Peer-to-Peer Match-making in Sharing Economy - from a Timebanking Approach

Project Proposal for SYS6014 Decision Analysis Spring2020

Ian Liu

03/24/2020

Project Introduction & Description

As the concept of sharing economy has been globally proved to be efficient and economical, a relative new economic model has been invented and introduced, which is to utilize time as a tradable currency, to let participants perform specific skills on tasks in order to exchange hours of work with other participants who request specific services. In specific, a participant can be in both roles of service provider and requester, depending on the dynamic needs of theirs and others. During the entire transaction, although the physical money is replaced by working hours, both parties of the transaction still has achieved their objective of exchanging services or needs by sharing common interests, or complementarities. This new type of economic model is called time-banking.

For example, Jack will be out of town over the weekend for a conference and needs someone reliable and trustable to take care of his child. Maria, Jack's neighbor, is a professional registered nurse at a local children's hospital and will be taking a break over the weekend. However, due to the nature of Maria's work, she must be on night shifts 4 days a week, which makes her hardly find a proper time window to mow her lawn. In fact, Jack is getting a 2-day break from work when he returns back from the conference, and he can mow the lawn for Maria as a return, in exchange of Maria's time for taking care of his child over the weekend.

There actually have been a few successful platforms out there that can help the participants of timebanking set up the matches. However, things just become more complicated as the number of participants is getting larger, which means that each participant must make a decision when multiple matches have become available for them to confront. And due to the lack of sufficient information provided to the service requester, making a decision among many choices and spending lots of time to pick potential service providers could be the major causes that the concept of timebanking cannot be adopted by majority of the public.

Let's get back to Jack and Maria's case. What if there are multiple help requests that Maria could accept for the weekend? What if there are multiple babysitters available in the neighborhood that Jack gets to choose? Here comes a decision-making problem, for the system itself.

The Decision Problem

In this project, the decision-maker is the system itself. It's run by an algorithm that clusters all the users that have similar profiles, as well as approximately matched requests, preferences, and interests. Then the system will decide which two candidates can be a good match by utilizing the k-means clustering technique, so that both of the users, candidates in this case, do not have to make a choice from long lists of potential candidates.

The set of options is defined using pre-set profiles and requests, despite complexity and diversity of each participant's profile. And the system would calculate scores of similarity, complementarity, and proficiency for potential participants based on the available information provided in their profiles. The stakes of the decision vary depending on the nature of each decision. However, we wish to find the best match, or the final decision with the highest final scores, for participants to choose. On the other hand, poor decisions will definitely lead to issues such as inequality, controversy, or disputation about timing hours. Since the physical money does not get involved in the transaction, it's pretty hard for participants to intuitively evaluate the work load with working hours. Having said that, it is definitely not going to work out if Maria babysits Jack's child for the entire weekend, but Jack only mows Maria's lawn for 5 minutes. This issue will be addressed by using the score of proficiency, which is calculated by the score of proficiency.

Mathematical Formulation

The core idea of the model follows a combination of scores of similarity, complementarity, and skill proficiency between service requester $i \in I$, and service provider $j \in J$:

$$Final\ Score_{i,j} = Score\ of\ Similarity\ for\ Profiles/Tasks_{i,j} + Score\ of\ Complemantarity_{i,j} + Score\ of\ Proficiency_{i,j}$$

Two types of data will be used for further analysis and calculation, the first type is *Static*, which includes the data directly coming from participants' profiles: Age, $Gender$, $Educational\_Background$, $Race$, $Location_{Address}$, $Jobs\_Good\_At$, $Jobs\_Not\_Good\_At$, $General\_Available\_Time\_Slots$, and $Preferences\_on\_Provider$; and the other type of data is *Dynamic*, which includes the data coming from the dynamic request pool: $Job\_Location$, $Job\_Category$,Job_Time, and $Proficiency\_Level\_Rqrmt$.

The following picture provides a glance of how the static dataset looks like, and more dummy profiles are being generated. When participants are setting up their profiles, questions of the following will be asked:

- General and personal information (Age, $Gender$, $Educational\_Background$, $Race$, and $Address$).
- What type(s) of job(s) are you good at? And ranking them from 1 to 10 ("10" here represents "Professionally Good at").
- What type(s) of job(s) are you NOT good at? And ranking them from 1 to 10 ("1" here represents "Have No Idea How To Do It").
- Please state what preferences do you have for the potential candidates? (Age range, gender, race, available time windows etc.)

Name	Age	Gender	Race	City	State	Educational_ Background	Jobs_Good_At (1-10)	Jobs_N ot_Good_At (1-10)	General_Availabl e_TimeSlots	Preferences_o n_Provider
Maria	45	Female	Asian	Charlottesvi lle	VA	Master	"cooking,9","b abysitting,10", "gardening,8"	"mechnic_repairing, 1"	Monday_Night	"40-50","Female", "repairing", "Sunday"
Jack	33	Male	Black	Charlottesvi lle	VA	Bachelor	NA	"cooking, 2", "babysitting,3"	Saturday_Morni ng	NA
Mike	55	Male	White	Richmond	VA	High school	"mechnic_repa iring, 9"	NA	Sunday_All_Day	NA
Jason	19	Male	White	Fairfax	VA	Bachelor	NA	NA	NA	NA
Emily	21	Female	Asian	Charlotte	NC	Bachelor	NA	NA	NA	NA
Anna	30	Female	Black	Atlanta	GA	Bachelor	NA	NA	NA	NA

Figure 1: A Help Requester's Profile

Notations for Mathematical Model:

- $$a \in A = \{20-30, 31-40, 41-50, 51-60, 61-70, 71+\}$$
- $$g \in G = \{Male, Female, Not\ Specified\}$$

- $$r \in R = \{American\ Indian, Asian, African\ American, Hispanic, Pacific\ Islander, White\}$$
- $$l \in L = \{All\ States\ and\ Territories\ of\ the\ U.S\ by\ Counties\}$$
- $$e \in E = \{High\ School, Associate's, Bachelor's, Master's, Doctorate, Post-Doc, MD, JD, Other\}$$
- $$c \in C = \{Cleaning, Repairing, Cooking, Gardening, Groceries, Delivery,\ etc\}$$
- $$t \in T = \{24\ hours\ in\ a\ day, by\ 1\ hour\}$$
- $$d \in D = \{365\ days\ in\ a\ year,\ in\ format\ of\ "MM-DD-YYYY"\}$$

People who request services (denoted by i):

$$Prof_{i,n} \in \{Prof_{i,age}, Prof_{i,gender}, Prof_{i,location}......Prof_{i,n}\}$$

Where

$$n \in N = \{A, G, R, L, E, C, T, D\}$$

And their preferences are denoted:

$$Pref_{i,n} \in \{Pref_{i,age}, Pref_{i,gender}, Pref_{i,location}, Pref_{i,job\_not\_good\_at}......Pref_{i,n}\}$$

Where

$$n \in N = \{A, G, R, L, E, C, T, D\}$$

People who offer services (denoted by j):

$$Prof_{j,n} \in \{Prof_{j,age}, Prof_{j,gender}, Prof_{j,location}......Prof_{j,n}\}$$

Where

$$n \in N = \{A, G, R, L, E, C, T, D\}$$

And their preferences are denoted:

$$Pref_{j,n} \in \{Pref_{j,age}, Pref_{j,gender}, Pref_{j,location}, Pref_{j,job\_not\_good\_at}......Pref_{j,n}\}$$

Where

$$n \in N = \{A, G, R, L, E, C, T, D\}$$

Successful Matches

A potential match (denoted by m) will take both of static and dynamic information into consideration while matching each other over a specific job category c. For example,

$$m_{i,j} \in M_{i,j} = \{matching(i,j) | Prof_{i,n} = Prof_{j,n}, Pref_{i,n} = Pref_{j,n}, ...\}$$

where $i \neq j$, $Prof_{i,a} = Prof_{j,a}$, $Prof_{i,l} = Prof_{j,l}$, $Prof_{i,c} = Prof_{j,c}$, $Prof_{i,t} = Prof_{j,t}$, $Prof_{i,d} = Prof_{j,d}$, $Pref_{i,a} = Pref_{j,a}$, $Pref_{i,g} = Pref_{j,g}$, $Pref_{i,r} = Pref_{j,r}$, $Pref_{i,l} = Pref_{j,l}$, $Pref_{i,e} = Pref_{j,e}$, $Pref_{i,c} = Pref_{j,c}$, $Pref_{i,t} = Pref_{j,t}$, $Pref_{i,d} = Pref_{j,d}$.

Now we step into the decision-making process. After the system picks all the potential candidates for possibles matches, it could be possible that each i could have multiple j's to pick, and how should we make the optimal decision? We calculate potential candidates' scores of similarity, complmentatiry, as well as the proficiency in order to recommend the most-matched j to i.

Calculation of the Final Score

Now we introduce the idea of calculating the Final Score for evaluating the degree of matching user between i, and j.

Similarity

$$Score\ of\ Similarity\ for\ Profiles = Prof_i(h) + Prof_j(h)$$

Where

$$h \in H = \{L, C, T, D\}$$

$$Score\ of\ Similarity\ for\ Preferences = Pref_i(n) + Pref_j(n)$$

Where

$$n \in N = \{A, G, R, L, E, C, T, D\}$$

Complementarity

$$Score\ of\ Complementarity = |Prof_i(c) + Prof_j(c)|$$

For instance, $Prof_i$ states that they are not good at gardening (rated "1"); while $Prof_j$ states that their gardening skill is nearly perfect (rated "9"). As result,

$$Score\ of\ Complemantarity_{i,j} = |Prof_i(c_{gardening}) + Prof_j(c_{gardening})| = |1 - 9| = 8$$

The higher score of complementarity is, the more likely two participants are a good match.

Proficiency

$$Score\ of\ Proficiency\ _{i\ or\ j} = \frac{\Sigma_i^{N_c} w_i(c)}{\#\text{ of times prof }_i\text{ or prof }_j\text{ have accepted requests }\ (N_c)}$$

The Score of Proficiency measures how proficient a service provider i might be on a specific job $c \in C$, C here denotes the job category, or a pool. And it sums up all the previous evaluations (in numeric scores) of the jobs that i has completed, and has it divided it by the total number of times that i has been evaluated by service requesters. $w_i(c)$ denotes the evaluation score that i has received for one job.

K-means Clustering

Basic Idea about K-means

K-means is one of the most famous clustering techniques that is being used to solve problems under the environment of unsupervised learning.

In the case of making matches based on profiles' contents (static data), as well as requests (dynamic), we could define the $Final\ Score\ of\ Matching$, a composite of Score of Similarity, Score of Complementarity, and Score of Proficiency, to be a new type of metric that helps determine the distance between data points.

Using the idea of k-means to solve Peer-to-peer matching problem is non-trivial. Here, the novelty is that we consider 'matching' as an unsupervised learning problem. In fact, any unsupervised learning algorithm can be used in this case We consider matching level as a redefined metric in feature space. In addition, based on the past transactions, we can do a metric learning to learn what is the best matching metric.

And by defining and calculating the new metric, we may be able to discover a new recommender system that clusters users by their matching scores into different groups. Then the users that have the minimum distances with others in a clustering group will be recommended for further matching selection.

Final Decisions

Normalization on the *Final Score* must be performed before being used for the utility function:

$$p^* = \begin{cases} 1 & \text{if it's a good match} \\ 0.5 & \text{if it's a partial match} \\ 0 & \text{otherwise} \end{cases}$$

- The initial idea of the matching process is shown in [Figure 1].

User Data Profile Setup and Data Collection

The process of data collection will be based on analyzing the information that users put in their profiles. In general, information needed for initiating a profile match by using Natural Language Processing techniques is listed below in [Figure 2] and [Figure 3]

In terms of data collection, I am proposing to ask a group of students at UVa to fill out a survey that contains the questions in the following diagram. This is evaluated as the data source for this project. As the users' pool is getting more sufficient, the matching processing is expected to be more and more efficient.

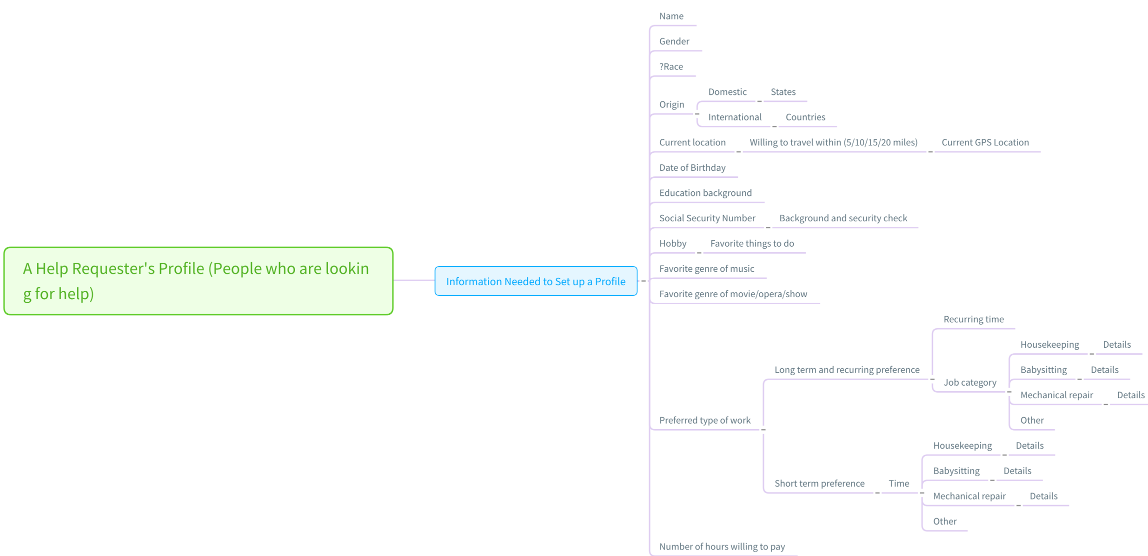

Figure 2: A Help Requester's Profile

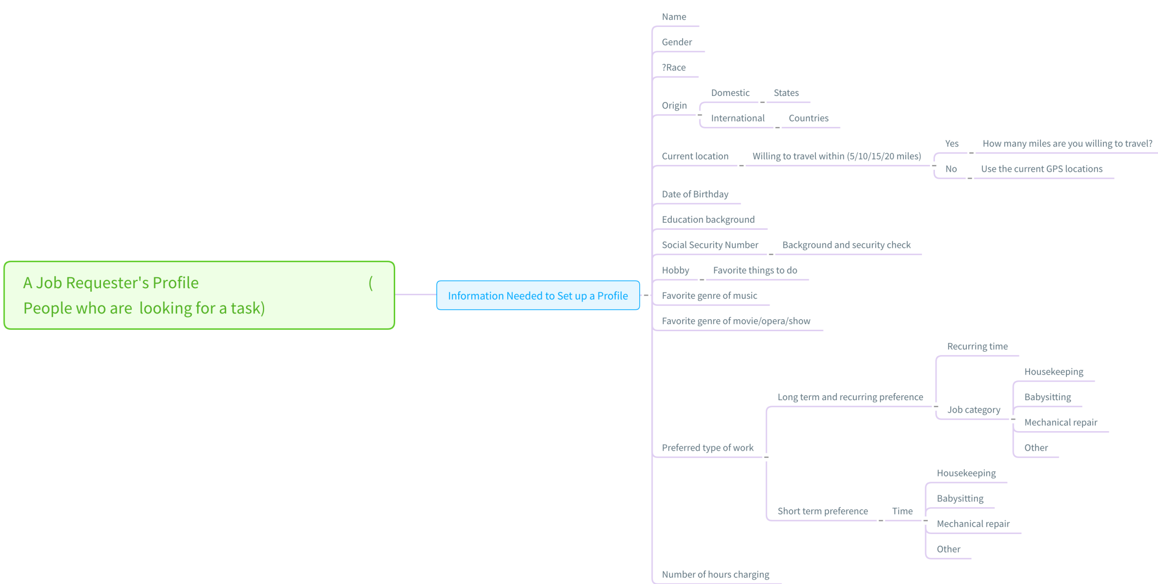

Figure 3: A Job Requester's Profile