where $\Lambda = [\lambda_{ij}]_{m\times n}$ is the parameter set, $\eta > 0$ is regularizer coefficient, I_{ij} is still the indicator of whether i purchased j in the training set, and q_{ij} is the actual purchasing quantity. The quantity constraints are left out because $M_j = \infty$.

In practice, we do not have to sum over infinite q's to compute an expectation over Poisson distribution, but only need to consider sufficiently many choices. In this work, we choose to sum up from $q = 0$ to 10 because $10! = 3,628,800$ is sufficiently large to diminish the residuals according to the theory of Taylor series expansion.

The minimization of both Eq.(12) and Eq.(17) can be conducted based on gradient descent. Once the distribution parameters Λ are obtained from Eq.(17), we have the expected allocation matrix $\bar{Q}$ as:

$$\bar{Q}_{ij} = \sum_{q=0}^{\infty} q \cdot \frac{\lambda_{ij}^q e^{-\lambda_{ij}}}{q!} = \sum_{q=0}^{\infty} \frac{\lambda_{ij}^q e^{-\lambda_{ij}}}{(q-1)!} = \lambda_{ij} \tag{18}$$

which we take for service allocation and product recommendation. Note that in the regularizer of Eq.(17), λ_{ij} is actually the expectation of quantity Q_{ij} according to the nature of Poisson distribution (Eq.(18)). As a result, the regularization component applies a guidance to the learning process, so that the estimated allocation quantities for those observed transections in the training dataset would be close to their actual values.

4.2 Online Peer-to-Peer Lending

In P2P lending services like Prosper, the borrowers are loan request producers, since the loan requests can be viewed as financial products. The lenders are consumers of these financial products. Here the OSA problem is how the lenders (i.e., consumers) should distribute their assets among the loan requests (i.e., determining the allocation matrix Q), so that total surplus in the system is maximized.

In a standard online lending process, the borrower (request producer) k initiates a loan request j by specifying two features: the size of the loan M_j, and its maximal interest rate r_j^{max} that she is willing to offer. Once a request is generated, the lenders (request consumers) i bid the request by providing the amount of money they would like to lend and the interest rates they ask for, which should be lower than or equal to r_j^{max}. When the total amount of money in bid exceeds the request in a given time period, the loan request then makes a deal, and the top bidders (those with the lowest interest rates) whose money amounts to the request win the bid. The highest interest rate among the winners is set as the final interest rate r_j for the loan j.

The consumer surplus for the lenders is the interest they obtain from this loan $r_j Q_{ij}$, less the opportunity cost $\hat{r}Q_{ij}$ of investing the money in other ways. For simplicity, we set $\hat{r}$ as the risk-free interest rate (e.g., to save the money in bank). As a result, we have:

$$CS_{ij}(Q_{ij}) = (r_j - \hat{r})Q_{ij} \tag{19}$$

Similarly, the producer surplus for the borrowers is the interest they would be willing to pay $r_j^{max}Q_{ij}$, less the actual interest they have to pay $r_j Q_{ij}$, namely,

$$PS_{ij}(Q_{ij}) = (r_j^{max} - r_j)Q_{ij} \tag{20}$$

Thus the total surplus is:

$$TS_{ij}(Q_{ij}) = CS_{ij}(Q_{ij}) + PS_{ij}(Q_{ij}) = (r_j^{max} - \hat{r})Q_{ij} \tag{21}$$

Because Q_{ij} represents the quantity of money that is a continuous variable, we apply a normal distribution to describe Q_{ij}, i.e., $Q_{ij} \sim \mathcal{N}(\mu_{ij}, \sigma_{ij})$.

Expected total surplus maximization in Eq.(10) for P2P lending thus becomes

$$\begin{aligned} &\underset{U,\Sigma}{\text{maximize}} \sum_i \sum_j \int \frac{(r_j^{max} - \hat{r})Q_{ij}}{\sqrt{2\pi}\sigma_{ij}} \exp\left(-\frac{(Q_{ij} - \mu_{ij})^2}{2\sigma_{ij}^2}\right) \mathrm{d}Q_{ij} \\ &s.t.\ \mathbf{1}^T \int \frac{Q}{\sqrt{2\pi}\Sigma} \exp\left(-\frac{(Q - U)^2}{2\Sigma^2}\right) \mathrm{d}Q \le M, Q_{ij} \in \mathbb{R}_+ \end{aligned} \tag{22}$$

where $U = [\mu_{ij}]_{m\times n}$ and $\Sigma = [\sigma_{ij}]_{m\times n}$ are the parameters. This boils down to:

$$\begin{aligned} &\underset{U,\Sigma}{\text{maximize}} \sum_i \sum_j \mu_{ij}(r_j^{max} - \hat{r}) \\ &s.t.\ \mathbf{1}^T U \le M,\ \mu_{ij} \in \mathbb{R}_+ \end{aligned} \tag{23}$$

which can be solved to find the optima with linear programming. Finally, we take the expected quantity under Gaussian distribution as the allocation matrix, i.e.,

$$\bar{Q}_{ij} = \mu_{ij} \tag{24}$$

This result is interesting in that, it allows us to allocate the investments in a greedy manner according to the per capita surplus $(r_j^{max} - \hat{r})$ of each loan request, which is an intuitional rule for investment in practice and easily applicable in real-world systems.

4.3 Online Freelancing Platforms

In online freelancing networks like Mturk and Upwork, the employer (job producer) k posts job j online, and the freelancers (job consumers) i apply for the jobs that they are willing to take. Because a job can only be assigned to a single freelancer and a freelancer can only decide to take a job or not rather than take part of a job, the elements Q_{ij} in allocation matrix Q can only be binary values in $\{0, 1\}$.

The employer and freelancer negotiate to decide the salary s_j for job j. After the job is accomplished, they make ratings on each other which indicates their satisfaction about the other side. We denote the rating given by freelancer i and employer k about the job j as r_{ij} and r_{kj}, respectively, which are integers in a specific rating scale.

To estimate the consumer and producer surplus experienced on a given job, we adopt the economic assumption that the percentage surplus against the price that the consumer pays or the producer obtains is proportional to the normalized ratings that they cast on each other [12, 42], i.e., a higher rating implies a higher percentage surplus.

To do so, we predict the freelancer-job ratings $\hat{r}_{ij}$ and employer-job ratings $\hat{r}_{kj}$, respectively, based on the Collaborative Filtering (CF) approach of Eq.(7) introduced in section 2.3. By the sigmoid function $h(x) = \frac{2}{1+\exp(-x)} - 1$, we further model the percentage surplus for freelancers as:

$$\frac{U_{ij}(Q_{ij}) - s_j}{s_j} = h(\hat{r}_{ij})Q_{ij} = \left(\frac{2}{1 + e^{-\hat{r}_{ij}}} - 1\right) Q_{ij} \tag{25}$$

and the percentage producer surplus as:

$$\frac{s_j - C_j(Q_{ij})}{s_j} = h(\hat{r}_{kj})Q_{ij} = \left(\frac{2}{1 + e^{-\hat{r}_{kj}}} - 1\right) Q_{ij} \tag{26}$$

where $Q_{ij} \in \{0, 1\}$ can be viewed as a binary indicator that whether or not a job is assigned, so that a surplus can be obtained for consumers and producers in Eq.(25) and (26).

As a result, the consumer, producer, and total surpluses implied in a specific job assignment i to j are:

$$\begin{aligned} CS_{ij}(Q_{ij}) &= U_{ij}(Q_{ij}) - s_j = h(\hat{r}_{ij}) s_j Q_{ij} \\ PS_{ij}(Q_{ij}) &= s_j - C_j(Q_{ij}) = h(\hat{r}_{kj}) s_j Q_{ij} \\ TS_{ij}(Q_{ij}) &= (h(\hat{r}_{ij}) + h(\hat{r}_{kj}))\, s_j Q_{ij} \end{aligned} \tag{27}$$

On considering that Q_{ij} is binary valued, we apply a Bernoulli distribution to model its probabilistic nature, i.e.:

$$p(Q_{ij} = 1) = \alpha_{ij}, P(Q_{ij} = 0) = 1 - \alpha_{ij} \tag{28}$$

where $0 \leq \alpha_{ij} \leq 1$. Let $A = [\alpha_{ij}]_{m \times n}$ be the parameter set, and let $M_j = 1$ because each individual job is by nature provided only once. The OSA problem for online freelancing services is thus specified as:

$$\begin{aligned} &\underset{A}{\text{maximize}} \sum_i \sum_j (h(\hat{r}_{ij}) + h(\hat{r}_{kj}))\, s_j \alpha_{ij} \\ &s.t.\ \mathbf{1}^T A \leq \mathbf{1},\ 0 \leq \alpha_{ij} \leq 1 \end{aligned} \tag{29}$$

Eq.(29) can be easily optimized using linear programming. Once the parameters in $A = [\alpha_{ij}]_{m \times n}$ are obtained, we assign the job j to the freelancer i of the maximum probability α_{ij} among $\alpha_{i'j}$ of all the freelancers on that job, namely:

$$\bar{Q}_{ij} = \begin{cases} 1, \text{ if } \alpha_{ij} = \max\{\alpha_{i'j}\}_{i'=1}^m \\ 0, \text{ otherwise} \end{cases} \tag{30}$$

This result is intuitive because it can also be achieved in a greedy manner by replacing α_{ij} with Q_{ij} in Eq.(29). In this way, we assign a given job j to the freelancer i who gains the highest value regarding $(h(\hat{r}_{ij}) + h(\hat{r}_{kj}))\, s_j$, which is actually a specification of the direct non-probabilistic framework in Eq.(9). Furthermore, this can be viewed as a surplus-augmented version of the traditional CF-based personalized recommendation algorithms, which will be discussed in the following together with the previous specifications.

4.4 Remarks

It is worthwhile to compare and contrast our framework with some traditional recommendation algorithms.

In the case of unlimited quantity where $M_j = \infty$, the quantity constraint $\mathbf{1}^T \int Q p(Q) \mathrm{d}Q \leq M$ in Eq.(10) can be removed and we obtain an unconstrained optimization function, just as shown in Eq.(17). In this case, the total surplus related to each consumer is independent from those of the others, and the optimal allocations for each consumer is independently isolated from each other. Take the e-commerce application for example, the allocation for a given consumer i can be obtained with the following equation:

$$\underset{\{\lambda_{ij}\}_{j=1}^n}{\text{maximize}} \sum_j \left(\sum_{q=0}^{\infty} \frac{(\hat{a}_{ij} \ln(1+q)) \lambda_{ij}^q e^{-\lambda_{ij}}}{q!} - \lambda_{ij} c_j \right) - \eta \Omega(\Theta) \tag{31}$$

This is similar to traditional Personalized Recommender System (PRS) [30] algorithms, where we consider the preferences of each targeted user and aim to provide the most relevant recommendations. The spirit of personalization has been inherently incorporated in the design of the personalized utility of Eq.(15), where $\hat{a}_{ij} = \alpha + \beta_i + \gamma_j + \vec{x}_i^T \vec{y}_j$ describes the consumer preference towards the goods in a collaborative manner based on the latent factors learned from the wisdom of the crowds, which is similar to the Collaborative Filtering approach in Section 2.3.

Similarly for online freelancing application denoted in Eq.(29), we see that for a given target job j, the employer-job rating $h(\hat{r}_{kj})$ (predicted by CF) and the hourly salary s_j would be known values. As a result, the greedy weight $(h(\hat{r}_{ij}) + h(\hat{r}_{kj}))\, s_j$ will only depend on the freelancer i. In this sense, we are actually assigning the job j to the freelancer i of the maximized $h(\hat{r}_{ij}) s_j$. This is actually a generalization of CF-based algorithms that recommend job j to the freelancer i of the maximum predicted rating $\hat{r}_{ij}$, where the only difference is that we further take the hourly salary s_j into consideration for a maximized total surplus that is measured on a basis of money.

Another interesting yet intuitive conclusion from the existence of a non-infinity solution to Eq.(31) is that, larger quantity of products that the producers sell is not necessarily preferred by the system, although we assume the quantities that producers can supply are unlimited. This results from the diminishing marginal utility experienced by consumers, and this conclusion is verified by the disadvantages observed on dumping in practical tradings.

However, when the constraint on quantity exists, the consumer surpluses are correlated with each other, so that the allocation matrix that gains a globally maximized total surplus does not necessarily imply a maximized surplus for each consumer or producer.

5. RESULTS

In this section, we take our framework to the data, and perform the traditional tasks of purchase prediction and personalized recommendation, as well as the new task of total surplus maximization. We first present a more detailed description of the e-commerce application, which we think is one of the most representative and easy-to-understand application scenarios that match the economic theories. Then we briefly sketch results on P2P lending and online freelancing applications, to illustrate the scope of our framework.

5.1 E-commerce Dataset Description

We adopt the consumer purchasing records dataset from Shop.com[1] for model evaluation, because an important information source leveraged in our framework is the quantity of product that a consumer purchased in each transaction, which is absent in many of the public datasets. In the Shop.com dataset, however, we have both the product price information and the quantity that a consumer purchased in each record.

To avoid the problem of cold-start [21, 41], and to focus on our key research target of total surplus maximization, we select those consumers and products with at least five purchasing records, which is a frequently adopted pre-processing method in previous work [21, 20, 37]. Some statistics of our dataset are summarized in Table 2.

Table 2: Statistics of the Shop.com dataset

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

We see that the dataset is extremely sparse with a density of only 0.03%, which is similar to previously seen recommen-

[1] `http://www.shop.com`

dation tasks. Furthermore, we randomly select 75% of the transactions from each consumer to construct the training set for model learning, and the rest 25% are used for testing. These amount to roughly $100k$ transactions by $34k$ consumers on $30k$ products in the testing dataset.

5.2 Parameter Selection and Estimation

The personalized KPR utility function $U_{ij}(q)$ indicated in Eq.(15) is parameterized solely by parameter a_{ij}, and the estimation of a_{ij} boils down to the inference of consumer and product biases and latent factors by optimizing Eq.(12).

In the estimates, we set the hyper-parameter λ involved in Eq.(12) and η in Eq.(17) based on cross validation, and they are primarily set as $\lambda = 0.05$ and $\eta = 5$, unless we tune these parameters to investigate their influences on model performance. Throughout the estimation procedure we set the number of latent factors (i.e., the dimensions of $\vec{x}_i$ and $\vec{y}_j$) $K = 20$ in Eq.(12), because we find that 20 factors are sufficiently enough to stable the model performance.

Once the estimated $\hat{a}_{ij}$'s are obtained with Eq.(12), we are able to evaluate the utility $U_{ij}(q)$ of an arbitrary consumer-product pair, which allows us to learn the average allocation quantities λ_{ij} in Eq.(17). Recognizing that λ_{ij} is consumer-product specific similar to a_{ij}, we once again parameterize it in a CF manner with $\lambda_{ij} = \alpha' + \beta'_i + \gamma'_j + \vec{x}'^T_i \vec{y}'_j$, and thus λ_{ij} can be estimated as a media parameter by gradient descending on $\Theta' = \{\alpha', \beta'_i, \gamma'_j, \vec{x}'_i, \vec{y}'_j\}$.

For simplicity, we set the cost $c_j = 0.5P_j$ for all the products in the dataset, where P_j is the price of a product j. The cost ratio 0.5 is an average estimation based on the surveys of 100 producers from different product categories. Besides, we find that the performance of our framework is not sensitive to different cost ratios in a reasonable range.

Please note that when the regularization term η in Eq.(17) is set sufficiently large, the effect of total surplus component will vanish and the equation turns into a mere CF problem to predict q_{ij}, which serves a baseline algorithm in our later comparative study.

The procedure ends up with the estimated values of λ_{ij} for any given consumer-product pair in our dataset. As suggested by Eq.(18), product recommendation list is thus provided to consumer i by ranking the products in descending order of λ_{ij}. For easy reference, the values of the involved hyper parameters are shown in Table 4.

Table 4: Summary of parameters. The number of latent factors K and the CF regularization coefficient λ are identified by cross validation, and are fixed throughout the reported results; η varies so as to examine its influence; c_j is the marginal cost of product j.

#Latent factors K	λ in Eq.(12)	η in Eq.(17)	c_j in Eq.(17)
20	0.05	5	$0.5P_j$

5.3 Purchase Prediction and Recommendation

We investigated the performance of our TSM framework for the task of personalized purchase prediction and recommendation. For performance comparison, we adopt the widely used CF algorithm in Eq.(6) and (7) to predict the purchasing quantities directly, which are integer values ranging from 1 to 20. For fair comparison, the hyper-parameters K and λ are set the same as those in Table 4.

Similar to our TSM framework, once the predicted quantities are obtained, we construct the top-N recommendation list for a consumer from the testing set in descending order of the quantities. We adopt the measure Conversion-Rate@N (CR@N) for performance evaluation on top-N recommendation, which is a typical metric widely adopted in real-world e-commerce systems [7].

For a given number of testing consumers and the recommendation lists of length N for each of them, CR@N is the percentage of lists that 'hit' the purchase records in testing set for the target consumer. In our exercise, N runs from 1 up to 100. For each consumer 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 in evaluation, 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.

The results for CF and our TSM framework with different choices on regularization coefficient $\eta = 0.1, 1, 5, 10$ with recommendation length $N = 5, 10, 20$ are presented in Table 3. And more complete results for N from 1 to 100 can be seen in Figure 2. The bolded improvements in Table 3 are significant at a 0.05 level.

The results show that our TSM framework outperforms CF for most choices of regularization coefficient η and recommendation length N. An interesting observation is that the performance of TSM$^\eta$ generally degrades with the increase of η on relatively long recommendation lists, all the way towards the performance given by the baseline algorithm of CF. This is actually reasonable as stated before, because when $\eta \to \infty$, TSM literally degenerates to CF and its performance will also converge to CF. This observation further emphasizes the importance of our surplus maximization component, and it suggests that maximizing with total surplus could be beneficial to the consumer experience on personalized recommendations.

Besides, the results also suggest that the choice of η should not be too small either, which would dismiss the quantity guidance of the observed purchases, especially for top precisions in shorter recommendation lists. One possible reason can be that without the quantity guidance, λ_{ij} would mostly depend on the personalized KPR utility U_{ij}. As KPR util-

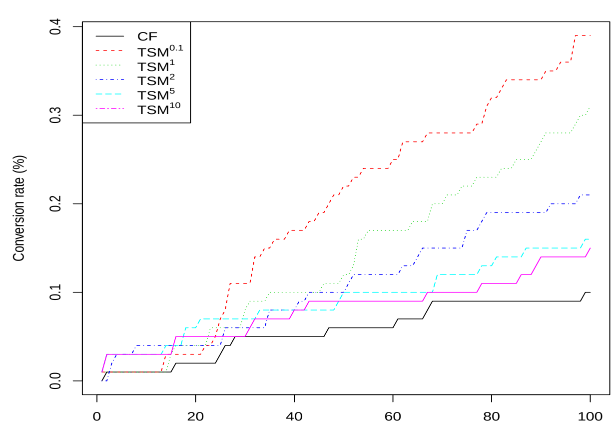

Figure 2: Comparison of the recommendation performance for CF and TSM$^\eta$. The y-axis is the conversion rate, and the x-axis is the length N of each recommendation list.

Table 3: Evaluation on Conversion Rate (CR@N) and Total Surplus (TS@N) for Top-N recommendation, where TSM^* stands for our TSM approach with regularization coefficient $\eta = *$ in Eq.(17).

N	5					10					20				
Method	CF	$TSM^{0.1}$	TSM^1	TSM^5	TSM^{10}	CF	$TSM^{0.1}$	TSM^1	TSM^5	TSM^{10}	CF	$TSM^{0.1}$	TSM^1	TSM^5	TSM^{10}
CR (%)	0.10	0.10	0.10	**0.30**	**0.30**	0.10	0.10	0.10	**0.30**	**0.30**	0.20	0.30	0.40	**0.60**	0.50
TS ($)	33.05	**1009.45**	**1009.45**	422.01	24.48	57.89	**2278.36**	2208.50	807.56	213.45	98.09	**2892.03**	3135.35	1137.89	676.65

ity function is rather limited in terms of shape flexibility, it could fail to describe the actual consumer utility for some products. We actually confirmed this by predicting the purchase quantities using the constraints in Eq.(11) directly, and the predictions turned out rather inaccurate with larger *root mean squared error* (RMSE) than that by CF. In summary, η influences the performance by balancing the importance between total surplus and quantity guidance, and it should be properly selected in practical applications.

5.4 Evaluation on Total Surplus

In this section, we closely examine the performance of our framework under the *total surplus* metric, which is a core notion of this work. The evaluation is carried out based on the recommendation results from the above section. Similar to the Top-N conversion rate, we are interested in calculating the *accumulated tocial surplus* of a Top-N recommendation list for each user, which is defined as,

$$TS@N = \frac{1}{M}\sum_{i=1}^{M}\sum_{j\in\Pi_{i,N}}\left(\hat{a}_{ij}\ln(1+\lambda_{ij}) - c_j\lambda_{ij}\right) \qquad (32)$$

where i and M are the index and the total number of testing consumers, respectively, N is the length of recommendation list, and $\Pi_{i,N}$ is the length-N personalized recommendation list for the i-th consumer.

Similarly, the resulted of $TS@N$ are reported in Table 3, and a full scope report under comprehensive choices of N can be found in Figure 3.

It can be seen from the results that our TSM approach consistently outperforms the CF method. This result is actually not surprising because our TSM framework is by nature able to maximize the total surplus by Eq.(17). Besides, we find that the smaller η is, the more total surplus our TSM approach gains. This observation on the influence of η further verifies the effects of the surplus maximization component and the quantity guidance in Eq.(17).

More interestingly, when combining this result with that on recommendation in the previous section, we find that our TSM framework can achieve decent results in terms of both total surplus and conversion rate when η is properly set. This is exciting because our framework is able to benefit the social good on total surplus, and at the same time improves the consumer experience in personalized recommendations.

5.5 P2P Lending Networks

To investigate the performance on Peer-to-Peer loan networks, we use the datasets from a famous P2P lending website Prosper[2][6]. Beginning from the third quarter in 2009, Prosper introduced an automatic bidding mechanism that bids the listings (i.e., loan requests) on behalf of the lenders automatically once a listing is created. However, as we intend to investigate the behaviour of consumers and producers in an economic system, we prefer the decisions made

[2]http://www.prosper.com

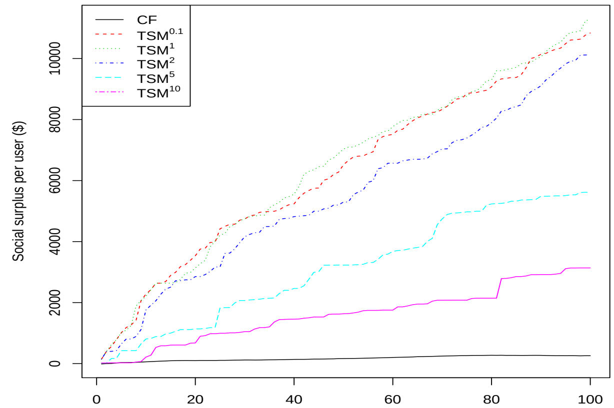

Figure 3: Comparison on Tocial Surplus (TS) for CF and TSM$^\eta$. Note that TS@N means the total social surplus of the top N recommendations when they are all accepted by the corresponding consumer.

directly by themselves, instead of those indirectly by the algorithms. As a result, we adopt those listing and bidding records before this mechanism was launched, which finally covers the period from November 9th 2005 to May 8th 2009.

As we do not consider risk control in our current model, we select those successfully funded listings whose status are not *Defaulted*, *Cancelled* or *Charge-off* from the dataset, because these listings are meant to be ruled out from the system by the intelligent risk control mechanisms. Finally, our dataset involves those funded listings of the status *Current*, *Late*, *Payoff in Progress*, or *Paid*, which correspond to 46,680 listings, 1,814,503 bids, and a total amount of $157,845,684 fundings. Some statistics of these records are summarized in Table 5.

Table 5: Statistics of the selected Prosper dataset, where 'rate' represents the interest rate of a loan.

#Listings	#Lenders	#Bids	TotalAmount
46,680	49,631	1,814,503	$157,845,684
MinimumRate	**MaximumRate**	**AverageRate**	**Amount/Listing**
0.0001	0.4975	0.1662	$3,381.44

To calculate the total surplus reached by an arbitrary allocation $Q = [Q_{ij}]_{m\times n}$, we take the yearly average bank deposit interest rate $\hat{r} = 0.01$ as the risk-free interest rate, and the TS for P2P loaning can be calculated as:

$$TS_{P2P} = \sum_i \sum_j Q_{ij}(r_j^{max} - \hat{r}) \qquad (33)$$

Based on this, the results on total surplus for the actual allocations (Actual) and our Total Surplus Maximization (TSM) framework are shown as follows:

Table 6: Results on total surplus with and without our Total Surplus Maximization (TSM) framework.

	TS($)	TS/Listing($)	TS/capita($)
Actual	25,174,131	539.29	0.1595
TSM	33,838,364	724.90	0.2144

The estimates indicate that the TSM framework achieves 34.42% higher total and per listing/capita surplus, from \$0.16 per capita to \$0.21 per capita, which is a major improvement in efficiency for the online lending systems. Based on two-tailed t-test on the large amount of listings, the improvements are significant at a 0.01 level.

The improvement on total surplus is not surprising because our framework intends to achieve a maximized surplus among all the possible allocations. However, we should further verify that our allocations are acceptable to the lenders in practice. As a result, we calculate the Percentage of Paid (PoP) listings among all the funded listings in our dataset, which indicates the safety factor of a funding allocation.

Results show that the PoP among all the listings in our selected dataset is 69.37%, while the PoP among the funded listings of our TSM allocation is 73.32%, which is no lower than the actual PoP. This suggests that our TSM framework is able to improve efficiency without impairing the safety of the system.

5.6 Online Freelancing

We used the dataset from Zhubajie[3](ZBJ) for empirical verification of online freelancing applications. ZBJ is a famous Chinese online marketplace website that includes online jobs across various categories. Each employment record includes the employer, freelancer, and job IDs, the hourly salary, as well as the employer-job and freelancer-job ratings, which are integers ranging from 0 to 5. Some of the basic statistics of the dataset that we collected are summarized in Table 7.

Table 7: Some key statistics of the ZBJ dataset.

#Employers	#Freelancers	#Jobs	AverageSalary
40,228	46,856	296,453	¥21.68/hr

#Employer Ratings	#Freelancer Ratings	Average Employer Rating	Average Freelancer Rating
276,103	241,638	2.336	2.405

Similar to our e-commerce application, we make job recommendations to freelancers based on the allocation matrix produced by our framework, then verify the performance on this task. To do so, we take all the freelancer-job ratings, and conduct personalized recommendation based on Collaborative Filtering (CF). In CF, a job j is assigned to freelancer i who has the highest predicted rating $\hat{r}_{ij}$, while in our Total Surplus Maximization (TSM) framework, it is assigned to the freelancer where $Q_{ij} = 1$ according to Eq.(30).

We conduct five-fold cross-validation for both methods, and we still adopt the Conversion Rate (CR) for performance evaluation, which is the percentage of properly assigned jobs in the testing dataset. Results of TSM and CF methods are presented in Table 8, under different choices of the number of latent factors K used for rating prediction (see Eq.(6)).

Table 8: Conversion rate on job recommendation.

K	5	10	20	30	40	50
CF(%)	0.165	0.216	0.244	0.258	0.262	0.266
TSM(%)	0.384	0.421	0.453	0.486	0.507	0.512

Results show that our TSM framework gains consistently better performance on conversion rate for job recommendation. The improvements are significant at 0.01 level for all choices of latent factors K. According to the discussions in Section 4.4, the improvement comes from the inherent consideration of salary rate in our model, which implies that the salary could be an extremely important factor when freelancers seek for jobs. Besides, we see that the results tend to be stable when $K \geq 40$ for both methods, which means that a dimensionality of 40 could be sufficiently enough to describe the factors considered by freelancers.

We further calculate the total surplus for the allocations given by CF and TSM under different choices of K's. Once an arbitrary allocation $Q = [Q_{ij}]_{m\times n}$ is realized in practice, we obtain the total surplus as:

$$TS_{Fr} = \sum_i \sum_j \left(h(\hat{r}_{ij}) + h(\hat{r}_{kj})\right) s_j Q_{ij} \tag{34}$$

We calculate the total surplus for each of the five testing folds, where there are 59,291 job allocations on average in each fold. Finally, the averaged total surplus among the five folds are shown in Table 9, where the surplus is measured in CNY (¥) and 'm' is for 'million'.

Table 9: Total surplus of online freelancing job allocations under typical choices of latent factor K.

K	5	10	20	30	ActualAllocation
CF(¥)	1.562m	1.758m	1.824m	1.860m	2,593,618
TSM(¥)	3.235m	3.862m	4.270m	4.336m	

The improvements on total surplus are significant at 0.001 level for all choices of K. We see that our TSM framework consistently gains more surplus than CF. It even leads to more surplus than the actual surplus of the testing dataset. The TSM framework gains a total surplus of ¥73.13/job on a job-level when $K = 30$, while that for the CF approach and the actual allocation are ¥31.37/job and ¥43.74/job, respectively.

The fact that the total surplus of the actual allocation is less than that gained by our TSM approach implies the failure of market equilibrium, which is frequently observed by economists in the research of antitrust and market regulations. For online freelancing as an example, this comes from the problem of information asymmetry between freelancers and employers, because it could be impossible for the freelancers to browse millions of jobs to make a final decision. This further stresses the importance of personalized recommendation techniques in service allocation, which help to push the appropriate jobs to freelancers, so as to overcome the problem of information overload.

When putting the evaluation results on total surplus and recommendation together, we find it extremely exciting because our TSM framework leads to better market efficiency even than the practical market of the system, while at the same time it benefits the freelancers with more acceptable job recommendations. This means that our allocation solution may well be applied in practice for a better off in online markets compared with current actually adopted recommendation techniques.

6. RELATED WORK

In mainstream economics, economic surplus [13, 9, 2], also known as total welfare or Marshallian surplus [25] (named after Alfred Marshall), refers to three closely related quantities: consumer surplus, producer surplus, and social/total surplus, where social surplus is the sum of surpluses experienced by both consumers and producers. The research

[3] `http://www.zbj.com`

of surplus has had quite a long history in the progress of economical theories, dating back to as early as the 19th century with the initial understandings of Surplus Values [26, 27], when the gigantic increase in wealth and population brought by the First and Second Industrial Revolution drove economists to investigate the nature of economical increase [33].

In modern economics, the concept of social surplus has been widely adopted by economists for economic system analysis and mechanism design, usually as a direct measure of *social good* to benefit the good of our human society [13, 28, 3]. However, although the Web has formed itself as a virtual society by continuously integrating the human activities from offline to online, the research community still has seldomly investigated the surplus nature of the Web as a social system.

Actually, a large number of Web-based services can be formalized as consumer-producer interaction systems, including the most commonly used E-commerce websites [23, 35], online financing [22, 6], crowd-sourcing systems [10, 5], and even social networks [32, 15], where the consumers consume normal goods, financial products, freelancing jobs, or information from the corresponding producers therein.

These applications raise the practical problem of matching services from producers to consumers. Perhaps the most closely related tasks for such matching processes are Personalized Recommendation [11, 30, 16] and Search [4, 24, 14], which feed the implicit or explicit needs of the users with recommendations and search results.

However, current approaches for such tasks mainly focus on the benefits of one side without explicitly modeling the benefits of the Web system as a whole. For example, the widely adopted Collaborative Filtering (CF) [11, 17, 38, 36, 29, 34, 40] techniques for personalized recommendation inherently focus on the maximization of consumer satisfaction based on their preferences. Although the satisfaction of consumers intuitionally benefits the surplus of producers by improving the potential of user clicks, there is no direct guarantee that such a single-side oriented modeling can benefit both sides.

In this work, however, we view the Web as a virtual society and propose to maximize the social surplus directly, based on well-developed and widely-accepted economic concepts and conclusions, which, to the best of our knowledge, is the first time to do so in the context of web-based applications.

7. CONCLUSIONS AND FUTURE WORK

Most existing literature on recommender systems focuses on developing new algorithms for standard evaluation metric such as RMSE, conversion rate or click through rate. There is little research on some fundamental questions, such as what metrics should be used to evaluate recommender systems and to what extent do the metrics reflect the goals of users, producers, platform providers, and the overall Web economy.

This paper is our first step towards finding principled answers to these questions based on established economic theory. Considering a recommender system as an information agent to support two-sided matching tasks, we introduce established economic surplus theory into recommender systems and meld it with recent data driven algorithmic approaches. Our proposed Total Surplus Maximization framework integrates the goals of users and suppliers, which can be a good metric to optimize for platform providers as it better reflects the overall economic value of the online system. We have illustrated how to realize this framework for different recommendation systems. The empirical results for several sets of industry data demonstrated the effectiveness of the proposed framework.

This paper focuses on the broadest metric about efficiency, or maximization of total surplus, and this inherent principle is not restricted to recommendation tasks that we primarily investigated, but applicable to the whole research effort of web intelligence for social good. In the future, we will also examine performance metrics about its two major components: producer surplus and consumer surplus. We can also try the ideas on new datasets, compare different functional form and specifications of utilities and profits. We will implement it both in static (one-time) and dynamic (multi-period/session/page) recommendation or search settings, and evaluate with real users to see the short term and long term impact of the total surplus based framework.

Acknowledgement

This work was sponsored by the National Science Foundation under grant number CCF-1101741 and IIS-0953908 of the US, as well as the National Key Basic Research Program (2015CB358700) and Natural Science Foundation (61532011, 61472206) of China. Any opinions, findings, conclusions, or recommendations expressed in this paper are the authors, and do not necessarily reflect those of the sponsors.

8. REFERENCES

[1] Adam Smith. An Inquiry into the Nature and Causes of the Wealth of Nations. *Strahan and Cadell, London*, 1776.

[2] P. A. Baran. Economic Progress and Economic Surplus. *Science & Society*, pages 289–317, 1953.

[3] T. Borgers, D. Krahmer, and R. Strausz. An Introduction to the Theory of Mechanism Design. *Oxford University Press*, 2015.

[4] S. Brin and L. Page. Reprint of: The Anatomy of a Large-Scale Hypertextual Web Search Engine. *Computer Networks*, 56(18):3825–3833, 2012.

[5] M. Buhrmester, T. Kwang, and S. D. Gosling. Amazon's Mechanical Turk: a New Source of Inexpensive, yet High-quality, Data? *Perspectives on psychological science*, 6(1):3–5, 2011.

[6] S. Ceyhan, X. Shi, and J. Leskovec. Dynamics of Bidding in a P2P Lending Service: Effects of Herding and Predicting Loan Success. In *Proceedings of the 20th International Conference on World Wide Web (WWW)*, pages 547–556. ACM, 2011.

[7] D. Chaffey, F. Ellis-Chadwick, R. Mayer, and K. Johnston. Internet Marketing: Strategy, Implementation and Practice. *Pearson Education*, 2009.

[8] J. S. Coleman and T. J. Fararo. Rational Choice Theory. *Nueva York: Sage*, 1992.

[9] J. M. Currie, J. A. Murphy, and A. Schmitz. The Concept of Economic Surplus and its Use in Economic Analysis. *The Economic Journal*, pages 741–799, 1971.

[10] A. Doan, R. Ramakrishnan, and A. Y. Halevy. Crowdsourcing Systems on the World-Wide Web. *Communications of the ACM*, 54:86–96, 2011.

[11] M. D. Ekstrand, J. T. Riedl, and J. A. Konstan. Collaborative Filtering Recommender Systems. *Foundations and Trends in Human-Computer Interaction*, 4(2):81–173, 2011.

[12] D. Friedman and S. József. Tractable Consumer Choice. *Theory and Decision*, 79(2):333–358, 2015.

[13] J. Gans, S. King, and N. G. Mankiw. Principles of Microeconomics. *Cengage Learning*, 2011.

[14] M. R. Ghorab, D. Zhou, A. O'Connor, and V. Wade. Personalised Information Retrieval: Survey and Classification. *User Modeling and User-Adapted Interaction*, 23(4):381–443, 2013.

[15] A. M. Kaplan and M. Haenlein. Users of the World, Unite! The Challenges and Opportunities of Social Media. *Business Horizons*, 53(1):59–68, 2010.

[16] J. A. Konstan and J. Riedl. Recommender systems: from algorithms to user experience. *User Modeling and User-Adapted Interaction*, 22(1-2):101–123, 2012.

[17] Y. Koren, R. Bell, and C. Volinsky. Matrix Factorization Techniques for Recommender Systems. *Computer*, 42(8):30–37, 2009.

[18] D. D. Lee and H. S. Seung. Learning the Parts of Objects by Non-negative Matrix Factorization. *Nature*, 401(6755):788–791, 1999.

[19] D. D. Lee and H. S. Seung. Algorithms for Non-negative Matrix Factorization. In *Advances in Neural Information Processing Systems (NIPS)*, pages 556–562, 2001.

[20] A. Levi, O. Mokryn, C. Diot, and N. Taft. Finding a Needle in a Haystack of Reviews: Cold Start Context-based Hotel Recommender System. In *Proceedings of the sixth ACM conference on Recommender systems (RecSys)*, pages 115–122. ACM, 2012.

[21] B. Lika, K. Kolomvatsos, and S. Hadjiefthymiades. Facing the Cold Start Problem in Recommender Systems. *Expert Systems with Applications*, 41(4):2065–2073, 2014.

[22] C. Luo, H. Xiong, W. Zhou, Y. Guo, and G. Deng. Enhancing Investment Decisions in P2P Lending: an Investor Composition Perspective. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 292–300. ACM, 2011.

[23] B. Mahadevan. Business Models for Internet-based E-commerce. *California Management Review*, 42(4):55–69, 2000.

[24] C. D. Manning, P. Raghavan, and H. Schütze. Introduction to Information Retrieval. *Cambridge: Cambridge university press*, 1, 2008.

[25] A. Marshall and C. W. Guillebaud. Principles of Economics. *World Scientific*, 1961.

[26] K. Marx. Wages, Prices and Value Added. *Paris: La Pleiade*, 1865.

[27] K. Marx, G. A. Bonner, and E. Burns. Theories of Surplus Value. *Capital*, Volume IV, 1863.

[28] E. S. Maskin. Mechanism Design: How to Implement Social Goals. *The American Economic Review*, pages 567–576, 2008.

[29] J. D. Rennie and N. Srebro. Fast maximum margin matrix factorization for collaborative prediction. In *Proceedings of the 22nd international conference on Machine learning*, pages 713–719. ACM, 2005.

[30] F. Ricci, L. Rokach, and B. Shapira. Introduction to Recommender Systems Handbook. *Springer US*, 2011.

[31] M. C. Rios, C. R. McConnell, and S. L. Brue. Economics: Principles, problems, and policies. *McGraw-Hill*, 2013.

[32] G. Ritzer and N. Jurgenson. Production, Consumption, Prosumption: The Nature of Capitalism in the Age of the Digital Prosumer. *Journal of Consumer Culture*, 10(1):13–36, 2010.

[33] M. Rothbard. An Austrian Perspective on the History of Economic Thought Volume II: Classical Economics. *Edward Elgar Publishing Ltd*, 1995.

[34] R. Salakhutdinov and A. Mnih. Bayesian Probabilistic Matrix Factorization using Markov Chain Monte Carlo. In *Proceedings of the 25th International Conference on Machine Learning (ICML)*, pages 880–887. ACM, 2008.

[35] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl. Analysis of Recommendation Algorithms for E-commerce. In *Proceedings of the 2nd ACM conference on Electronic Commerce (EC)*, pages 158–167. ACM, 2000.

[36] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl. Item-based Collaborative Filtering Recommendation Algorithms. In *Proceedings of the 10th International Conference on World Wide Web (WWW)*, pages 285–295. ACM, 2001.

[37] X. Su and T. M. Khoshgoftaar. A Survey of Collaborative Filtering Techniques. *Advances in Artificial Intelligence*, 2009(4), 2009.

[38] G. Takács, I. Pilászy, B. Nemeth, and D. Tikk. Investigation of Various Matrix Factorization Methods for Large Recommender Systems. In *Data Mining Workshops, 2008. ICDMW'08. IEEE International Conference on*, pages 553–562. IEEE, 2008.

[39] H. R. Varian. Microeconomic Analysis (3rd Edition). *WW Norton, New York*, 1992.

[40] Y. Zhang, G. Lai, M. Zhang, Y. Zhang, Y. Liu, and S. Ma. Explicit Factor Models for Explainable Recommendation based on Phrase-level Sentiment Analysis. In *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 83–92. ACM, 2014.

[41] Y. Zhang, M. Zhang, Y. Zhang, Y. Liu, and S. Ma. Understanding the Sparsity: Augmented Matrix Factorization with Sampled Constraints on Unobservables. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management (CIKM)*, pages 1189–1198. ACM, 2014.

[42] Q. Zhao, Y. Zhang, D. Friedman, and F. Tan. E-commerce Recommendation with Personalized Promotion. In *Proceedings of the 9th ACM Conference on Recommender Systems (RecSys)*, pages 219–226. ACM, 2015.