Computer Science
&
Data Science

Capstone Report - Fall 2024

Game Theoretic Formulation of Personal Data Sharing

Marco Li,
Tianshi Zhou,
Evelyn Feng

supervised by
Hongyi Wen & Mathieu Laurière

Preface

As students in the field of computer science and data science, we have always been fascinated by the interplay between individual rights and technological advancements. This study is inspired by the growing concern about personal data misuse in the digital era. The target audience for this work includes researchers, scientists, and policymakers who are engaged in the fields of data privacy, game theory, and algorithm design. The paper introduces a novel framework to address a pressing societal challenge, contributing to the broader discourse on ethical and effective data-sharing practices.

Acknowledgements

We would like to express our heartfelt gratitude to the study's advisors, Professor Wen and Professor Laurière, for their invaluable guidance and constructive feedback throughout this project. Their expertise and encouragement were pivotal in shaping the direction of this work. We are also grateful to other capstone instructors and the study's peers for their thoughtful insights and discussions, which enriched this research. Finally, we deeply appreciate the resources and collaborative environment provided by NYU Shanghai, which made this project possible.

Abstract

While users enjoy personalized services by sharing data to platforms, the increasing societal focus on privacy makes it an interesting question how users can share data strategically to protect their privacy without sacrificing service quality dramatically. Despite some previous efforts on this privacy-utility trade-off in specific domains like genomic statistics or the Internet of Things (IoT), it remains a challenge to find a general solution taking into account major characteristics of popular services due to diverse data-using settings and the technical difficulty of modeling the strategic interactions between users and service providers. In this paper, the study starts by developing a game model that considers the personalization of service with recommendations. Via analysis, fixed action simulations, and DQN, the study demonstrates optimal stationary policies for users as well as some systemic challenges for minority groups through the learned value functions in both random environments and Nash Equilibria of multiple strategic users. The study provides practical guidance for users with privacy concerns and an analytical framework along with possible directions for future studies.

Keywords

Game Theory; Personal Data Privacy; Reinforcement Learning; Recommendation Systems; Mean Field Game; Nash Equilibrium

Contents

1 Introduction

Personal data has become a key ingredient in the success of many population applications. They allow systems to learn the preferences of users and, hence, to provide personalized services or recommendations. However, sharing private data comes with some concerns for users. In fact, service providers such as Google, Twitter, and Facebook are enabling their users to revisit, erase, and rectify their historical profiles. Previous studies have challenged the common belief that "more data is essential to better recommendation performance" and suggest a potential win-win solution for services and end users. Each user thus faces a trade-off between getting a better service and protecting their privacy. The study believes a general game-theoretic model on personal data sharing would be a step towards data minimization, revealing near-optimal strategies for the users to preserve or share personal data for desired privacy preservation and quality of service.

The utility-privacy trade-off for users in a recommendation system is not trivial due to the complicated interactions between multiple users and the recommendation system. Users naturally don't observe other users' states and actions but are affected by the trained recommendation model. Each user needs to learn the strategy dependent on the unknown recommendation model conditional to population behavior and deduce the population behavior by observing different components of the reward. To tackle this complexity, this project provides a game-theoretic viewpoint on personal data sharing between a group of strategic agents and a central unit. Each agent can decide the proportion of personal data to share, get a reward that depends on the quality of the service provided by the central unit, and get a penalty that depends on the amount of personal data shared. The research first studies the optimal strategy of a single strategic user in a stable environment with analysis and reinforcement learning and then moves to the game with multiple-strategic users and the mean field game.

2 Related Work

In this section, the study reviews existing research relevant to this project, categorizing the literature into three primary domains: General Privacy-Preserving Techniques, Game Theory Applications for Privacy, and Recommendation Systems. Each domain provides a coherent narrative that informs and motivates the study.

Legal Foundations. Regulatory frameworks have provided a foundational understanding of privacy from a legal and ethical perspective [1]. discusses the problems associated with privacy self-management from the cognitive and structural, pointing out the common cognitive mistakes and systematic obstacles preventing individuals from managing their own private data properly. In 2016, the General Data Protection Regulation (GDPR) [2] proposed a more detailed concept of data minimization, and [3] operationalized the data minimization principles.

2.1 General Privacy Preserving Techniques

Privacy preservation is a long-standing area of research. Previous studies specific to data privacy have proposed privacy-preserving techniques from different directions, including privacy-preserving mechanisms, noise and signal filtering technologies, and user-centric control frameworks, such as Personal Data Stores [4]. A frequently used concept is differential privacy [5] with a clear mathematical definition, which addresses the privacy security of an algorithm by its different output probability on two adjacent datasets.

2.1.1 Privacy Preserving Mechanisms

Those mechanisms mainly aim to preserve privacy using engineering techniques around the processing stage without much influence on data sharing. Some featuring methods include a fully homomorphic encryption method [6] for secure computation without decryption, k-anonymity [7], and the application of blockchain [8].

Another set of notable studies related to data-based learning tasks proposed federated learning implemented through a synchronous iterative model averaging algorithm [9], providing privacy guarantee through differential privacy [5] and secure multi-party computation [10].

2.1.2 Noise and Signal Filtering

Data reconstruction attacks are attacks on a learned model or system in order to reconstruct information about data used for training. Liu et al. give a systematic evaluation of different data reconstruction attack methods and defenses [11]. These defense methods mainly include filtering and adding noise to gradient information, such as L2-gradient noise, gradient clipping, dropout, and gradient pruning.

2.1.3 Application in Internet of Things

The IoT domain presents unique privacy challenges due to the massive data collection and communication involved. Shen et al. review various game-theoretic methods that can be applied in the current Internet of Things (IoT) environments, such as non-cooperative and differential games [12]. These methods address conflicts between data attackers and defenders, making them a significant advancement over traditional privacy-preserving methods. Their research provides a solid foundation on how game theory can be leveraged to balance privacy and data utility, which is central to the study's goal of achieving privacy-utility trade-offs.

Wang et al. further extend this analysis by researching a non-cooperative game considering the optimization of communication and privacy in intelligent transportation systems. They developed an internal optimization model, which weights the data privacy against network delay, and an external anonymous update model, which achieves the maximum of both communication and privacy functions [13]. In this approach, the communication quality of vehicles is not affected while achieving self-adaptation to improve privacy. These techniques inspire the model building where the study seeks to maximize the reward of each user.

2.2 Recommendation System

Recommendation systems have evolved significantly to address privacy concerns while maintaining accuracy. This section examines various approaches, including image-based systems, collaborative filtering, data minimization techniques, and their application in Internet of Things.

2.2.1 Image-based Recommendation

McAuley et al. introduce an image-based recommendation system that provides a valuable way to recommend complementary and alternative goods using non-private data [14]. Their method utilizes convolutional neural networks (CNN) to extract image features, which reduces the reliance on traditional user data (such as purchase history). However, these systems fail to address privacy concerns related to user preferences and interactions. In the context of the study, the system highlights potential ways to minimize the use of personal data by relying more on non-sensitive data in recommendation models, with constructive implications for future applications in special areas.

2.2.2 Collaborative Filtering

In contrast to the previous approach, Polat and Du propose a privacy-protecting collaborative filtering (CF) system that is directly in line with the goal of this study: to provide accurate recommendations while minimizing the use of personal data [15]. They introduce stochastic perturbation techniques to mask personal data while maintaining recommendation accuracy. The balance between maintaining high recommendation accuracy and protecting user data in their model is in line with the research goal.

Benkessira et al. studied a collaborative filtering approach based on game theory for recommendation systems. They further enhance CF systems by integrating game theory through Shapley Value clustering, which reflects the marginal value that the player brings to the game and the bargaining potential of the player [16]. It improves community-based recommendations by accounting for user-specific contributions. The superiority of their method is that the preselection of neighborhood takes into account the importance of every user from its closest representative and other users in the same cluster.

2.2.3 Data Minimization and User-Controlled Data Filtering

Data minimization strategies play a critical role in privacy-preserving recommendation systems. Biega et al. highlight the importance of aligning these strategies with GDPR principles and data minimization principles in the context of personalization systems [3]. They propose performance-based interpretations of the data minimization principle that combine the limitations of data collection and quality metrics. The per-user minimum performance is relevant to this study.

Wen et al. conducted research on recommendations under user-controlled data filtering, demonstrating the effect of time-sensitive user data filtering. Users choose to share only recent "N days" of data while maintaining the overall performance of the recommendation system. They evaluate three widely used collaborative filtering algorithms: Probabilistic Matrix Factorization (PMF), Bayesian Personalized Ranking (BPR), and Collaborative Metric Learning with Uniform Weights (U-CML) [17]. Their findings provide valuable guidance for the project's goal to reach a win-win situation for service providers and end users.

2.3 Game Theory Application for Privacy

Since the study is constructing a game, game theory comes hand in hand with the topic. Game theory serves as a powerful lens through which to understand the trade-offs and interactions

between privacy, data utility, and platform performance. It provides tools for analyzing situations in which parties make decisions that are interdependent. This interdependence causes each player to consider the other player's possible decisions or strategies, in formulating strategy. It offers insights into how platforms can achieve a balance between the use of personal data and maintaining privacy.

2.3.1 Bayes Theorem and Mixed-strategy game model

Bayesian game-theoretic framework is introduced by Zhang et al. to optimize privacy protection in genomic data sharing [18]. This model protects genomic data against Bayesian attacks by adding noise, highlighting vulnerabilities in traditional defense mechanisms like likelihood ratio testing. This approach effectively balances privacy risks and utility through a hybrid strategy and neural network, demonstrating its superiority in reducing data quality degradation while safeguarding sensitive information.

Xie et al. introduce a mixed-strategy game model to address privacy risks in mobile learning environments, where platforms decide between protecting user data or monetizing it [19]. This model incorporates factors such as data sensitivity and platform credibility and analyzes in detail how each participant can maximize the benefits. It not only considers the game between users but also the game between the platform and the attacker. It is a rather complex and comprehensive model, and the platform's benefits are quantified by service quality.

2.3.2 Summary and Possible Use

These game theory frameworks provide valuable insights into how platforms can manage user data while minimizing privacy risks. The study aims to extend these models by applying Mean Field Games (MFG) to simulate and capture user behavior as realistically as possible. In addition, the income function and strategic considerations of Zhang et al. [18], and Xie et al. [19] are applied to the study after modification. For example, reward formulas for attackers and platforms that take into account data sensitivity and trustworthiness help to model the trade-offs involved in minimizing the use of personal data. Mixed strategy equalization, where each participant adjusts strategies to maximize returns, can be used for user satisfaction and privacy protection on the platform. Together, these existing frameworks provide a solid foundation for understanding the dynamic relationship between privacy, utilities, and platform performance.

3 Solution

3.1 Model

When formulating the game model, the study wants the model to capture essential features of reality while not too complex to study. To that end, the study considers primarily two features to be met in the model. a) **Interaction:** The data provided by users should affect the parameters of the trained recommender model and hence the prediction made to all users. b) **Temporal effect:** A user's action might affect themselves in the future. With the two features above, the study can model how strategies of the population affect each user's behavior and look for a good long-term stationary strategy.

3.1.1 Model details

Consider a game with $|U|$ users, $|I|$ items, and an infinite time horizon, where U is the set of users, and I is the set of items. The total data I_u^t of user u on the day $t+1$ is a $t \times |I|$ matrix storing the user's ratings on items, with t the number of days starting from the day when the first rating was made and recorded. Each row of I_u^t, which the study denote by ΔI_u^τ, stores the ratings (in the range 1-3) that u made on day τ and stores *empty* for the items not rated on that day. Use 0 to denote *empty*.

Data filtering options: Data filtering options are the settings or actions users can make to manipulate their I_u^t. The items are split into different categories $\mathcal{C}$. Each element C_j of $\mathcal{C}$ is a set of item indices. Each user can decide the ratings on which categories of items are shared with the system. In addition, they can choose to delete their data before the most recent N days, N a number of their choice. The first several rows of I_u^t are deleted until I_u^t only has N rows left.

The game consists of the following components.

- state $(I_u^t, \mathcal{C}_u^t \subseteq \mathcal{C})$. $\mathcal{C}_u^t$ is a subset of $\mathcal{C}$ denoting the categories of data that u chooses to share. For each column i of I_u^t, if item $i \in C_j$ for some $C_j \in \mathcal{C}_u^t$, the last non-empty value is stored in the ith entry of a vector $x_u^t \in \mathbf{R}^{|I|}$. This means x_u^t stores the most up-to-date ratings of u on the day t for the items that are not rated or filtered. x_u^t stores 0 for the empty entries. x_u^t is not the state, just what's visible to the system.

- New data ΔI_u^{t+1}. To generate data for the current day $t+1$, the study generates a revealing vector of length $|I|$ by assigning 0 or 1 to each entry based on a Bernoulli distribution with

probability q for 1. The study take the element-wise product of this revealing vector with an underlying rating vector $\tilde{I_u}$ for user u to get ΔI_u^{t+1}.

- the one-day reward

$$r_u^{t+1}(\Delta I_u^{t+1}, x_u^t; x_0^t, x_1^t, ..., x_{|U|}^t) = \lambda f_u^{t+1} + (1-\lambda) g_u^{t+1} \tag{1}$$

. f_u^{t+1} represents the data utility given by a function $f(\Delta I_u^{t+1}, s(x_0^t, x_1^t, ..., x_{|U|}^t))$ where s is the system prediction, and g_u^{t+1} is the privacy score or data minimization score given by a function $g(x_u^t)$.

- $s(x_0^t, x_1^t, ..., x_{|U|}^t)$ On day $t+1$, x_u^t of all users u are stacked to be $X^t \in \mathbf{R}^{|U| \times |I|}$. The system fits a matrix factorization model and predicts each user's preference. The prediction for each user is a vector s_u^{t+1}.

- $f(\Delta I_u^{t+1}, s_u^{t+1})$: $\mathbf{R}^{|I|} \to \mathbf{R}$ is given by

$$f(\Delta I_u^{t+1}, s_u^{t+1}) = a - RMSE(\Delta I_u^{t+1}, s_u^{t+1}) \tag{2}$$

with a a constant.

- $g(x_u^t) : \mathbf{R}^{|I|} \to \mathbf{R}$ is given by

$$g(x) = c \times \text{proportion of nonzero entries of } x \tag{3}$$

for some constant c. Here the study assume each item is equally important in terms of utility and privacy.

- action $a_u^{t+1} \in (\mathcal{A}_{cat} \times \mathcal{A}_{time})$. The user takes action after getting the reward. They can set $\mathcal{C}_u^{t+1}$ and delete data before N days ago (permanently). Action is thus a conditional distribution $a_\pi(\mathcal{C}_u^{t+1}, N_u^{t+1} | I_u^t, \mathcal{C}_u^t, f_u^{t+1}, g_u^{t+1})$.

- transition probability

$$p(I_u^{t+1}, C_u^{t+1}, | I_u^t, C_u^t, a_u^{t+1}) = p(I_u^{t+1} | I_u^t, a_u^{t+1}) \tag{4}$$

Note that since $P(I_u^{t+1} | I_u^t, a_u^{t+1}, \Delta I_u^{t+1})$ is deterministic because it's simply appending ΔI_u^{t+1} to I_u^t and deleting the out-of-date data based on a_u^{t+1}, and $P(\Delta I_u^{t+1} | I_u^t, a_u^{t+1}) = P(\Delta I_u^{t+1})$

is uniform, the transition probability is just the average of $P(I_u^{t+1}|I_u^t, a_u^{t+1}, \Delta I_u^{t+1})$ for all possible ΔI_u^{t+1}. Hence, the probability of the new row $p(I_{u;t+1,i}^{t+1} = \tilde{I}_{u;i}|I_u^t, a_u^{t+1}) = q$ for every item i. The probability for the past data is deterministic $p(I_{u;t':t}^{t+1} = I_{u;t':t}^t|I_u^t, a_u^{t+1}) = 1$, where t' is $t - N(a_u^{t+1})$, the most distant day with available data.

- $$p(x_{u;i}^{t+1} = \tilde{I}_{u,i}|x_{u;i}^t = \tilde{I}_{u,i}, a_u^{t+1}) = q + (1-q)\frac{1-(1-q)^{N(a_u^{t+1})-1}}{1-(1-q)^{N(a_u^t)}} \tag{5}$$

  If item i is not in filtered categories. This is what I calculated from the transition probability. The study might use it to simplify the iteration.

- Strategy π. The strategy is a conditional probability given by $\pi(a_u^{t+1}|I_u^t, \mathcal{C}_u^t)$.

Implementation. The study implement this environment using a gymnasium. For simplicity and efficiency in training, the environment doesn't store the whole data history I_u^t but simulates the transition of x_u^t. The study uses $N_u^t, \tilde{I}_u^t$ and a multi-binary revealing vector indicating the non-zero entries of x_u^t to replace I_u^t in the observation.

3.2 Optimal Strategies

Problem setup. Recall the stationary model introduced before and consider the case when the system uses an algorithm that learns item representations(and possibly user representations), such as FunkSVD. The item representations thus introduce item clusters K based on a similarity measure. Note that the clusters K (sets of items) are different from the categories C, which are specified in the action space. Items in the same clusters may belong to different categories but are liked only by a similar group of users. Here, the study tries to deduce the optimal strategy for a user u in two different settings: i) when other users take fixed or random actions and ii) when other users are strategic. All users are still influenced by the randomness of the rating-revealing process each day. For any set of shared item ratings I_u, this study refers to the number of items in the neighborhood as **data coverage**, in math, $|\{j|j \in \bigcup_{i \in I_u} B(i, \delta)\}|$ where $B(i, \delta) = \{x|||x - i||_2 < \delta\}$.

3.2.1 Random or Fixed Population

In the case of a random or fixed population, this study proposes that the optimal strategies should be the one that minimize the size and optimize the data coverage of the shared set of data. The following of this session shows the reasoning behind this idea.

High level intuition. Consider the case where there are enough users who provide enough data each day, so without the data shared by player u, the algorithm can already learn the item representation very well, so that the item ratings that player u shares have little effect on the final item representation learned. The major effect of ratings shared by player u is hence for the algorithm to learn a mapping from the item representation to item ratings specific to player u. Each item rating shared by u can ensure small error on items in a neighborhood of some width dependent on the algorithm. In the nontrivial case with some constant a, c, and λ, u should be able to gain reward by sharing the rating of an item with enough items in its neighborhood.

Now for player u, each category of items corresponds to several points in the item representation space, and the neighborhoods of all these points form a cover. The game is to maximize the number of items covered by the categories while minimizing the number of items in the categories.

FunkSVD. As a special case, assume the user-item rating matrix I^t is of rank m and the system uses FunkSVD with n_factor=2 without bias to estimate the matrix.

$$I^t = PQ \tag{6}$$

where $P \in \mathrm{R}^{|U| \times m}$ and $Q \in \mathrm{R}^{m \times |I|}$ store the representation of each user and each item respectively.

Assume the final item representations Q learned aren't changed by the ratings shared by u. Given the ratings that player u shares, called I_u^t, and the set of shared items S, the algorithm is trying to solve a linear system to estimate $P_u = p_u$

$$Q_S^T p_u = I_{u,S}^t \tag{7}$$

where $Q_S^T \in \mathrm{R}^{|S| \times m}$ and $I_{u,S} \in \mathrm{R}^{|S|}$ are the $|S|$ rows or entries of S. We can see that $|S| = m$ items with different representations are needed to get the optimal p_u.

In addition, since I^t is of rank m, there are only m different underlying item representations and the items' representations should form m item clusters centering these underlying representations.

In the case without any noise in the data, the m clusters are just m points. If player u had control on whether each item rating is shared, they should share no more than m items and probably select only some of them based their cluster size. If player u can only decide C_u^t and N, they should choose the categories to base on the additional diversity of representation the categories can contribute and their size and set N to control the probability that items are revealed in each category.

3.2.2 Strategic Population

Now consider the case when other users try to maximize their reward by learning a policy as well. If there exist some users that still provide enough data to the system, the data provided by any strategic user doesn't have a large effect on the item representation, and hence, the interaction between strategic users can be ignored. It becomes the same situation as in a fixed random population, as discussed before.

What is more interesting is when all users are strategic and only care about their own reward. Then there is no guarantee of the quality of the item representation learned by the system and the environment for any user is possibly changing during their exploration. Using Double DQN, this work studies the game with the same number of users of two kinds of underlying preferences in two groups and finds their behavior at a Nash Equilibrium. It further shows that the optimal strategy in this setting shares more data.

4 Results

4.1 Experimentation protocol

The code is publicly available at https://github.com/MarCO-COrle0NE/capstone_2024.

4.1.1 Single Action Test

Motivation & Purpose As mentioned in high-level intuition above, the study here designed a single action test to demonstrate its hypothesis. The Single Action Test means that the same fixed action is used repeatedly throughout the episode (or episodes) to observe the average utility and privacy reward of that action over time. In this case, the study can have a deeper under-

standing of the performance of a specific action in different environments, especially under some extreme conditions. Moreover, the most reasonable environment the study chose provides not only a baseline reference for later policy optimizations but also a sanity check on the complex environment settings.

Experimental Setup In this test, the environment parameters involve two groups of users: the observed player and multiple background (auto) users. For the observed player, the study introduces two key parameters:

- The probability of new data revealing q: This simulates the frequency at which the observed player generates new rating data. For example, a higher q corresponds to more frequent user interactions, akin to how often a user scrolls through short videos on a platform like TikTok.
- The weight of utility λ: Recall that the one-day reward for a user u is given by $r_u^{t+1} = \lambda f_u^{t+1} + (1-\lambda) g_u^{t+1}$, where f_u^{t+1} measures data utility and g_u^{t+1} measures privacy or data minimization. By adjusting λ, the study can control the trade-off between the importance of utility and privacy in the player's objective.

For the background (auto) users, the study controls their behaviors using $\mathcal{A}^0_{\text{auto}}$ and $\mathcal{A}^1_{\text{auto}}$:

- $\mathcal{A}^0_{\text{auto}}$: The probability that each category is shared by the auto users. For example, if $\mathcal{A}^0_{\text{auto}} = 0.7$, then each category is revealed with probability 0.7 independently.
- $\mathcal{A}^1_{\text{auto}} \in$ Random, Parsy, Full: A mechanism that determines the data retention window N^t_{auto} for the auto users. Different realizations of this random process lead to varying historical data lengths over time.

The baseline environment is $q = 0.3, w = 0.5, \mathcal{A}^0_{\text{auto}} = 0.7, \mathcal{A}^1_{\text{auto}} = \textit{Random}$.

Data Generation. The study employs a latent factor model to construct synthetic data that exhibits a meaningful category-level structure to demonstrate the hypothesis. The process begins by sampling two independent latent factors from standard normal distributions:

$$c_2 \sim \mathcal{N}(0,1), \quad c_3 \sim \mathcal{N}(0,1). \tag{8}$$

These factors represent underlying latent dimensions of user-item interactions.

Next, a third latent factor is formed as a linear combination of c_2 and c_3:

$$c_1 = a \cdot c_2 + b \cdot c_3, \tag{9}$$

where the coefficients a and b capture how c_2 and c_3 jointly influence c_1.

These three factors are then assembled into a single latent vector:

$$\mathbf{c} = \text{Stack}[c_1, c_2, c_3], \tag{10}$$

which serves as a compact representation of the latent structure underlying item categories.

Finally, to generate the observed ratings, $\mathbf{c}$ is mapped through a category-to-item transformation, and an additive noise term is introduced:

$$P = \sum(\mathbf{c} \cdot \text{CategoryToItem}) + \text{Noise}. \tag{11}$$

The resulting ratings P reflect both the latent relationships encoded in $\mathbf{c}$ and realistic variability via the noise. This procedure generates a representative dataset for evaluating the proposed recommender framework under controlled, yet informative, conditions.

Experiment Procedure. To evaluate how different parameter configurations influence the observed user's performance, the study runs a series of experiments by varying q, λ, and the auto user parameters $(\mathcal{A}^0_{\text{auto}}, \mathcal{A}^1_{\text{auto}})$. For each selected tuple $(q, \lambda, \mathcal{A}^0_{\text{auto}}, \mathcal{A}^1_{\text{auto}})$, the study proceed as follows:

1. **Environment initialization:** Set the environment parameters according to the chosen values of q, λ, $\mathcal{A}^0_{\text{auto}}$, and $\mathcal{A}^1_{\text{auto}}$. In the implementation, for instance, the study might assign $\mathcal{A}^0_{\text{auto}} = 0.7$ and $\mathcal{A}^1_{\text{auto}} =$ `"random"` for the auto users, while the observed player's q and weight λ (referred to as `weight` in the code) is also fixed for this run.

2. **Episode execution:** With the environment set, the study lets the observed user take actions (C_1, C_2, C_3, N) over multiple runs (referred to as *num_ rounds* in the code) or episodes. At each step, the environment updates the state based on both the observed player's chosen actions and the auto users' probabilistic category sharing and random data retention governed by $(\mathcal{A}^0_{\text{auto}}, \mathcal{A}^1_{\text{auto}})$.

3. **Model training and reward computation:** After each step, the environment trains (or updates) the recommendation model using the currently revealed data from all users. It then computes the observed player's reward $r_u^{t+1} = \lambda f_u^{t+1} + (1-\lambda) g_u^{t+1}$ based on the utility-privacy trade-off defined by λ, as described previously.

4. **Averaging results:** Once a set number of runs is completed, the study computes the average reward of the observed user under the given parameter configuration. This result is stored along with the corresponding $(q, \lambda, \mathcal{A}^0_{\text{auto}}, \mathcal{A}^1_{\text{auto}})$ tuple.

5. **Comprehensive exploration:** By iterating over different values of q, λ, and $(\mathcal{A}^0_{\text{auto}}, \mathcal{A}^1_{\text{auto}})$, the study obtain a collection of average performance outcomes for various environmental conditions.

4.1.2 Double DQN.

Motivation & Purpose. For an optimal strategy in the game studied, actions should depend on the state and not the time. Hence, this study uses a reinforcement learning approach to study the game in both the random population and strategic population case. To directly observe the value function and compare the value of the same action in different states, this study uses Double DQN in this game with finite observation and action space.

Random or fixed population. The setup of training a DQN agent in a random or fixed population environment is the same as the single action test except that a learning agent explores the environment instead of taking a fixed action. We use a Double DQN algorithm implemented in stable_baseline3 [20] and uses an exponential learning rate decay and linear decay for exploration rate ϵ for the ϵ-greedy exploration of the learning agent.

Strategic Population and Nash Equilibrium. To explore the behavior of the population and optimal strategies at Nash Equilibrium, this study adopts a two-stage learning algorithm that learns the value functions for different users and test whether a Nash Equilibrium is reached. Due to time constraint, this project only focus on Nash Equilibrium in the average environment as mentioned in 4.1.1. The users are set to have the same utility preference weight w.

At the first stage, the users are put into groups based on their underlying preference. Users in the same group share a Q-network and a Q-target-network. Each group gets a change to explore in turn. When it explores with ϵ-greedy, other groups have $\epsilon = 0$, which means deterministic

actions. This is to ensure that no two groups explore at the same time. Since each user does not observe the actions taken by other users, other users are seen as part of the environment and simultaneous exploration might lead to biased experience. This is based on the assumption that users in the same group are homogeneous. Let Q_j be the network for the j-th group and G be the total number of groups. Details of the first-stage algorithm is shown in 1 with E the environment, T_{max},T_{learn},T_{target} respectively the maximum timestep, the training frequency, and update frequency of the target networks.

Algorithm 1 MultiDQN1(E,Q_j,G,b_j,e_{max},T_{max},T_{learn},T_{target},α,$\delta\alpha$,ϵ,$\delta\epsilon$,ϵ_{min})

for episode = 1,e_{max} **do**
  initialize E and get initial observation s_0
  $j \leftarrow 0$
  for $t = 0, T_{max}$ **do**
    $j \leftarrow (j+1)\%G$
    for $i = 1, G$ **do**
      $a_{t,i} \leftarrow Q(s_{t,i}, \epsilon)$ (ϵ-greedy) **if** $i = j$ **else** $Q(s_{t,i}, 0)$
    end for
    $s_t, r_t, E \leftarrow E(A, a_t, s_t)$
    $b_j[t] \leftarrow (s_{t,j}$,$a_{t,j}$,$r_{t,j}$,$s_{t+1,j})$
    $Q_i \leftarrow DQN(Q_i, Q_{target,i}, b_i), \forall i$ **if** t mod $T_{learn} = 0$
    $Q_{target,i} \leftarrow Q_i$,$\forall i$ **if** t mod $T_{target} = 0$
  end for
  $\alpha \leftarrow \delta\alpha \cdot \alpha$
  $\epsilon \leftarrow \epsilon - \delta\epsilon$ **if** $\epsilon > \epsilon_{min}$
end for
return $Q_j, \forall j$

Note that stage one does not guarantee a Nash Equilibrium since DQN algorithms lead to pure policies. Hence, a stage-two algorithm is used to finetune a separate Q-network for each user copied from the networks learned in stage-one. Let U be the total number of users. The algorithm is shown in 2.

In stage two, each user take turns to explore for an episode with other users taking their learned policy deterministically. This allows each user to increase their reward if the Nash Equilibrium has not been reached. If all of them stop increasing their reward, they are at a Nash Equilibrium.

Hyper-Parameters. We tuned 9 different hyper-parameters, batch size, learning rate, learning decay, training steps, learning start step, target update frequency, traning frequency, and exploration fraction which controls how fast the exploration rate drops to the minimum. We evaluate the hyperparameters by how training loss and the average reward changes through the training process.

Algorithm 2 MultiDQN2(E,Q_u,U,b_u,e_{max},T_{max},T_{learn},T_{target},α,$\delta\alpha$,ϵ,$\delta\epsilon$,ϵ_{min})

for episode = 1,e_{max} **do**
  for $u = 1, U$ **do**
    initialize E and get initial observation s_0
    for $t = 0, T_{max}$ **do**
      for $i = 1, U$ **do**
        $a_{t,i} \leftarrow Q(s_{t,i}, \epsilon)$ (ϵ-greedy) **if** $i = u$ **else** $Q(s_{t,i}, 0)$
      end for
      $s_t, r_t, E \leftarrow E(A, a_t, s_t)$
      $b_u[t] \leftarrow (s_{t,u}, a_{t,u}, r_{t,u}, s_{t+1,u})$
      $Q_u \leftarrow DQN(Q_u, Q_{target,u}, b_u)$ **if** t mod $T_{learn} = 0$
      $Q_{target,u} \leftarrow Q_u$ **if** t mod $T_{target} = 0$
    end for
    $\alpha \leftarrow \delta\alpha \cdot \alpha$
    $\epsilon \leftarrow \epsilon - \delta\epsilon$ **if** $\epsilon > \epsilon_{min}$
  end for
end for
return $Q_u, \forall u$

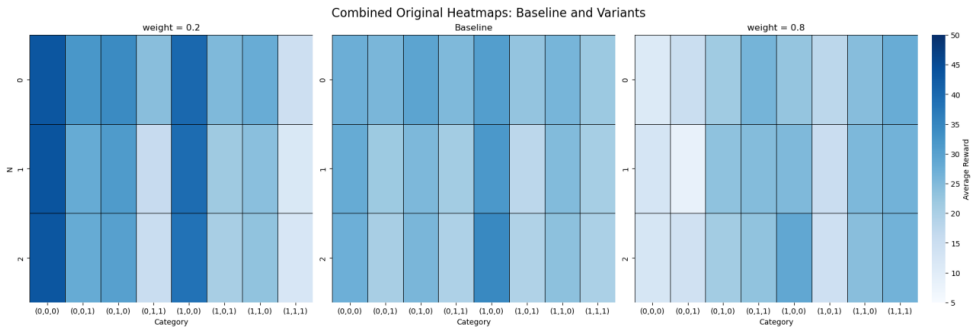

Figure 1: The action performance with different weights

4.2 Results

4.2.1 Single Action Test

First, category one is the more representative category with a smaller item size. Recall that the hypothesis is that the user would choose category one as the optimal strategy. Here are the individual analyses of different parameters. The analysis result was visualized using heat maps.

Performance when w changes : According to Figure 1, action$(1, 0, 0)$ for category revealing is always the optimal strategy because its column is always darker than others except in an extreme condition. Further applications with different user's privacy preferences can be based on this.

Performance when $\mathcal{A}^0_{\text{auto}}$ changes : According to Figure 2, similarly, action$(1, 0, 0)$ is still the optimal choice. While more probability for auto-players to reveal, more data are provided for

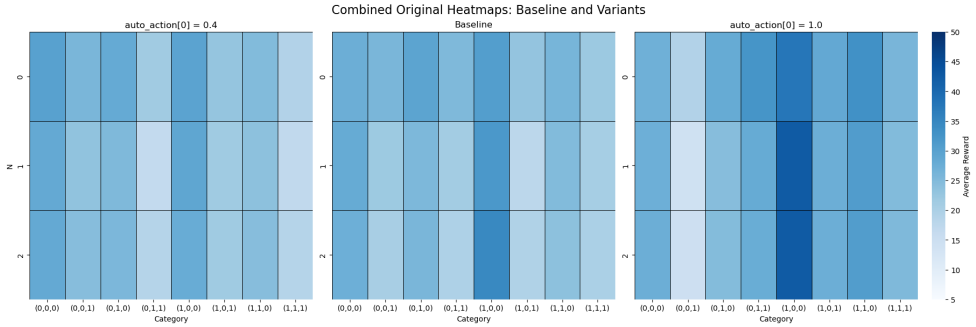

Figure 2: The action performance with different $\mathcal{A}^0_{\text{auto}}$

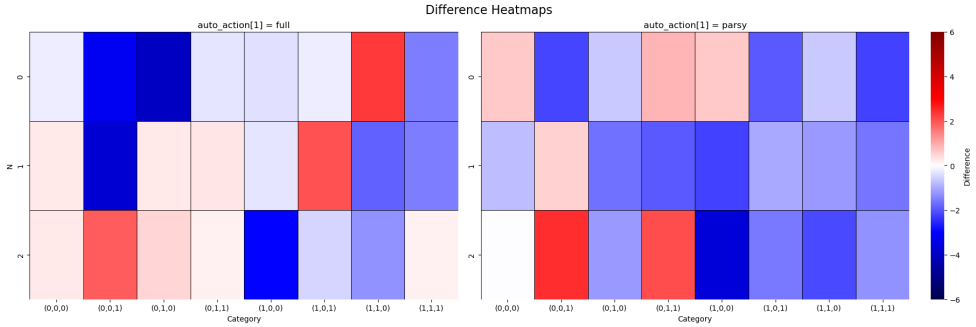

Figure 3: Enter Caption

SVD to predict. In this case, fewer items revealed are needed for accurate prediction, and a representative category is preferred.

Performance when $\mathcal{A}^1_{\text{auto}}$ changes : According to Figure 3, in either extreme case, the representative category would have less reward. Furthermore, extreme actions such as closing or revealing all would result in a higher reward.

Performance when q changes : According to Figure 4, as q gets larger, more data are generated, for the same reason above, action $(1, 0, 0)$ would have a higher reward.

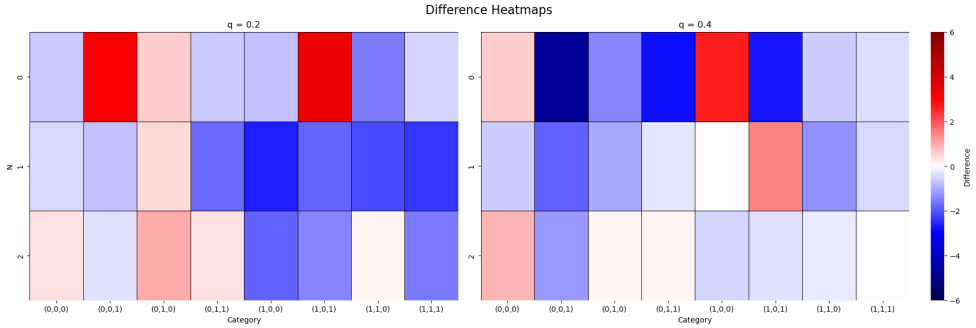

Figure 4: Enter Caption

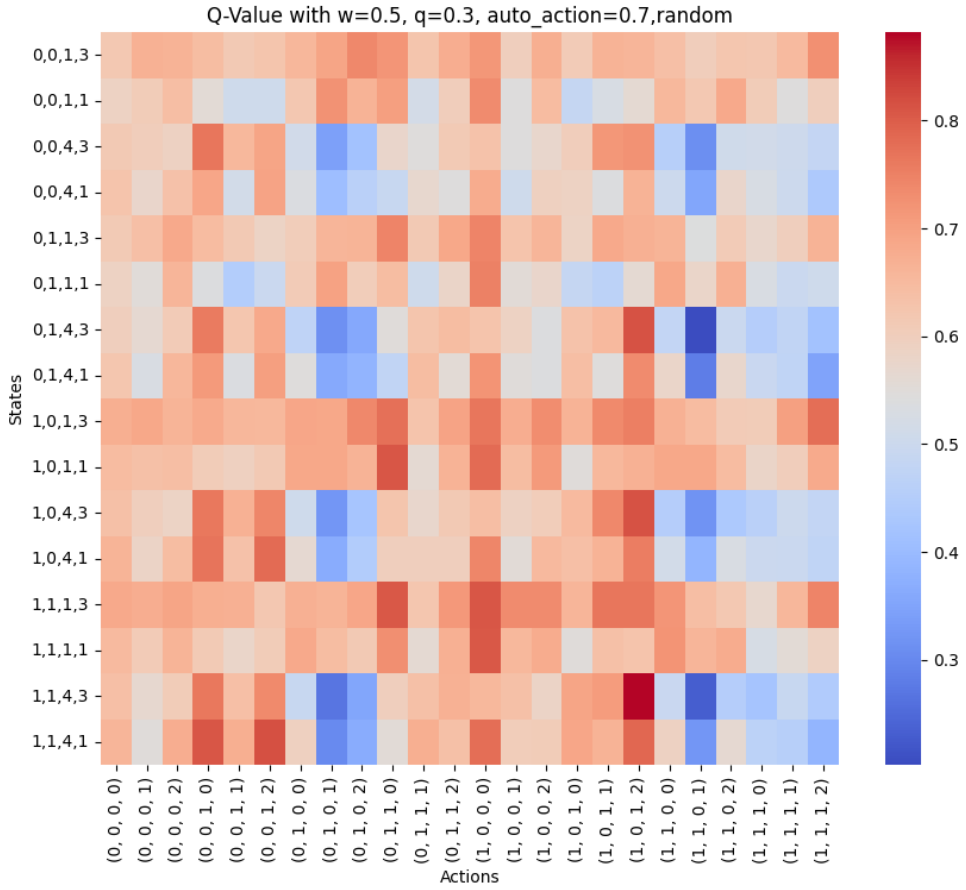

Figure 5: The Q-value (heat) learned in random population as a function of all actions (x-axis) and different categories of states (y-axis). The four digits on y-axis respectively represents a) # of items of Feature 1 in Category 1, b) # of items of Feature 2 in Category 1, c) # of items of Feature 1 in Category 2, d) # of items of Feature 2 in Category 3. To deduce a pure policy from the graph, the policy should select the action of the highest Q-value (shown in the warmest color) for each state.

4.2.2 DQN - Random or Fixed Population

Figure 5 shows the Q-value learned in the average environment with the utility preference $w = 0.5$, data generating frequency $q = 0.3$, and the other users taking automatic actions with parameters (0.7,random). Based on the assumption that items of the same underlying representation in the same category are equalevant, the states are put into categories to remove redundant information. Comparing the columns, we can observe similar results as the single action test that sharing Category 1 is in general the best strategy across states. However, in the states when most items of Feature 1 are revealed already (when the third digits is four, equal to the total number of items in Category 2), sharing Category 3 without Category 2 can sometimes lead to higher reward. This is because the population do not provide enough data in the average environment and sharing more item in Category 3 (all of Feature 2) lead to perfect accuracy on the last item. Nevertheless,

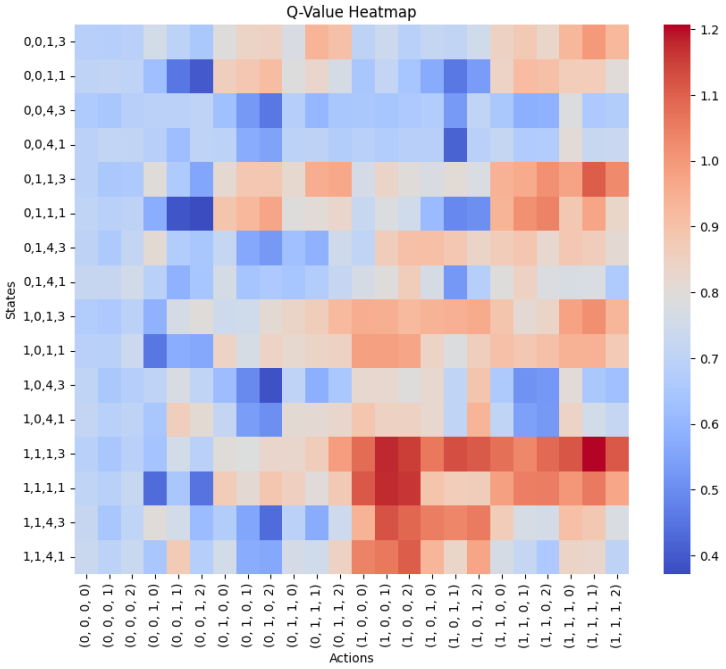

Figure 6: Value function when $q = 0.2$

this should rarely happen in practice since few people could rate the majority of items of a specific feature.

The results of experiments studying different environment parameters have similar results as the single action test. Figure 6 shows the result when $q = 0.2$. Compared with the average environment, a smaller q moves the best action significantly to the right lower corner, which means the strategy is to share more. This goes in line with the hypothesis that suggest sharing more when data is scarce.

4.2.3 DQN - Strategic Population

Figure 7 shows the Nash Equilibrium in an environment with two different groups, each with 5 users. The results of the stage-two algorithm do not show a higher reward. Compared with the case of a random population, users tend to share more data together in order to achieve higher rewards. Details supporting this are that the warmest grids are concentrated in the lower right corner, where more data are revealed and more categories are shared.

5 Discussion

Challenges The formulation of the game model requires efforts to balance the connection with the real world and the insights the study can extract from the model. The study supports this

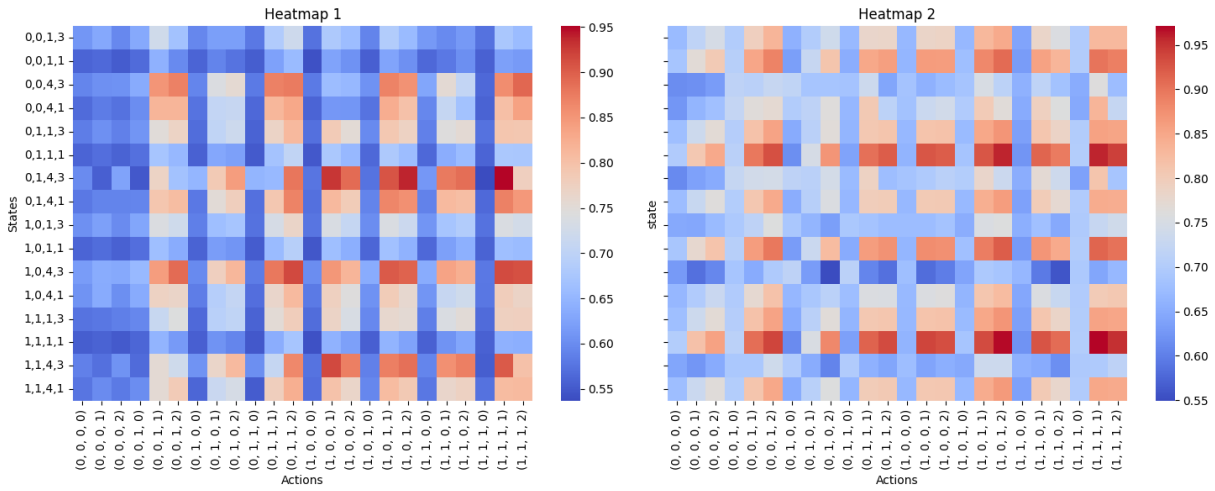

Figure 7: Value functions for two groups in an environment.

with much time spent exploring practical settings in the real world and analyzing how to turn them into abstract features in the model. Compared with other works using a game approach for operation research, our domain focus has an impact on most people in society, and the model in this study is highly realistic.

Limitations Due to time constraints, there are several limitations.

- Revealing too much data might restrict the kind of items to be tested.
- Limitation of DQN. The DQN can provide some insights into value, but the use of DQN is not totally suitable in a multi-agent learning environment due to the possibility of missing certain mixed-policy Nash Equilibrium.
- The hypothesis and results are general. Connection with real-world data was not studied (a conclusion is not directly applicable).
- Privacy risk measures other than data minimization. The study expects the privacy risk to be more than how much data is shared. Some data pieces should be more risky to share. We encourage future studies to cover this.

6 Conclusion

This paper introduces a game-theoretic framework to model and analyze users' data-sharing behaviors with a central unit. This formulation captures the strategic interactions between users, providing valuable insights into how individual decisions impact collective outcomes. The study

investigates the optimal strategies and value functions in scenarios where other users adopt random or fixed actions, offering a comprehensive understanding of the dynamics involved. Furthermore, our analysis extends to derive the conditions for a Nash equilibrium, illustrating the balance between individual incentives and group dynamics.

Based on our findings, the study offers actionable advice for effective data-sharing strategies. First, sharing a large volume of data with similar features provides diminishing returns in terms of rewards. Instead, prioritizing the diversity of shared features can significantly enhance the outcomes. Additionally, when other users contribute less data, a moderate increase in data sharing—particularly of the same features can optimize the reward.

These insights emphasize the importance of strategic data sharing, balancing privacy concerns with service quality. Our study provides a theoretical foundation and practical guidance for designing systems that foster responsible and efficient data-sharing practices. There are at least two major directions that future works can focus on. The first is to refine the definition of the privacy score function, simulate real-world data, etc. to build a stronger connection with reality. The second is to dig deeper into the Nash Equilibrium.

References

[1] D. J. Solove, "Introduction: Privacy self-management and the consent dilemma," *Harvard Law Review*, vol. 126, no. 7, pp. 1880–1903, 2013. [Online]. Available: http://www.jstor.org/stable/23415060

[2] P. Regulation, "Regulation (eu) 2016/679 of the european parliament and of the council," *Official Journal of the European Union*, 2016.

[3] A. J. Biega, P. Potash, H. D. I. au2, F. Diaz, and M. Finck, "Operationalizing the legal principle of data minimization for personalization," 2020. [Online]. Available: https://arxiv.org/abs/2005.13718

[4] K. U. Fallatah, M. Barhamgi, and C. Perera, "Personal data stores (pds): A review," *Sensors*, vol. 23, no. 3, 2023. [Online]. Available: https://www.mdpi.com/1424-8220/23/3/1477

[5] C. Dwork, "Differential privacy," in *Automata, Languages and Programming*, M. Bugliesi, B. Preneel, V. Sassone, and I. Wegener, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 1–12.

[6] C. Gentry, "A fully homomorphic encryption scheme," Ph.D. dissertation, Stanford, CA, USA, 2009, aAI3382729.

[7] P. Samarati and L. Sweeney, "Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression," 1998. [Online]. Available: https://api.semanticscholar.org/CorpusID:2181340

[8] G. Zyskind, O. Nathan, and A. S. Pentland, "Decentralizing privacy: Using blockchain to protect personal data," in *2015 IEEE Security and Privacy Workshops*, 2015, pp. 180–184.

[9] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," 2023. [Online]. Available: https://arxiv.org/abs/1602.05629

[10] S. Goryczka, L. Xiong, and V. Sunderam, "Secure multiparty aggregation with differential privacy: a comparative study," in *Proceedings of the Joint EDBT/ICDT 2013 Workshops*, ser. EDBT '13. New York, NY, USA: Association for Computing Machinery, 2013, p. 155–163. [Online]. Available: https://doi.org/10.1145/2457317.2457343

[11] S. Liu, Z. Wang, Y. Chen, and Q. Lei, "Data reconstruction attacks and defenses: A systematic evaluation," 2024. [Online]. Available: https://arxiv.org/abs/2402.09478

[12] Y. Shen, C. Shepherd, C. M. Ahmed, S. Shen, X. Wu, W. Ke, and S. Yu, "Game-theoretic analytics for privacy preservation in internet of things networks: A survey," *Engineering Applications of Artificial Intelligence*, vol. 133, p. 108449, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0952197624006079

[13] J. Wang, N. He, F. Mei, D. Tian, and Y. Ge, "Optimization and non-cooperative game of anonymity updating in vehicular networks," *Ad Hoc Networks*, vol. 88, pp. 81–97, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1570870518303846

[14] J. McAuley, C. Targett, Q. Shi, and A. van den Hengel, "Image-based recommendations on styles and substitutes," 2015. [Online]. Available: https://arxiv.org/abs/1506.04757

[15] H. Polat and W. Du, "Privacy-preserving collaborative filtering," *International Journal of Electronic Commerce*, vol. 9, no. 4, pp. 9–35, 2005. [Online]. Available: https://doi.org/10.1080/10864415.2003.11044341

[16] S. Benkessirat, N. Boustia, and R. Nachida, "A new collaborative filtering approach based on game theory for recommendation systems," *Journal of Web Engineering*, vol. 20, no. 2, p. 303–326, Mar. 2021. [Online]. Available: https://journals.riverpublishers.com/index.php/JWE/article/view/1287

[17] H. Wen, L. Yang, M. Sobolev, and D. Estrin, "Exploring recommendations under user-controlled data filtering," in *Proceedings of the 12th ACM Conference on Recommender Systems*, ser. RecSys '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 72–76. [Online]. Available: https://doi.org/10.1145/3240323.3240399

[18] T. Zhang, R. Venkatesaramani, R. K. De, B. A. Malin, and Y. Vorobeychik, "A game-theoretic approach to privacy-utility tradeoff in sharing genomic summary statistics," 2024. [Online]. Available: https://arxiv.org/abs/2406.01811

[19] Y. Xie, Y. Ma, J. Shen, and A. Li, "A game theoretic approach toward privacy preserving for mobile learning data sharing," in *2022 2nd International Conference on Consumer Electronics and Computer Engineering (ICCECE)*, 2022, pp. 360–363.

[20] A. Raffin, A. Hill, A. Gleave, A. Kanervisto, M. Ernestus, and N. Dormann, "Stable-baselines3: Reliable reinforcement learning implementations," *Journal of Machine Learning Research*, vol. 22, no. 268, pp. 1–8, 2021. [Online]. Available: http://jmlr.org/papers/v22/20-1364.html