood attacks. In *IEEE 24th International Conference on Data Engineering (ICDE)*, pages 506–515. IEEE, 2008.

[195] B. Zhou, J. Pei, and W. Luk. A brief survey on anonymization techniques for privacy preserving publishing of social network data. *ACM SIGKDD Explorations Newsletter*, 10(2):12–22, 2008.

[196] H. Zhu, H. Xiong, Y. Ge, and E. Chen. Mobile app recommendations with security and privacy awareness. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 951–960. ACM, 2014.

[197] R. Zhu, A. Zhang, J. Peng, and C. Zhai. Exploiting temporal divergence of topic distributions for event detection. In *Proceedings of the IEEE International Conference on Big Data (IEEE BigData)*, pages 164–171. IEEE, 2016.

[198] L. Zou, L. Chen, and M. Özsu. K-automorphism: A general framework for privacy preserving network publication. *Proceedings of the VLDB Endowment*, 2(1):946–957, 2009.

Appendix

Publications during the Ph.D. Study

Below is a list of my publications during the Ph.D. study as of March 25, 2017 (a few other papers are under review).

- A. Zhang, L. Garcia-Pueyo, J. B. Wendt, M. Najork, and A. Broder.
  Email Category Prediction [182].
  In Proceedings of the 26th International World Wide Web Conference (**WWW**), 2017.

- S. Yao, S. Hu, Y. Zhao, A. Zhang, and T. Abdelzaher.
  DeepSense: A Unified Deep Learning Framework for Time-Series Mobile Sensing Data Processing [180].
  In Proceedings of the 26th International World Wide Web Conference (**WWW**), 2017.

- A. Zhang, A. Goyal, R. Baeza-Yates, Y. Chang, J. Han, C. A. Gunter, and H. Deng.
  Towards Mobile Query Auto-Completion: An Efficient Mobile Application-Aware Approach [183].
  In Proceedings of the 25th International World Wide Web Conference (**WWW**), 2016.

- A. Zhang and Q. Gu.
  Accelerated Stochastic Block Coordinate Descent with Optimal Sampling [185].
  In Proceedings of the 22nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining (**KDD**), 2016.

- S. Demetriou, W. Merrill, W. Yang, A. Zhang, and C. A. Gunter.
  Free for All! Assessing User Data Exposure to Advertising Libraries on Android [39].
  In Proceedings of the Network and Distributed System Security Symposium (**NDSS**), 2016.

- R. Zhu, A. Zhang, J. Peng, and C. Zhai.
  Exploiting Temporal Divergence of Topic Distributions for Event Detection [197].
  In Proceedings of the IEEE International Conference on Big Data (**IEEE BigData**), 2016.

- A. Zhang, A. Goyal, W. Kong, H. Deng, A. Dong, Y. Chang, C. A. Gunter, and J. Han.
  adaQAC: Adaptive Query Auto-Completion via Implicit Negative Feedback [184].
  In Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval (**SIGIR**), 2015.

- W. Kong, R. Li, J. Luo, A. Zhang, Y. Chang, and J. Allan.
  Predicting Search Intent Based on Pre-Search Context [81].
  In Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval (**SIGIR**), 2015.

- H. Fu, A. Zhang, and X. Xie.
  Effective Social Graph De-anonymization based on Graph Structure and Descriptive Information [48].
  In ACM Transactions on Intelligent Systems and Technology (**ACM TIST**), Vol. 6, No. 4, 2015.

- X. Lu*, A. Zhang*, C. A. Gunter, D. Fabbri, D. Liebovitz, and B. Malin (*equal contribution).
  Discovering De Facto Diagnosis Specialties [101].
  In Proceedings of the 6th ACM Conference on Bioinformatics, Computational Biology, and Health Informatics (**ACM BCB**), 2015.

- A. Zhang, X. Xie, K. C.-C. Chang, C. A. Gunter, J. Han, and X. Wang.
  Privacy Risk in Anonymized Heterogeneous Information Networks [186].
  In Proceedings of the 17th International Conference on Extending Database Technology (**EDBT**), 2014.

- H. Fu, A. Zhang, and X. Xie.
  De-anonymizing Social Graphs via Node Similarity [47].
  In Proceedings of the 23rd International World Wide Web Conference (**WWW**), 2014.