Pareto Efficient Recommendation

Yongfeng Zhang
College of Information and CS
UMass Amherst, MA 01003
yongfeng@cs.umass.edu

Yi Zhang
School of Engineering
UC Santa Cruz, CA 95060
yiz@soe.ucsc.edu

Daniel Friedman
Economics Department
UC Santa Cruz, CA 95060
dan@ucsc.edu

ABSTRACT

A fundamentally important role of the Web economy is Online Service Allocation (OSA) from producers to consumers, such as product allocation in E-commerce, job allocation in freelancing platforms, and driver resource allocation in P2P riding services. Since users have the freedom to choose, such allocations are not provided in a forced manner, but usually in forms of personalized recommendation, where users have the right to refuse.

Current recommendation approaches mostly provide allocations to match the preference of each individual user, in stead of treating the Web application as a whole economic system – where users therein are mutually correlated on the allocations. This lack of global view leads to *Pareto inefficiency* – that we can actually improve the recommendations by bettering some users while not hurting the others, which means that the system did not achieve its best possible allocation. This problem is especially severe when the total amount of each resource is limited, so that its allocation to one (set of) user means that other users are left out.

In this paper, we propose *Pareto Efficient Recommendation* (PER) – that the system provides the best possible (i.e., *Pareto optimal*) recommendations, where no user can gain further benefits without hurting the others. To this end, we propose a *Multi-Objective Optimization* (MOO) framework to maximize the *surplus* of each user simultaneously, and provide recommendations based on the resulting Pareto optima. To benefit the many existing recommendation algorithms, we further propose a *Pareto Improvement Process* (PIP) to turn their recommendations into Pareto efficient ones. Experiments on real-world datasets verify that PIP improves existing algorithms on recommendation performance and consumer surplus, while the direct PER approach gains the best performance on both aspects.

KEYWORDS

Pareto Efficiency; Online Service Allocation; Multi-Objective Optimization; Recommendation; Computational Economics

ACM Reference format:
Yongfeng Zhang, Yi Zhang, and Daniel Friedman. 2017. Pareto Efficient Recommendation. In *Proceedings of The 23rd ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Halfiax, Nova Scotia, Canada, Aug 2017 (KDD'17),* 10 pages.
DOI: 10.475/123_4

1 INTRODUCTION

With the trending of human activities shifting from offline to online, the Web has turned into a whole unified economic system just like our physical world, where users can accomplish various types of daily tasks conveniently.

Like our real-world economic system that has been drawing the attention of economists for centuries, an important functionality of the Web is Online Service Allocation (OSA), which distributes online services from producers to consumers at the speed of internet. For example, E-commerce systems like Amazon distribute normal goods from retailers to users, while house sharing applications like Airbnb distribute housing facilities from hosts to guests.

Because users are granted by law of the freedom to choose, enforced online service allocation is not favorable. As a result, such an allocation process is usually conducted implicitly by personalized recommendation [38], which "suggests" the users to choose particular services.

Since the beginning of modern economics, economists have been taking care of the efficiency of economic systems [15]. The key insight is *Pareto efficiency*, named after economist Vilfredo Pareto (1906) [33] – one of the pioneers of microeconomics, who is also famous for his "80/20 rule" derived from this concept. Pareto efficiency claims that an efficient system should be one in such an optimal status – that no one can gain further benefits without hurting the others. If a system is not Pareto efficient, we can promote it by *Pareto improvements* – to increase the benefits of some users while *not* decreasing the others' – until the system is efficient.

Pareto efficiency is widely considered not only in economic analysis [13], but in various engineering tasks, e.g., electric power distribution [48], network bandwidth allocation [51], and task scheduling in cloud computing systems [46], etc.

However, though as a major form of service allocation on the Web, current recommender systems seldom consider whether the allocations are Pareto efficient or not in an economic sense. Basically, both the content-based [34] and Collaborative Filtering (CF)-based [12, 20] approaches attempt to maximize the degree that user preferences are matched by the recommended items [38], rather than viewing the Web economy as a whole, where the benefits of different users can be mutually correlated. This results in the Pareto inefficiency of the recommendations – that the system is not sufficiently optimized to reach the best status, so that it is still possible to benefit the experience of some users without hurting the others.

In this work, we propose Pareto Efficient Recommendation (PER) for optimal online service allocation. With solid economic theories and real-world user behavior records, we estimate the per-user utility and surplus on each item to measure the user benefits. Based on this, we propose a Multi-Objective Optimization (MOO) framework to maximize the user benefits jointly, so as to find the Pareto optimal solution for efficient service allocation among users. This allocation

KDD'17, Halfiax, Nova Scotia, Canada

DOI: 10.475/123_4

can thus be adopted to provide personalized recommendations in recommender systems.

Over the years, many effective recommendation algorithms have been developed and they are widely applied to practical systems. Although their provided recommendation results may not be Pareto efficient, taking advantage of the many existing algorithms can help to reduce the complexity in practical applications. Therefore, we further propose a Pareto Improvement Process (PIP) on top of any given recommendation algorithm, which promotes the recommendation results to Pareto efficient ones by Pareto improvements.

Results on real-world E-commerce datasets show that, PIP increases both the recommendation performance and allocation efficiency of traditional recommendation algorithms, while a direct PER approach gains the best performance on both aspects.

In the rest of the paper, Section 2 introduces some basic concepts for the work. In Section 3 we propose our Pareto efficient recommendation framework based on MOO, and in Section 4, we further provide the Pareto improvement process to boost the efficiency of existing recommendation algorithms. Some discussions are made in Section 5, with experimental results provided in Section 6. We present the related work in Section 7 and conclusions in Section 8.

2 BASIC CONCEPTS AND DEFINITIONS

In this section, we formally introduce the key concepts of utility, surplus, and Pareto efficiency from economics, which will form the theoretical basis for our framework, and the notations are to be used throughout the following text.

2.1 Utility and Surplus

In economics, *utility* measures one's satisfaction over one or a portfolio of goods/services. It is the basic concept that serves as the underpinning of the *rational choice theory* [11].

Utility $U(q)$ is usually a function of the consumption quantity q, and is inherently governed by the *Law Of Diminishing Marginal Utility* [39], which states that as a person increases the consumption of a product, there is a decline in the marginal utility that he derives from consuming each additional unit of the product, *e.g.*, when forcing a person who is full to consume an additional bread. This gives $U''(q) < 0$, while the marginal utility $U'(q) > 0$.

Economists have introduced various forms for utility, for example, the most representative King-Plosser-Rebelo (KPR) utility:

$$U(q) = a \ln(1 + q) \tag{1}$$

where the parameter a measures the risk aversion as well as the overall lift of the curve.

Based on the theories of rational choice, a consumer would purchase a product/service only if she thinks that the utility she gains from the product is higher than the price that she has to pay, and *surplus* measures the extra amount of satisfaction she gains beyond the paid price in the transaction. Let P be the per-product price, thus the surplus $S(q)$ is:

$$S(q) = U(q) - P \cdot q \tag{2}$$

In modern economics, the concept of surplus has solid theoretical origins from the supply-demand economic analysis framework [5, 6], and it is adopted to quantify the benefits of consumers in economic systems.

2.2 Multi-Objective Optimization

Multi-Objective Optimization (MOO) originally grew out of three areas [29]: economic *equilibrium* and *welfare theories*, *game theory*, and pure mathematics. The mathematical and economic approaches to MOO were united with the inception of game theory by Borel in 1921 [45]. The general MOO problem is posed as follows:

$$\begin{aligned} \underset{\mathbf{x}}{\text{maximize}}\ & \mathbf{F}(\mathbf{x}) = [F_1(\mathbf{x}), F_2(\mathbf{x}), \cdots, F_k(\mathbf{x})]^\intercal \\ s.t.\ & g_r(\mathbf{x}) \le 0,\ r = 1, 2, \cdots, s \\ & h_l(\mathbf{x}) = 0,\ l = 1, 2, \cdots, e \end{aligned} \tag{3}$$

where k is the number of objective functions, s is the number of inequality constraints, and e is the number of equality constraints. $\mathbf{x} \in \mathbb{R}^n$ is a vector of *decision variables*, and $\mathbf{F}(\mathbf{x}) \in \mathbb{R}^k$ is a vector of objective functions $F_i(\mathbf{x})$: $\mathbb{R}^n \to \mathbb{R}$. All vectors are column vectors, and any comparison ($\le$, $\ge$, etc.) of vectors applies to each element.

The objective functions $F_i(\mathbf{x})$ in multi-objective optimization problems are usually mutually correlated, so that the increase on one objective may lead to the decease of another. As a result, given an MOO problem, we are usually interested in its Pareto efficient solutions, which are introduced in the following subsection.

It is worthwhile to note that, as a frequently used optimization formulation, MOO has been applied to plenty of research in recommender systems to jointly maximize mutually unaligned objectives [1, 2, 28, 32, 36, 37, 41, 49], e.g., accuracy, novelty, and diversity. In this work, however, we aim to quantify and maximize the consumer surplus (welfare) with MOO directly from an economic point of view so as to model the online economy, which is different from the motivation of previous MOO-based research.

2.3 Pareto Efficiency and Improvement

Different from single-objective optimization, there is typically no single global solution to an MOO problem. Usually there exist a set of points that all fit a predetermined definition for an optimum. The predominant concept in defining an optimal point is Pareto efficiency (or Pareto optimal) (Pareto 1906) [33], which is,

Definition 2.1. A point $\mathbf{x}^*$ is *Pareto efficient* iff there does not exist another point $\mathbf{x}$, such that $\mathbf{F}(\mathbf{x}) \ge \mathbf{F}(\mathbf{x}^*)$ with at least one $F_i(\mathbf{x}) > F_i(\mathbf{x}^*)$. Otherwise, $\mathbf{x}^*$ is *Pareto inefficient.*

Definition 2.2. Pareto Frontier: The set of all Pareto efficient points is called the Pareto (efficient) frontier.

It would be important to note that Pareto efficiency itself does not imply equality or fairness in an economic system. In fact, a Pareto efficient status can be extremely unbalanced – that a few individuals may take a great share of the benefits, while the remaining ones possess only a small portion. As a result, we may need to take special consideration to find the economically favorable Pareto efficient solution to a system.

Definition 2.3. Pareto Improvement: A status from point $\mathbf{x}_1$ to $\mathbf{x}_2$ is called a Pareto improvement iff $\mathbf{F}(\mathbf{x}_1) \le \mathbf{F}(\mathbf{x}_2)$ with at least one $F_i(\mathbf{x}_1) < F_i(\mathbf{x}_2)$.

Definition 2.4. Utopia Point: A point $\mathbf{F}^\circ \in \mathbb{R}^k$ is a utopia point iff for each $i = 1, 2, \cdots, k$, $F_i^\circ = \underset{\mathbf{x}}{\text{maximize}} F_i(\mathbf{x})$. The corresponding variable $\mathbf{x}_i^\circ$ is the utopia decision variable.

Intuitively, a Pareto improvement is an action conducted in a system that harms no one and helps at least one person. Pareto improvements will keep adding to the economy until it achieves a Pareto optimum (convergence), where no more Pareto improvements can be made [29].

Utopia point (or *ideal point*) is the most ideal solution to an MOO problem, but in general it is unattainable. However, it can help to find the Pareto optimal solution to MOO.

3 PARETO EFFICIENT RECOMMENDATION

We propose our Pareto Efficient Recommendation (PER) framework in this section. We first formalize the problem into an Online Service Allocation (OSA) framework with multi-objective surplus maximization, and further propose methods for computational estimation of personalized utility and surplus with economic theories.

3.1 Problem Formalization

Suppose there exist m users $\{u_1, u_2, \cdots, u_m\}$ in an online economic system, and there are n items $\{v_1, v_2, \cdots, v_n\}$ to be allocated among the users, where the maximum quantity of each item consists the quantity vector $\mathbf{q} = [q_1, q_2, \cdots, q_n]^\intercal$. In the following, we use $1 \le i \le m$ and $1 \le j \le n$ to index the users and items, respectively.

The Online Service Allocation (OSA) [53] problem aims to find an *allocation matrix* $Q = [Q_{ij}]_{m \times n}$, where $Q_{ij} \ge 0$ is the quantity that user u_i is provided with item v_j, under the maximum quantity constraint of $\sum_{i=1}^{m} Q_{ij} \le q_j, \forall j$.

Pareto efficient recommendation thus attempts to provide recommendations according to the Pareto efficient allocation of the online services in terms of per-user benefits (i.e., surplus). Let $\mathbf{Q}_i$ be the allocation vector for user u_i so that $Q = [\mathbf{Q}_1 \mathbf{Q}_2 \cdots \mathbf{Q}_m]^\intercal$, and let $S_i(\mathbf{Q}_i)$ be the surplus that user u_i achieves from his/her allocated items. We aim to solve the following MOO problem,

$$
\begin{aligned}
\underset{Q=[Q_{ij}]_{m\times n}}{\text{maximize}} \quad & \mathbf{S}(Q) = [S_1(\mathbf{Q}_1), S_2(\mathbf{Q}_2), \cdots, S_m(\mathbf{Q}_m)]^\intercal \\
& s.t.\ \mathbf{Q}_i \ge 0,\ i = 1, 2, \cdots, m \\
& \sum_{i=1}^{m} Q_{ij} \le q_j,\ j = 1, 2, \cdots, n
\end{aligned} \tag{4}
$$

which maximizes the surpluses of different users jointly. The model produces Pareto efficient allocations on user benefits, which are taken to make system decisions.

3.2 Model Specification

The key to specifying the mathematical form of Eq.(4) is to calculate the per-user surplus $S_i(\mathbf{Q}_i)$ given an arbitrary allocation matrix $\mathbf{Q}_i$. Let P_j be the price of item v_j, which is pre-known, we have,

$$
S_i(\mathbf{Q}_i) = \sum_{j=1}^{n} S_i(Q_{ij}) = \sum_{j=1}^{n} \Big(U_{ij}(Q_{ij}) - P_j \cdot Q_{ij} \Big) \tag{5}
$$

where $U_{ij}(\cdot)$ is the utility of user u_i on item v_j. It is important to notice that the utility function $U_{ij}(\cdot)$ is personalized – that different users may gain different utilities even on the same quantity of the same item. This is because of the different preferences of users, namely, a product favored by this user may not be quite favored by another one, and this nature serves as the inherent driving power for personalized recommendation. In this work, we estimate the personalized utility $U_{ij}(q)$ per user-item level to represent the personalized user preference on each item. Specifically,

$$
\begin{aligned}
U_{ij}(q) &= a_{ij} \ln(1 + q) \\
&\doteq (\alpha + \beta_i + \gamma_j + \mathbf{u}_i^\intercal \mathbf{v}_j) \ln(1 + q)
\end{aligned} \tag{6}
$$

where the risk aversion parameter a_{ij} is re-parameterized in the spirit of collaborative filtering: $\alpha, \beta_i, \gamma_j$ are the global, user, and item offsets; and $\mathbf{u}_i, \mathbf{v}_j \in \mathbb{R}_+^k$ are the user and item representation vectors, respectively.

As a result, the calculation of per-user surplus boils down to the estimation of per user-item utility function $U_{ij}(q)$.

3.3 Personalized Utility Estimation

Similar to [53], we conduct personalized utility estimation based on the observed user purchasing records. Let q_{ij} be the quantity that user u_i purchased item v_j, and $S_{ij}(q_{ij}) = U_{ij}(q_{ij}) - P_j q_{ij}$ be the surplus that user u_i gains from this purchase. Then the *Law of Zero Surplus for the Last Unit* [14] in economics states that,

$$
\begin{aligned}
\Delta S_{ij}(q_{ij}) &= S_{ij}(q_{ij}) - S_{ij}(q_{ij} - 1) \ge 0 \\
\Delta S_{ij}(q_{ij} + 1) &= S_{ij}(q_{ij} + 1) - S_{ij}(q_{ij}) < 0
\end{aligned} \tag{7}
$$

Intuitively, it means that the reason a user purchases a quantity of q_{ij} on an item, is that he/she can still obtain increased surplus with the last unit, but even a single more unit of purchase will decease the surplus. With this, we maximize the following log-likelihood of observing the whole purchasing records dataset D,

$$
\begin{aligned}
&\underset{\Theta}{\text{maximize}} \log p(D) \\
&= \sum_{i=1}^{m} \sum_{j=1}^{n} I_{ij} \Big(Pr\Big(\Delta S_{ij}(q_{ij}) \ge 0\Big) Pr\Big(\Delta S_{ij}(q_{ij} + 1) < 0\Big) \Big) - \lambda \|\Omega\|_F^2
\end{aligned} \tag{8}
$$

where $\Omega = \{\alpha, \beta_i, \gamma_j, \mathbf{u}_i, \mathbf{v}_j\}_{i=1 j=1}^{m\ n}$ is the parameter set, I_{ij} is an indicator whose value is 1 when user u_i purchased item v_j in the dataset and 0 otherwise, $\lambda > 0$ is the regularization coefficient, and $\mathbf{u}_i, \mathbf{v}_j \in \mathbb{R}_+^k$ are non-negative. We adopt the commonly used sigmoid function to model the probabilities:

$$
\begin{aligned}
Pr\Big(\Delta S_{ij}(q_{ij}) \ge 0\Big) &= \frac{1}{1 + \exp(-\Delta S_{ij}(q_{ij}))} \\
Pr\Big(\Delta S_{ij}(q_{ij} + 1) < 0\Big) &= 1 - Pr\Big(\Delta S_{ij}(q_{ij} + 1) \ge 0\Big)
\end{aligned} \tag{9}
$$

We optimize Eq.(8) by Stochastic Gradient Descent (SGD) to get the optimal parameter set Θ, thus obtain the per user-item utility functions $U_{ij}(q) = (\alpha + \beta_i + \gamma_j + \mathbf{u}_i^\intercal \mathbf{v}_j) \ln(1 + q)$.

3.4 MOO Scalarization

Given utility functions $U_{ij}(q)$, prices P_j, and quantity constraints $\mathbf{q} = [q_1, q_2, \cdots, q_n]^\intercal$, the MOO function in Eq.(4) is determined only on the allocation matrix $Q = [Q_{ij}]_{m \times n}$.

Since its existence, researchers have proposed various approaches to multi-objective optimization, and they can be generally classified into *a priori*, *a posteriori* and *interactive* methods, which involve the preference information from the decision maker priorly, posteriorly, or interactively during the model learning process [29].

In this work, we adopt the *a priori* approach because we aim at an offline learning algorithm for Pareto efficient allocation. To do so, we adopt the scalarization method to transform the multi-objective problem into a single-objective one. When scalarization is done neatly, Pareto optimality of the solutions obtained can be guaranteed [30]. It is proven that minimizing the following function is necessary and sufficient for Pareto optimality of Eq.(4) [29, 50]:

$$\begin{aligned} &\underset{Q=[Q_{ij}]_{m\times n}}{\text{minimize}}\ S(Q) = \sum_{i=1}^{m} w_i\Big(S_i(\mathrm{Q}_i) - S_i^{\circ}\Big)^2 \\ &\quad s.t.\ \mathbf{Q}_i \geq 0,\ \sum_{i=1}^{m} Q_{ij} \leq q_j,\ \forall i, j \end{aligned} \tag{10}$$

where $w_1, w_2, \cdots, w_m \geq 0$ are the weights set by the decision maker, whose relative values reflect the decision maker's preference on the importance of each user. Mathematically, different choices of weights will result in different Pareto efficient optima, although many of them can be extremely unbalanced. Because we treat the benefits of each user equally and do not pose any special preference on specific users, we adopt identical weights $w_i = 1$ in this work.

S_i° is the Utopia point for user u_i, which is determined by the maximum possible surplus of the user. We already have:

$$S_i(Q_{ij}) = U_{ij}(Q_{ij}) - P_j Q_{ij} = a_{ij}\ln(1 + Q_{ij}) - P_j Q_{ij} \tag{11}$$

Let derivative be zero with respect to global quantity limits:

$$S'(Q_{ij}) = \frac{a_{ij}}{1 + Q_{ij}} - P_j = 0,\ w.r.t.\ Q_{ij} \leq q_j \tag{12}$$

We thus have the utopia decision variable Q_{ij}° and utopia point S_i°:

$$\begin{aligned} Q_{ij}^{\circ} &= \min\left(\frac{a_{ij}}{P_j} - 1,\ q_j\right),\ \forall j = 1, 2, \cdots, n \\ S_i^{\circ} &= \sum_{j=1}^{n}\Big(U_{ij}(Q_{ij}^{\circ}) - P_j \cdot Q_{ij}^{\circ}\Big),\ \forall i = 1, 2, \cdots, m \end{aligned} \tag{13}$$

With these components, we plug Eq.(5)(6)(13) and $w_i = 1$ into Eq.(10) for model specification.

3.5 Model Learning

We transform Eq.(10) into a non-constrained problem with penalties, so that it can be conveniently solved with gradient optimization. More concretely, we transform the two constraints into penalty terms added to the objective to penalize infeasible solutions:

$$\begin{aligned} \underset{Q=[Q_{ij}]_{m\times n}}{\text{minimize}}\ L(Q) &= \sum_i \bigg(\sum_j \Big(a_{ij}\ln(1 + Q_{ij}) - P_j \cdot Q_{ij}\Big) - S_i^{\circ}\bigg)^2 \\ &\quad -\lambda_1 \sum_{i,j} \log(Q_{ij}) - \lambda_2 \sum_j \min\Big\{0,\ q_j - \sum_i Q_{ij}\Big\} \end{aligned} \tag{14}$$

where the second term is the log-barrier function that prevents the elements in Q from approaching the nonnegative cone, and the last term is a hinge-loss that penalizes an allocation that violates the quantity constraint. One may also use the hinge-loss $-\lambda_1 \sum_{i,j} \min\{0, Q_{ij}\}$ for $Q \geq 0$, but this would give sparse optima, which are not favored in this work because we rely on the result allocation matrix Q to rank all the items for each user.

In Eq.(14), $\lambda_1, \lambda_2 \in \mathbb{R}_+$ determine the tradeoff between the accuracy of the approximation procedure and the feasibility of the solution. With λ_1 and $\lambda_2 \to \infty$, any violation of the constraints will be greatly penalized and the solution is guaranteed to be in the feasible region. However, the objective term maybe dominated by these two penalty terms, causing the solution to be inaccurate.

```
Algorithm 1: Pareto Efficient Allocation
    Input: Initial penalty parameters $\lambda_1^{(0)}, \lambda_2^{(0)} > 0$, step factor
           $\mu > 1$, initial learning rate $\gamma^{(0)} < 1$, threshold
           $0 < \epsilon_1, \epsilon_2 \ll 1$, dimension $k$ for latent factors $\mathbf{u}_i', \mathbf{v}_j'$
    Output: Pareto efficient allocation matrix $Q^*$
    Initialize $\Theta_*^{(0)} = \{\alpha', \beta_i', \gamma_j', \mathbf{u}_i', \mathbf{v}_j'\}_{i=1 j=1}^{m\ n}$ randomly;
    repeat
        $\Theta_0^{(t)} \leftarrow \Theta_*^{(t-1)}$;
        repeat
            Compute the gradient $\nabla_\Theta L(\Theta_l^{(t)})$ by Eq.(18);
            $\Theta_{l+1}^{(t)} \leftarrow \Theta_l^{(t)} - \gamma^{(t)} \nabla_\Theta L(\Theta_l^{(t)})$;
        until $\|\Theta_{l+1}^{(t)} - \Theta_l^{(t)}\| < \epsilon_1$;
        $\Theta_*^{(t)} \leftarrow \Theta_L^{(t)}$;
        $\lambda_1^{(t+1)} \leftarrow \mu\lambda_1^{(t)}, \lambda_2^{(t+1)} \leftarrow \mu\lambda_2^{(t)}, \gamma^{(t+1)} \leftarrow \gamma^{(t)}/\mu$;
    until $\lambda_1^{(t)} > 1/\epsilon_2$ and $\lambda_2^{(t)} > 1/\epsilon_2$;
    return $Q^* \leftarrow Q(\Theta_*^{(t)})$;
```

To combine the best of two worlds, we propose a sequential minimization framework: for each fixed pair (λ_1, λ_2), we optimize Eq.(14) to find a local minimum solution $Q(\lambda_1, \lambda_2)$ which depends on the current (λ_1, λ_2), then we increase the parameters (λ_1, λ_2) by a factor $\mu > 1$ and restart the optimization with $Q(\lambda_1, \lambda_2)$ as the initial point. To increase the approximation accuracy when λ_1 and λ_2 get larger, we increase the number of steps per iteration, ensuring that the penalty terms will vanish as the solution becomes feasible, and the objective term will get fully optimized. The process is repeated until the current solution is a stationary point.

To apply the sequential minimization framework, we derive the gradient for Eq.(14), where $\nabla_Q^1, \nabla_Q^2, \nabla_Q^3$ denote the gradient for each of the three components:

$$\begin{aligned} \nabla_Q^1 &= \left[2\Big(\sum_j S_i(Q_{ij}) - S_i^{\circ}\Big)\Big(\frac{a_{ij}}{1 + Q_{ij}} - P_j\Big)\right]_{m\times n} \\ \nabla_Q^2 &= -\lambda_1 \left[\frac{1}{Q_{ij}}\right]_{m\times n},\ \nabla_Q^3 = -\lambda_2\Big(\mathbf{1}_m \cdot \mathbb{I}_{\mathbf{q}^\top - \mathbf{1}_m^\top Q < 0}\Big) \end{aligned} \tag{15}$$

where $\mathbf{1}_m$ is a column vector of 1's, and $\mathbb{I}_{\mathbf{q}^\top - \mathbf{1}_m^\top Q < 0}$ is the indicator function $\mathbb{I}_{x<0}$ applied to every element of the vector $\mathbf{q}^\top - \mathbf{1}_m^\top Q$. As a result, the gradient of Eq.(14) will be:

$$\nabla_Q L(Q) = \nabla_Q^1 + \nabla_Q^2 + \nabla_Q^3 \tag{16}$$

However, the huge number of users and items in practical systems makes it computationally infeasible to directly gradient on the allocation matrix Q as a whole. As a result, we still re-parameterize the allocation matrix according to the spirit of collaborative filtering. Specifically, let:

$$Q_{ij} = \alpha' + \beta_i' + \gamma_j' + \mathbf{u}_i'^\top \mathbf{v}_j' \tag{17}$$

where $\Theta = \{\alpha', \beta_i', \gamma_j', \mathbf{u}_i', \mathbf{v}_j'\}_{i=1 j=1}^{m\ n}$ is the parameter set, and $\mathbf{u}_i', \mathbf{v}_j' \in \mathbb{R}_+^k$ are latent factorization parameters. As a result, Q becomes an intermediate parameter and the gradient on the low-dimensional parameters Θ is:

$$\nabla_\Theta L(\Theta) = \nabla_Q L(Q) \cdot \nabla_\Theta Q = (\nabla^1_Q + \nabla^2_Q + \nabla^3_Q) \cdot \nabla_\Theta Q \tag{18}$$

Once we obtain a closed form solution to compute the gradients, we can apply various kinds of unconstrained minimization algorithms, e.g., gradient descent, conjugate gradient or L-BFGS [3]. For illustrative purposes, we provide an algorithm using gradient descent in Algorithms 1. Assuming the algorithm will perform τ iterations in each inner loop, the time complexity of Algorithm 1 is $O(\tau mn \log_\mu(\frac{1}{\lambda\epsilon}))$.

With the Pareto efficient allocation matrix Q^* produced by the algorithm, we can thus provide the personalized recommendation list for each user u_i by ranking the items v_j in descending order of the quantity Q^*_{ij} allocated to him/her.

4 PARETO IMPROVEMENT PROCESS

Researchers have development many successful personalized recommendation algorithms, which are widely applied to various practical systems. Taking advantages of their insightful designs can be beneficial to both the practitioners and recommendation results.

Although they do not primarily consider Pareto efficiency of the recommendations, their results can be promoted into Pareto efficient ones, which improvements the user benefits meanwhile keeps a satisfactory recommendation performance. To this end, we propose a Pareto Improvement Process (PIP) in this section.

We first construct the replication allocation matrix $\hat{Q}$ from the recommendation results of a given recommendation algorithm. Let the top-K recommendation list for user u_i given by the algorithm be $R_i = \{v_{i_1}, v_{i_2}, \cdots, v_{i_K}\}$, we construct the corresponding recommendation vector $\mathbf{r}_i = [\mathbb{I}_{v_j \in R_i}]_{n\times 1}$, where the j-th element is 1 if the j-th item is in the recommendation list, and 0 otherwise. With the vectors for all users, we have $\hat{Q} = [\mathbf{r}_1 \mathbf{r}_2 \cdots \mathbf{r}_m]^\top$.

Note that $\hat{Q}$ may not satisfy the quantity constraint $\mathbf{q} = [q_1, q_2 \cdots q_n]^\top$. To make it feasible, we examine each column of $\hat{Q}$, and if $\sum_{i=1}^m \hat{Q}_{ij} > q_j$, we only retain a total number q_j of 1's in that column. These 1's correspond to the users in whose recommendation list R_i the item v_j is ranked in higher position, i.e., we allocate the limited number of item v_j to those users who better prefer the item. To help with a clearer understanding, we present the procedure in Algorithm 2.

Based on the replication allocation matrix $\hat{Q}$, we calculate the current benefit $\hat{S}_i = S_i(\hat{\mathbf{Q}}_i)$ for each user u_i, and further solve the following Pareto improvement problem:

$$\begin{aligned} &\underset{Q=[Q_{ij}]_{m\times n}}{\text{maximize}} \ \mathbf{S}(Q) = [S_1(\mathbf{Q}_1), S_2(\mathbf{Q}_2), \cdots, S_m(\mathbf{Q}_m)]^\top \\ &s.t.\ \mathbf{Q}_i \geq 0,\ \sum_{i=1}^m Q_{ij} \leq q_j,\ S_i(\mathbf{Q}_i) \geq \hat{S}_i,\ \forall i, j \end{aligned} \tag{19}$$

where the third constraint is incorporated to guarantee Pareto improvements.

Similarly, Eq.(19) can be scalarized and further converted into the following non-constrained optimization problem:

$$\begin{aligned} \underset{Q=[Q_{ij}]_{m\times n}}{\text{minimize}}\ \hat{L}(Q) = &\sum_i \left(S_i(\mathbf{Q}_i) - S_i^\circ\right)^2 - \lambda_1 \sum_{i,j} \log(Q_{ij}) \\ - \lambda_2 &\sum_j \min\left\{0, q_j - \sum_i Q_{ij}\right\} - \lambda_3 \sum_j \min\left\{0, S_i(\mathbf{Q}_i) - \hat{S}_i\right\} \end{aligned} \tag{20}$$

where the gradient for the last term is:

$$\nabla^4_Q = -\lambda_3 \left(\mathbb{I}_{\mathrm{S}(Q)-\hat{\mathrm{S}}<0} \cdot \mathbf{1}_n^\top\right) \tag{21}$$

With the gradients $\nabla_Q \hat{L}(Q) = \nabla^1_Q + \nabla^2_Q + \nabla^3_Q + \nabla^4_Q$ and $\nabla_\Theta \hat{L}(\Theta) = \nabla_Q \hat{L}(Q) \cdot \nabla_\Theta Q$, we can still adopt Algorithm 1 for model learning, and take the output Pareto efficient allocation matrix for personalized recommendation.

Algorithm 2: REPLICATION MATRIX CONSTRUCTION

```
Input: Top-$K$ recommendation list
       $R_i = \{v_{i_1}, v_{i_2}, \cdots, v_{i_K}\}$ for each user $u_i$, quantity
       constraint $\mathbf{q} = [q_1, q_2, \cdots, q_n]^\top$
Output: Replicate allocation matrix $\hat{Q}$
for each user $u_i$ ($i \leftarrow 1$ to $m$) do
    $\mathbf{r}_i \leftarrow [\mathbb{I}_{v_j \in R_i}]_{n\times 1}$; //identification vector
    $\mathbf{p}_i \leftarrow [\infty]_{n\times 1}$; //ranking position vector
    for each item $v_j$ ($j \leftarrow 1$ to $n$) do
        if $v_j \in R_i$ and $v_j \equiv v_{i_k}$ then
            $\mathbf{p}_{ij} \leftarrow k$; //record the ranking position
$\hat{Q} \leftarrow [\mathbf{r}_1, \mathbf{r}_2, \cdots, \mathbf{r}_m]^\top$;
for each item $v_j$ ($j \leftarrow 1$ to $n$) do
    if $\sum_{i=1}^m \hat{Q}_{ij} > q_j$ then
        Rank users ($u_i$'s) in ascending order of $\mathbf{p}_{ij}$;
        Keep $\hat{Q}_{ij} = 1$ for top-$q_j$ users and change the
        remaining to 0;
return $\hat{Q}$;
```

5 DISCUSSIONS

We further discuss some of the properties of the Pareto efficient recommendation framework and Pareto improvement process.

We can see that when $q_j = \infty$ $(\forall j)$ in Eq.(4), the quantity constraint components in Eq.(10)(14) (19)(20) all vanish, and $\nabla^3_Q = 0$ in Eq.(15). As a result, the model learning process automatically turns into a non-constrained one. Besides, the surplus $S_i(\mathbf{Q}_i)$ that each user achieves from an allocation are no longer mutually correlated in this case, thus the Pareto efficient model learning actually maximizes the surplus of each user independently, and reaches the utopia points. As a result, the quantity constraints (i.e., limited resource allocation) serve as the critical component to bridge the relationship of different users, because the allocation of an item to one user may imply that other users are neglected, which is similar to our physical economic world.

Except for the consumer surplus (i.e., utility beyond price: $U(q) - Pq$) that we have considered in this work, economists have also studied producer surplus, i.e., price beyond cost: $Pq - cq$, where c is the per-item cost. In this work, we do not incorporate producer surplus into consideration for total surplus maximization (as [53]) for two reasons, 1) the producer surplus will mathematically be a fixed value when all items are allocated out, and 2) the price component (Pq) offsets when adding the producer and consumer surplus for total surplus maximization, as a result, some users may be sacrificed for the maximization of collective interest (total surplus), which is philosophically not favored. In this work, however, we guarantee that each user benefits and no one has to sacrifice for the whole economic system.

As stated before, the weight parameters w_i in Eq.(10) determine the distribution of benefits among users, and different choices of w_i lead to different Pareto efficient allocations. Although we choose to treat users equally by setting identical w_i's in this work, we can actually control the distribution of benefits among different users with different weights, so as to achieve targeted (but still Pareto efficient) allocations for specific business intelligence goals in real-world systems.

6 EXPERIMENTS

In this section, we conduct extensive experiments to evaluate the performance of both the Pareto Efficient Recommendation (PER) and Pareto Improvement Process (PIP), in terms of recommendation performance, economic efficiency, and Pareto efficiency.

6.1 Dataset Description

We take the user purchasing records dataset from Shop.com for experiments, because two of the most important information sources needed in our framework are the price of items and quantity of purchasing, which are absent in many of the other datasets. To avoid the problem of cold-start [24, 52] so as to focus on our key research target of Pareto efficient recommendation, we select those users and items of at least five purchasing records, which is a frequently adopted pre-processing method in previous work [23, 24, 47]. Some statistics of our dataset are summarized in Table 1.

Table 1: Statistics of the Shop.com dataset

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

For experimental purpose, we randomly select 75% of the transactions from each user to construct the training set, and adopt the remaining transactions as testing set for evaluation. This amounts to around $100k$ transactions from $34k$ users towards $30k$ items in the testing set.

6.2 Experimental Setup

After careful parameter selection, we set the initial penalty parameter $\lambda_1^{(0)} = \lambda_2^{(0)} = 0.1$, step factor $\mu = 2$, initial learning rate $\gamma^{(0)} = 0.1$, threshold $\epsilon_1 = \epsilon_2 = 0.1$, and dimension of latent factors $k = 20$ in Algorithm 1. In the experiments, we aim to provide and evaluate top-N recommendation lists for users, where N runs from 1 to 50. As a result, we set the length of recommendation list $K = 50$ when constructing the replication allocation matrix in Algorithm 2, so that the baselines get fair (or even better) treatment.

When estimating the personalized utility function in Eq.(8), we set $\lambda = 0.05$ and the dimension of representation vectors $\mathbf{u}_i, \mathbf{v}_j \in \mathbb{R}_+^k$ as $k = 20$, because we find that 20 factors are sufficiently enough to stable the model performance.

We take the following representative and state-of-the-art recommendation algorithms for performance comparison:

• **NMF**: **N**on-negative **M**atrix **F**actoration [22], which is a representative and one of the most frequently used matrix factorization approach for personalized recommendation. To apply, we construct the user-item purchasing quantity sparse matrix and predict the missing values, based on which to provide personalized recommendation list for each user in descending order of the predictions.

• **BPRMF**: **B**ayesian **P**ersonalized **R**anking with **M**atrix **F**actorization [35] that is one of the state-of-the-art approaches for ranking-based recommendation. In implementation, we conduct balanced negative sampling on un-purchased items for model learning.

• **TSM**: **T**otal **S**urplus **M**aximization [53] approach that maximizes the total (i.e., consumer plus producer) surplus of an economic system for service allocation and recommendation. It is a state-of-the-art economic surplus-based approach for recommendation.

For each approach we carefully tune the parameters to achieve the best performance. In NMF and BPRMF, we select 20 factors with regularization coefficient $\lambda = 0.1$, and for TSM we also use 20 factors and use $\lambda = 0.05, \eta = 5$ for regularization. In the following, we use PER to represent our Pareto Efficient Recommendation approach proposed in this work. Besides, we also apply our Pareto Improvement Process (PIP) on three of the baselines for evaluation, which are denoted as PIP-NMF, PIP-BPRMF, and PIP-TSM in the following, respectively.

We adopt different evaluation metics for different experimental tasks, which will be exposited in detail in each of the following subsections.

6.3 Recommendation Performance

We first evaluate the performance on personalized recommendation for each approach. To do so, we construct top-N recommendation list for each user in the testing set based on each algorithm, and take the Conversion-Rate@N (CR@N) for evaluation.

For a set of testing users and the top-N recommendation list for each of them, CR@N is the percentage of lists that 'hit' the purchase records in the testing set of the target user. In our experiment, N runs from 1 to 50 with a step of 5. For each user in the testing set, there are as many as $30k$ candidate products for recommendation, and all the candidate products are present in the training dataset. For computational efficiency, we randomly select 1000 users to evaluate average CR at each time, and the CR performance of 30 testing rounds are averaged to provide the final evaluation results.

Figure 1 shows the experimental results of each algorithm, and more specific numbers on typical choices of N are shown in Table 2. We see that our PER approach generally gains better performance than traditional algorithms (NMF and BPRMF). On considering that the key different of PER from previous algorithms is to model user preferences with economic basis (personalized utility and surplus integrating the consideration of price and quantity), this implies the effectiveness of principled economic basis in business intelligence.

Another observation is that PER also outperforms TSM, which is interesting because the allocation results produced by TSM is in fact also a Pareto efficient solution, while the difference is that TSM takes producer surplus into consideration for allocation of benefits. This observation verifies the economic intuition that producers and consumers have conflicts of interest in the allocation of total internet welfare (surplus), so that each party has to "steal" surplus from the other side so as to benefit themselves. As a result, maximizing the benefits of consumers alone (instead of together with producers) helps consumers to gain better experience.

Table 2: Evaluation on averaged Conversion Rate (CR@N), Consumer Surplus (CS@N), Percentage Utopia Surplus (PUS@N) on Top-N recommendation lists, where $N = 1, 5, 10, 20$. In each column, the improvement from a^i to b^j is significant at 0.05 level for $i < j$, but numbers from different columns are not comparable.

N	1			5			10			20			Q
Algorithm	CR(%)	CS($)	PUS(%)	CR(%)	CS($)	PUS(%)	CR(%)	CS($)	PUS(%)	CR(%)	CS($)	PUS(%)	PUS$_Q$(%)
NMF	0.06^1	4.65^1	0.001^1	0.10^1	15.52^1	0.003^1	0.10^1	27.86^1	0.008^1	0.20^1	48.15^1	0.011^1	/
BPRMF	0.10^1	8.84^1	0.001^1	0.24^2	20.54^1	0.006^1	0.28^2	36.62^1	0.013^1	0.51^3	72.04^1	0.014^1	/
TSM	0.20^1	62.25^2	0.002^1	0.30^2	205.84^3	0.012^1	0.30^2	398.25^3	0.017^1	0.62^4	562.20^3	0.027$^/$	54.61^2
PIP-NMF	0.15^1	26.92^2	0.001^1	0.20^1	73.06^2	0.007^1	0.23^2	108.75^2	0.013^1	0.34^2	228.35^2	0.018$^/$	32.25^1
PIP-BPRMF	0.15^1	41.60^2	0.002^1	0.26^2	133.85^2	0.013^1	0.36^2	187.76^2	0.016^1	0.65^4	366.67^2	0.021$^/$	37.58^1
PIP-TSM	0.31^2	72.30^2	0.002^1	**0.42**^{3}	226.76^3	**0.020**^{1}	0.45^3	413.17^3	0.017^1	0.73^5	573.05^3	0.035$^/$	86.14^3
PER	**0.34**^{2}	**96.35**^{3}	**0.003**^{1}	0.38^3	**335.75**^{4}	0.018^1	**0.46**^{3}	**560.28**^{4}	**0.026**^{1}	**0.77**^{5}	**710.58**^{4}	**0.052**^{2}	**93.31**^{4}

We also see that the PIP-boosted algorithms are generally better than their corresponding non-PIP version, which is not surprising because we re-optimize their outputs to better align with the personalized utilities of users for Pareto efficiency. But what is surprising is that they did not significantly outperform the randomly initialized PER approach.

More detailed examinations show that the recommendation lists of NMF and BPRMF lead to heavily unbalanced service allocations among different users after processing by Algorithm 2, that those active users tend to possess much higher surplus than inactive users. This is because the inactive users are usually recommended with the similar (mostly popular) items, so that they have to compete for the limited benefits from these items. As a result, the optimization procedure of Algorithm 1 will finally converge into unbalanced Pareto efficient solutions. But because TSM has already taken user balance into consideration, its PIP-boosted version achieves comparable performance with PER.

6.4 Economic Efficiency

We evaluate the economic efficiency of the proposed algorithm in this section. To do so, we calculate the accumulated consumer surplus of a top-N recommendation list and average across the users, which is defined as follows:

$$CS@N = \frac{1}{M}\sum_{i=1}^{M}\sum_{j\in\Pi_{i,N}}\left(a_{ij}\ln(1+Q_{ij}) - P_j Q_{ij}\right) \qquad (22)$$

where i and M are the index and the total number of testing users, N is the length of recommendation list, and $\Pi_{i,N}$ is the recommendation list for the i-th user. For fair comparison between algorithms, we take $Q_{ij} = 1$ for the recommended items, which means that a user will finally purchase one for each item in the recommendation list. Figure 2 shows the results on averaged consumer surplus v.s. the length of recommendation lists, and detailed results on selected recommendation lengths are displayed in Table 2.

We see that the average consumer surplus is significantly boosted by the Pareto Improvement Process (PIP) on traditional recommendation algorithms of NMF and BPRMF, which validates the effectiveness of our PIP approach on promoting the economic efficiency of traditional algorithms. However, we also observe that PIP does not achieve significant improvement on TSM. The underlying reason can be that TSM is also designed with a philosophy of surplus maximization. Although it attempts to maximize the joint surplus of both consumers and producers, it has indeed taken the benefit of users into consideration. Nevertheless, by maximizing the joint consumer surplus in a direct manner, the PER approach gains the best performance on economic efficiency in terms of accumulated consumer surplus.

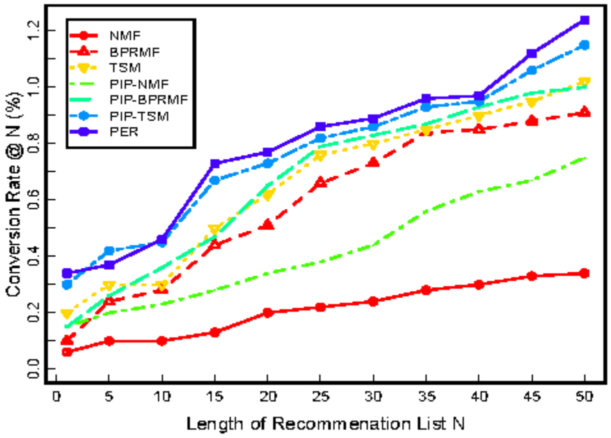

Figure 1: Comparison of different algorithms on recommendation performance. The x-axis is the length of recommendation list that we provide, and y-axis is the conversion rate at the corresponding length.

Besides, we see that the CS@N of NMF and BPRMF increases approximately linearly with the increase of recommendation length N, while in the surplus-driven approaches (e.g., PER, TSM, PIP-TSM), CS@N tends to increase with diminishing marginal increments. This shows that the surplus-driven approaches are able to rank those items of higher surplus for users to top positions in the list, while traditional algorithms tend to distribute the items according to the probability of user likeness, with little consideration of the amount of benefits that an item brings to a user.

6.5 Pareto Efficiency

We also care about the Pareto efficiency of the recommendation/allocation results, which measures the degree that a Web economic system is close to its best possible efficient resource allocation status. To do so, we calculate the averaged Percentage Utopia Surplus (PUS) when a top-N recommendation list is provided to each user:

$$PUS@N = \frac{1}{M}\sum_{i=1}^{M}\left(S_i(Q_N)/S_i^{\circ}\right) \tag{23}$$

where Q_N is the allocation matrix corresponding to the top-N recommendation lists of M testing users – that each recommended item is allocated with a quantity of one to its corresponding user. We also calculate the PUS value given the directly estimated allocation matrix Q for those applicable algorithms (i.e., PER, PIP-X, and TSM), and this result is shown as PUS_Q in the last column of Table 2.

We see that the PER approach generally gains the best PUS@N performance on most selections of recommendation length N, but the results are not statistically significant except $N = 20$ and against NMF or BPRMF, because the consumer surplus is too small when only a few (i.e., N) items are provided. However, results become clearer when we do not limit the number of provided items, and evaluate PUS on the output allocation matrix Q directly, as shown in the last column of Table 2. Note that the NMF and BPRMF algorithms do not produce any allocation matrix directly, so the PUS_Q measure is not applicable to them.

It is seen that PIP-NMF/BPRMF did not achieve satisfactory PUS even after promoting by PIP. Detailed examinations indicate that this is also due to the unbalanced initial allocations given by NMF/BPRMF, so that some users possess very high initial surplus. Per the nature of Pareto improvement that avoids hurting anyone, those users dominate the allocations so that other users eventually gain only small surplus values, leading to low average PUS results. In contrast, by treating consumer surplus in a balanced manner, TSM gives better initial surpluses and the PUS can be further promoted by the PIP framework. Finally, by optimizing the consumer surpluses toward utopia points directly, the PER approach gains significantly the best average PUS – which when combined with the observed improvement from TSM to PIP-TSM – verifies the Pareto efficiency of our proposed optimization algorithms.

7 RELATED WORK

The efficiency of economic systems has long been a vitally important research consideration in economics [4], especially in the research and practice of welfare economics [17]. Ever since the existence of human social production in the ancient age, people have been considering the proper, acceptable, and mutually beneficial ways for allocation of resources [43]. The breakthroughs of human production capability brought about by the first and second Industrial Revolution has further driven the human society to rethink about the ways to efficiently allocate goods, capital, energy, and various other types of resources [40].

Although philosophically debated among economists for centuries [31], it was not until the 1900's that economic efficiency were proposed to be clearly and objectively measured – which is called the Pareto criterion after Italian economist Vilfredo Pareto (1848-1923) [33], stating that an efficient status should be a situation in which it is impossible to make anyone better-off without making someone worse-off. Pareto efficiency (or Pareto optimality) has been one of the most basic concepts in mainstream economics, and it is widely used to analyze and optimize the resource allocation in economic and engineering systems [29].

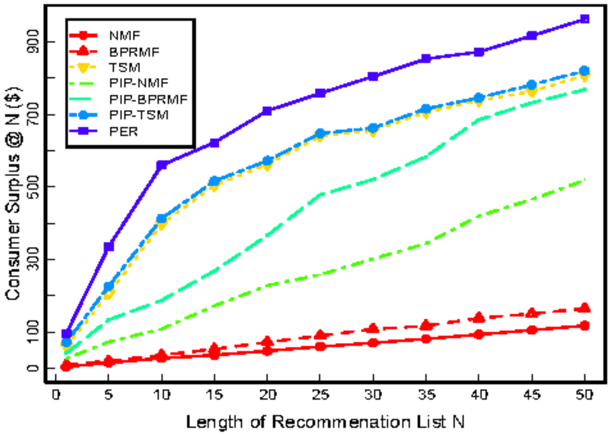

Figure 2: Comparison of different algorithms on economic efficiency. The x-axis is the length of recommendation list that we provide, and y-axis is the average consumer surplus at the corresponding length.

However, although the Web has shaped into a tremendous online economy by continuously integrating human activities from offline to online, the research community has taken seldom care of the Pareto efficiency of the Web economy. Actually, recent work [53] has shown that many of the Web-based services can be formalized into the classical producer-consumer economic paradigm, where consumers/users consume normal goods, financial products, jobs, or news feeds from producers in E-commerce [21], online financing [9, 27], crowd-sourcing [16, 18], or news portals [25], etc. By implicitly assigning certain items to targeted users in technical forms of personalized recommendation [19, 38] or search [7, 44], the Web actually serves a fundamental role in resource allocation on the massive online economy.

Nevertheless, traditional Web-based approaches mostly make allocations to match the preference of each individual user, without treating the system as a whole to examine the mutual influence between user benefits. For example, both the Collaborative Filtering (CF)-based [12, 20, 42], Content-based [26, 34], or hybrid [8] recommendation approaches adopt meticulously designed algorithms to model user preferences, but the recommendation of an item to one user does not pose effect to others. This lack of economic system-wise view leads to the Pareto inefficiency of recommendation/allocation results.

Recent research lines have begun to tackle the problem. [53] proposed economic recommendation by automatically estimating the utility and surplus for each user-item pair from large-scale Web data, such as consumer purchasing logs and online transactions. Based on this, [54] further extended the work to multiple product (i.e., product set) utility estimation, so that we can calculate the utility of a combination of products by considering their complementary and substitutive relationships. Recently, [10] proposed to investigate recommendation under capacity constraints, so that different items are mutually related to each other because each single item can not be recommended in an unlimited manner, and in this way, the Web economy is modeled as a whole system instead of a set of isolated items.

In this work, we propose to boost the *Pareto Efficiency* of Web allocations based on principled and widely-accepted economic concepts and theories, which, to the best of our knowledge, is the first time in the context of web-based recommendation systems.

8 CONCLUSIONS AND FUTURE WORK

Existing approaches of recommender systems focus on providing targeted recommendations to match the preference of each individual user. Although many Web applications have turned into intact online economic systems, little consideration is put on the essential problem of online economic efficiency – that why and how Web-based systems can achieve efficient service allocations.

This paper attempts to answer these principled questions based on established economic theories melt with solid data-driven algorithmic approaches. To do so, we first formalized the *online service allocation* problem that serves as the basic functionality of most Web-based systems, and further proposed algorithms to measure the user benefits in terms of *utility* and *surplus*. Based on this, we proposed *Pareto efficient recommendation* that attempts to find the Pareto optimal service allocation of the system with consideration of mutual correlations among users. To benefit the many existing personalized recommendation algorithms, we further proposed the *Pareto improvement process* to re-optimize their recommendation results into Pareto efficient ones. Experimental results on real-world industry data verified the proposed approaches in terms of recommendation performance, consumer surplus, and Pareto efficiency.

This is our first step towards an economic efficient Web, and there is much room for further improvements. In the future, we will study different utility functions in measuring consumer surplus. We can compute other Pareto efficient solutions beyond the balanced one to get the Pareto frontier and investigate their differences and relations. More importantly, our basic philosophy to promote Web efficiency is not restricted to e-commerce, and it can well be generalized to various other Web services, which is promising to both interdisciplinary research and practical applications.

ACKNOWLEDGEMENT

We thank the reviewers for the careful reviews and constructive suggestions. This work was sponsored by NSF grant CCF-1101741 and IIS-0953908, part of the work is sponsored by the Center for Intelligent Information Retrieval. Any opinions, findings, conclusions, or recommendations expressed in this paper are the authors', and do not necessarily reflect those of the sponsors.

REFERENCES

[1] Gediminas Adomavicius and YoungOk Kwon. 2015. Multi-criteria recommender systems. In *Recommender systems handbook*. Springer, 847–880.
[2] Deepak Agarwal, Bee-Chung Chen, Pradheep Elango, and Xuanhui Wang. 2012. Personalized click shaping through lagrangian duality for online recommendation. In *SIGIR*. ACM, 485–494.
[3] Galen Andrew and Jianfeng Gao. 2007. Scalable training of L 1-regularized log-linear models. In *ICML*. ACM, 33–40.
[4] Kenneth Arrow. 1962. Economic welfare and the allocation of resources for invention. In *The rate and direction of inventive activity: Economic and social factors*. Princeton University Press, 609–626.
[5] Paul A. Baran. 1953. Economic Progress and Economic Surplus. *Science & Society* (1953), 289–317.
[6] Paul A. Baran. 1966. Monopoly Capital. *NYU Press* (1966).
[7] Sergey Brin and Lawrence Page. 2012. Reprint of: The Anatomy of a Large-Scale Hypertextual Web Search Engine. *Computer Networks* 56, 18 (2012), 3825–3833.
[8] Robin Burke. 2007. Hybrid web recommender systems. In *The adaptive web*. Springer, 377–408.
[9] Simla Ceyhan, Xiaolin Shi, and Jure Leskovec. 2011. Dynamics of Bidding in a P2P Lending Service: Effects of Herding and Predicting Loan Success. *WWW* (2011), 547–556.
[10] Konstantina Christakopoulou, Jaya Kawale, and Arindam Banerjee. 2017. Recommendation under Capacity Constraints. *arXiv preprint arXiv:1701.05228* (2017).
[11] James S. Coleman and Thomas J. Fararo. 1992. Rational Choice Theory. *Nueva York: Sage* (1992).
[12] Michael D. Ekstrand, John T. Riedl, and Joseph A. Konstan. 2011. Collaborative Filtering Recommender Systems. *Foundations and Trends in Human-Computer Interaction* 4, 2 (2011), 81–173.
[13] Walter Eucken. 2012. The foundations of economics: History and theory in the analysis of economic reality. *Springer Science & Business Media* (2012).
[14] Daniel Friedman and Skovics Jzsef. 2013. Tractable Consumer Choice. *Theory and Decision* (2013), 1–26.
[15] Joshua Gans, Stephen King, and N. Gregory Mankiw. 2011. Principles of Microeconomics. *Cengage Learning* (2011).
[16] Jeff Howe. 2006. The rise of crowdsourcing. *Wired magazine* 14, 6 (2006), 1–4.
[17] Richard Just, Darell L Hueth, and Andrew Schmitz. 2008. *Applied welfare economics*. Edward Elgar Publishing.
[18] Aniket Kittur, Ed H Chi, and Bongwon Suh. 2008. Crowdsourcing user studies with Mechanical Turk. In *SIGCHI*. ACM, 453–456.
[19] Joseph A. Konstan and John Riedl. 2012. Recommender Systems: from Algorithms to User Experience. *UMUAI* 22, 1-2 (2012), 101–123.
[20] Y. Koren, R. Bell, and C. Volinsky. 2009. Matrix Factorization Techniques for Recommender Systems. *Computer* (2009).
[21] Kenneth C Laudon, Carol Guercio Traver, and Alfonso Vidal Romero Elizondo. 2007. *E-commerce*. Pearson/Addison Wesley.
[22] D. D. Lee and H. S. Seung. 2001. Algorithms for Non-negative Matrix Factorization. *NIPS* (2001), 556–562.
[23] Asher Levi, Osnat Mokryn, Christophe Diot, and Nina Taft. 2012. Finding a Needle in a Haystack of Reviews: Cold Start Context-Based Recommender System. *RecSys* (2012), 115–122.
[24] Blerina Lika, Kostas Kolomvatsos, and Stathes Hadjiefthymiades. 2014. Facing the Cold Start Problem in Recommender Systems. *Expert Systems with Applications* 41, 4 (2014), 2065–2073.
[25] Jiahui Liu, Peter Dolan, and Elin Rønby Pedersen. 2010. Personalized news recommendation based on click behavior. In *IUI*. ACM, 31–40.
[26] Pasquale Lops, Marco De Gemmis, and Giovanni Semeraro. 2011. Content-based recommender systems: State of the art and trends. In *Springer*. 73–105.
[27] Chunyu Luo, Hui Xiong, Wenjun Zhou, Yanhong Guo, and Guishi Deng. 2011. Enhancing Investment Decisions in P2P Lending :An Investor Composition Perspective. *KDD* (2011), 292–300.
[28] Nikos Manouselis and Constantina Costopoulou. 2007. Analysis and classification of multi-criteria recommender systems. *World Wide Web* 10, 4 (2007), 415–441.
[29] R Timothy Marler and Jasbir S Arora. 2004. Survey of multi-objective optimization methods for engineering. *Structural and multidisciplinary optimization* 26, 6 (2004), 369–395.
[30] Kaisa Miettinen and Marko M Mäkelä. 2002. On scalarizing functions in multiobjective optimization. *OR spectrum* 24, 2 (2002), 193–213.
[31] Luis R Murillo-Zamorano. 2004. Economic efficiency and frontier techniques. *Journal of Economic surveys* 18, 1 (2004), 33–77.
[32] Fernando Ortega, José-Luis SáNchez, JesúS Bobadilla, and Abraham GutiéRrez. 2013. Improving collaborative filtering-based recommender systems results using Pareto dominance. *Information Sciences* 239 (2013), 50–61.
[33] Vilfredo Pareto. 1971. Manual of Political Economy (Pareto 1927). *Macmillan* (1971).
[34] M. J. Pazzani and D. Billsus. 2007. Content-Based Recommendation Systems. *The Adaptive Web LNCS* (2007), 325–341.
[35] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt Thieme. 2009. BPR: Bayesian Personalized Ranking from Implicit Feedback. *UAI* (2009).
[36] Marco Tulio Ribeiro, Anisio Lacerda, Adriano Veloso, and Nivio Ziviani. 2012. Pareto-efficient hybridization for multi-objective recommender systems. In *Proceedings of the sixth ACM conference on Recommender systems*. ACM, 19–26.
[37] Marco Tulio Ribeiro, Nivio Ziviani, Edleno Silva De Moura, Itamar Hata, Anisio Lacerda, and Adriano Veloso. 2015. Multiobjective pareto-efficient approaches for recommender systems. *TIST* 5, 4 (2015), 53.
[38] Francesco Ricci, Lior Rokach, and Bracha Shapira. 2011. Introduction to Recommender Systems Handbook. *Springer US* (2011).
[39] Manuel C. Rios, Campbell R McConnell, and Stanley L Brue. 2013. *Economics: Principles, problems, and policies*. McGraw-Hill.
[40] George Ritzer and Nathan Jurgenson. 2010. Production, Consumption, Prosumption: The Nature of Capitalism in the Age of the Digital Prosumer. *Journal of Consumer Culture* 10, 1 (2010), 13–36.

[41] Mario Rodriguez, Christian Posse, and Ethan Zhang. 2012. Multiple objective optimization in recommender systems. In *RecSys*. ACM, 11–18.

[42] B. Sarwar, G. Karypis, and others. 2001. Item-Based Collaborative Filtering Recommendation Algorithms. *WWW* (2001), 285–295.

[43] Michael E Smith. 2004. The archaeology of ancient state economies. *Annual Review of Anthropology* (2004), 73–102.

[44] Mirco Speretta and Susan Gauch. 2005. Personalized search based on user search histories. In *WI*. IEEE, 622–628.

[45] Wolfram Stadler. 1988. Fundamentals of multicriteria optimization. In *Multicriteria Optimization in Engineering and in the Sciences*. Springer, 1–25.

[46] Sen Su, Jian Li, Qingjia Huang, Xiao Huang, Kai Shuang, and Jie Wang. 2013. Cost-efficient task scheduling for executing large programs in the cloud. *Parallel Comput.* 39, 4 (2013), 177–188.

[47] Xiaoyuan Su and Taghi M. Khoshgoftaar. 2009. A Survey of Collaborative Filtering Techniques. *Advances in Artificial Intelligence* 4 (2009).

[48] Bogdan Tomoiagă, Mircea Chindriş, Andreas Sumper, Antoni Sudria-Andreu, and Roberto Villafafila-Robles. 2013. Pareto optimal reconfiguration of power distribution systems using a genetic algorithm based on NSGA-II. *Energies* 6, 3 (2013), 1439–1455.

[49] Yan Zheng Wei, Luc Moreau, and Nicholas R Jennings. 2005. A market-based approach to recommender systems. *TOIS* 23, 3 (2005), 227–266.

[50] Andrzej P Wierzbicki. 1986. A methodological approach to comparing parametric characterizations of efficient solutions. In *Large-scale modelling and interactive decision analysis*. Springer, 27–45.

[51] Haïkel Yaïche, Ravi R Mazumdar, and Catherine Rosenberg. 2000. A game theoretic framework for bandwidth allocation and pricing in broadband networks. *IEEE/ACM Transactions on Networking (TON)* 8, 5 (2000), 667–678.

[52] Yongfeng Zhang, Min Zhang, Yi Zhang, Yiqun Liu, and Shaoping Ma. 2014. Understanding the Sparsity: Augmented Matrix Factorization with Sampled Constraints on Unobservables. *CIKM* (2014).

[53] Yongfeng Zhang, Qi Zhao, Yi Zhang, Daniel Friedman, Min Zhang, Yiqun Liu, and Shaoping Ma. 2016. Economic Recommendation with Surplus Maximization. In *Proceedings of the 25th International Conference on World Wide Web*. 73–83.

[54] Qi Zhao, Yongfeng Zhang, Yi Zhang, and Daniel Friedman. 2017. Multi-Product Utility Maximization for Economic Recommendation. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*. 435–443.