Romantic Partner Recommender based on Speed Dating Experiment

Shih-Ying Jeng, Yijia Xu, Min Fu
Dept. of Computer Science, Dept. of Electrical Engineering
Columbia University
sj2909@columbia.edu yx2489@columbia.edu mf3200@columbia.edu

***Abstract*—Most dating recommenders are based on online dating sites that make use of virtual user profiles portrayed by the users themselves. While speed dating collects real-life face-to-face dating interactions among people, the data is rarely applied in dating recommender in any form. This report explores the research gap and presents a novel dating recommender which combines speed dating study with dating recommender system. Speed dating data suggests that user profile portrayed by themselves may not accurately reflect their likeability as a potential romantic partner. We introduce an approach of extracting objective evaluation based on the objective ratings given by the dating partners in the speed dating events to construct an objective profile library. Due to the low match rate in speed dating, there is a high class imbalance on the match label in the speed dating dataset. The project applies SMOTE oversampling to mitigate the class imbalance issue. A random forest regression-based reciprocal recommender is presented. Experiment results confirm the effectiveness of the proposed approach. The combination of the objective profile library, SMOTE oversampling, and a random forest regression-based reciprocal recommender achieves a match prediction accuracy of 92.19%, outperforming existing benchmark algorithms.**

***Index Terms*—Dating Recommender, Speed Dating, Reciprocity, SMOTE**

I. INTRODUCTION

DATING service industry has seen dramatic growth in recent years. Online dating websites along with speed dating events have become popular platforms to search for potential romantic partners, providing access to a much larger pool of potential dates that is otherwise unavailable via traditional means.

Dating recommender is most commonly seen in online dating, which is based on the virtual profile of the users. Researchers have proposed various ways to improve online dating recommender. Speed dating is used as an invaluable tool in the study of romantic attraction and dating behavior Finkel et al. [6]. However, rarely has the insights discovered in the face-to-face speed dating experiments been exploited in improving dating recommendation systems.

One important discovery from speed dating, as it turned out, is the non-negligible difference between people's belief and the reality. This difference is reflected in two aspects. The first aspect is that an individual's subjective profile portrayed by himself/herself differs from the objective profile portrayed by others who have interacted with this individual. The second aspect is that an individual's stated preference for romantic partner differs from the statistical decision factors that influences whether the individual wants a second date with a potential partner.

The discoveries based on face-to-face speed dating experiment provides invaluable insights for improving dating recommendation systems. Given the research gap between the insights from speed dating experiment and dating recommeders, this paper proposes a novel dating recommender based on speed dating experiment. The dating recommender, which is a random forest regression-based reciprocal recommender, incorporates the data from speed dating to construct an objective profile library for the participants of the speed dating event, conducts SMOTE oversampling, and makes recommendation based on the estimation of reciprocal scores between people. The proposed dating recommender achieves a significantly boosted prediction accuracy of 92.19%.

This report contains eight sections, including Introduction, Related Work, Exploratory Data Analysis, System Overview, Algorithm, Software Package Description, Experiment Result, and Conclusion.

II. RELATED WORK

This section introduces previous work related to dating recommender systems and explains the research gap this project attempts to fill.

Traditional recommender systems have mainly been based on the aim of recommending a set of items that the system predicts the user to like with high probability. Reciprocal recommender systems, although important, received rather little attention and remained an area to be explored in the field of study. The class of reciprocal recommender, as Pizzato et al. [11] defined, is the class of recommender where people are both objects and subjects of the recommendation. The most important characteristic of reciprocal recommender system is that a successful recommendation occurs only when both people like each other, or reciprocate.

Pizzato et al. [11] introduced RECON, a reciprocal recommender for online dating, which is verified on an Australian dating site. Their work shows the importance of considering reciprocity in the prediction with the improvement of success rate of the top ten recommendations from 23% to 42%.

Xia et al. [12] introduced similarity measures obtained from the users' interactions to capture the features and characteristics of online dating network and to compute a reciprocal score as the measure of the compatibility between pairs of users. The system was verified on major online dating sites in China.

Zhao et al. [14] proposed a hybrid collaborative filtering algorithm, which takes reciprocal links into account and demonstrates satisfactory performance in the recommendation of initial and reciprocal contacts.

Akehurst et al. [3] built a content-collaborative reciprocal recommender based on the hypothesis that users with similar profiles like and disliked similar people while they are liked and disliked by similar people and verified the recommender on online dating sites.

Raymond Fisman studied people's dating behavior [7] using the same dataset of the speed dating experiment that we used in our project. They found some gender differences in romantic partner selection which gave us valuable inspiration in extracting effective user profile to build our recommender.

Most of the studies in dating recommendation systems are based on online dating sites and the online profile of the users. The importance of reciprocity for dating recommendation has been identified and verified by researchers. A number of studies have explored the characteristics of users by scrutinizing the user interaction and behavior on dating sites. However, a research gap between the study of speed dating and dating recommender remains. Studies of speed dating mostly focus on the psychological explanation of people's dating behavior, whereas dating recommendation is mostly concerned with the online dating recommender.

This project addresses the research gap based on the belief that the invaluable insights on people embedded in the face-to-face real-life speed dating provided that cannot be obtained in the online virtual profile of users could be of great potential to be applied to design an improved dating recommender.

III. Exploratory Data Analysis

This section contains five subsections, including Dataset Overview, Stated Preference vs. Actual Decision Factor, Stated Preference vs. Self Measure Up, Subjective Evaluation vs. Objective Evaluation, and Match Result Correlation Analysis.

A. Dataset Overview

The dataset used in the project is a questionnaire collection from speed dating experiments during the period of 2002 to 2004, collected by Columbia Business School and is publicly available on Kaggle. Montoya [10]. The dataset contains 8378 rows and 195 columns, including participants' demographics, interests, attribute scores, etc. After each 4 minute speed date, participants were asked to rate their dates on 6 attributes: attractiveness(physical), fun, shared interest, intelligence and ambitiousness. Participants were also asked if they would like to see their date again. A match is made when both partners want a second date. The female to male ratio is roughly one to one and most of the participants are in their mid twenties and early thirties. The match rate(number of matched dates / number of total dates) of the dataset is about 20%.

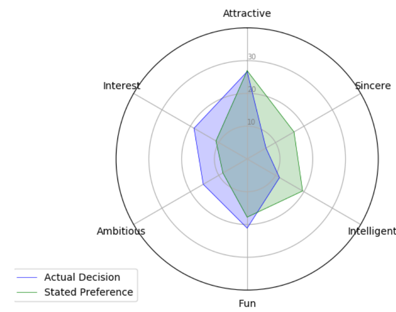

Fig. 1. Attribute comparison of male's stated preference and decision factors

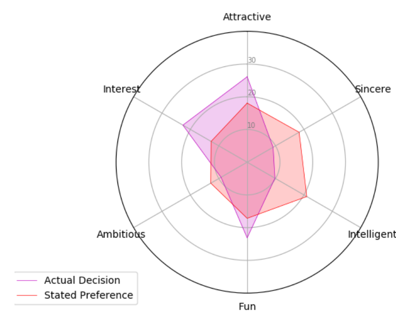

Fig. 2. Attribute comparison of female's stated preference and decision factors

B. Stated Preference VS Actual Decision Factor

This subsection explores the difference between what people state they look for and what actually influence their decision in choosing a romantic partner.

Before the participants attend the speed dating, they were asked to score on the 6 attributes according to their preference on their romantic partners. We average the scores on each attribute for male and female respectively, and get the stated preference result as shown in the radar chart Fig.1 and Fig.2. During all events, each participant meets about 10 people of the opposite sex and have 4 minutes to talk with them. After each speed date, participants were asked if they want to have a second date with this partner. We calculate how many positive responses a person got from the partners and obtain the positive response rate for each participant [9]. For example, if a man met 10 women and 6 of them indicated that they would like to meet him again, his positive response rate would be 60%. The average score a person got on each attribute is also calculated. The correlation coefficient can then be calculated for each attribute to show how strong the correlation between the positive response rate and the score of the attributes is. In order to compare with the stated preference, which is on the scale of a total of 100 points for 6 attributes, we scale the correlation coefficients of the attributes proportionally to a total of 100 points, shown as the actual decision result in Fig.1 and Fig. 2.

From Fig. 1 and Fig. 2, we have the following observations:

1) By comparing the stated preference difference between male and female, which is the green area in Fig. 1 and the red area in Fig. 2, it is noted that males strongly

prefer physical attractiveness while females prefer well-roundedness.

2) By comparing the two colored areas in one chart for male and female, respectively, we can conclude that males' actual decision perfectly matched their stated preference on the attractiveness score but the power of shared interest and ambitiousness were underestimated. Females' stated interests were drastically different from what actually influenced their decisions in most attributes. It seems that there is huge gap between what women say and what they really want.
3) The 3 most important attributes for both males and females to get positive response are attractiveness, fun, and shared interest.

C. Stated Preference VS Self Measure Up

This subsection discusses the relationship between people' stated preference and their self measure up scores on the 6 attributes.
We calculate the difference of these two metrics for male and female, respectively. The result is shown in Fig. 3.

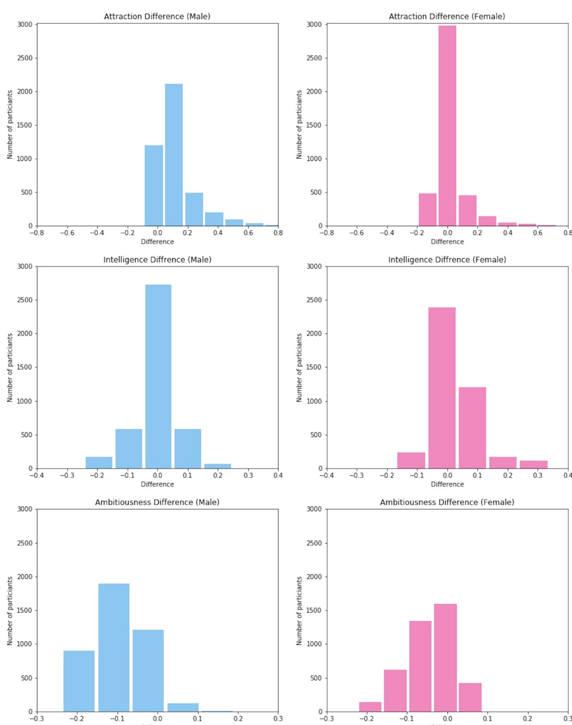

Fig. 3. Difference between people and their ideal romantic partner

In Fig. 3, the blue graphs represent male's scores on attractiveness, intelligence and ambitiousness and the pink graphs are female's corresponding scores. X axis is the difference of these two metrics, and the y axis is the number of participants. We can get some interesting findings from Fig.3:

1) 89% men prefer their romantic partner more physically attractive than themselves.
2) 69% women prefer their romantic partner more intelligent than themselves.
3) Only 5.7% men prefer their romantic partner more ambitious than themselves.

D. Subjective Evaluation VS Objective Evaluation

This section presents the comparison of the subjective evaluation which is rated by the people themselves and the objective evaluation which is the average rating by their dates. From the results shown in Fig. 4 and Fig. 5, we conclude that for both males and females, subjective scores on 6 attributes are all about 20% higher than objective scores. People appear to be overconfident about themselves in general.

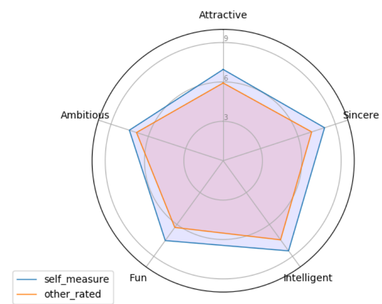

Fig. 4. Comparison of male's subjective evaluation and objective evaluation

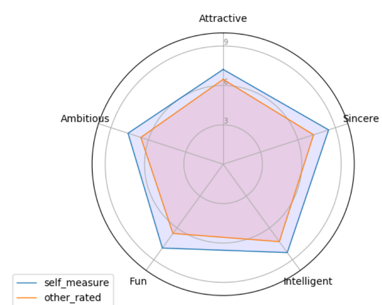

Fig. 5. Comparison of female's subjective evaluation and objective evaluation

E. Match Result Correlation Analysis

This subsection discusses the analysis of the correlation between attributes and match.
To analyze how the 6 attributes influence the match result, we calculate the correlation matrix of subjective ratings, objective ratings, decision result (dec_o) and match result (match). The result is visualized as the heatmap in Fig. 6
From the left bottom part of the Fig.6, we observe that whether one gets positive response (dec_o = 1) has a high correlation (red square) with the objective scores on attractiveness, fun, and shared interest where the correlation coefficients are 0.79, 0.66 and 0.62, respectively.
However, there is no obvious correlation between subjective scores on attractiveness and fun and dec_o with the corresponding coefficients both being 0.19. The same conclusion can be drawn on the correlation of the attribute scores and

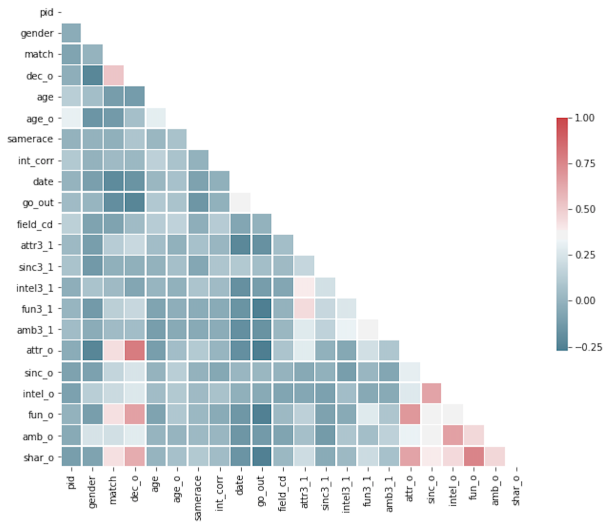

Fig. 6. Features and match result correlation matrix

match result. Therefore, it is believed that the objective evaluation data should be more accurate than subjective data for the dating recommender.

IV. SYSTEM OVERVIEW

This section describes the overall recommendation system.

The proposed dating recommender system is a reciprocal recommender system comprised of a recommender and a profile library which contains a pool of potential dates.

The profile library is the library of all the people whose data has been collected via questionnaire in the speed dating experiments. Every person is represented as a feature vector in the library. The person feature vector contains the subjective 5 attributes scores (attractiveness, fun, intelligence, sincerity, ambitiousness) of the person, the person's stated preferences for romantic partner, the person's interest, the person's gender, and the person's age.

The major innovation in the construction of this profile library is the use of objective attribute scores computed from the average of the person's dates in the speed dating events in the place of subjective attribute scores rated by the person himself/herself. The advantage of an objective profile constructed from the speed dating dates' rating over the subjective profile rated by the person himself/herself is two-fold. The first advantage is that the ratings given by the dates who met the person provide measures of the person in a real-life face-to-face scenario, which cannot be obtained in the online interactions. The second advantage is that the objective profile rated by others is a more truthful and realistic measure of the person's attribute in a dating setting than its subjective counterpart.

The recommender takes a person in the form of a feature vector as input, and outputs a list of potential partners from the profile library as the recommendations. The system structure is shown in Fig. 7.

To capture the reciprocity, the list of potential partner is recommended based on the reciprocal scores between the input person and the potential partner. The recommender recommends potential partners with the highest reciprocal score when paired with the person.

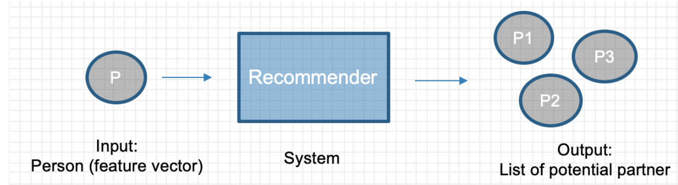

Fig. 7. The structure of the reciprocal recommender

The underlying model of the reciprocal recommender is a regression model, which computes the reciprocal scores of a given pair of people. Fig. 8 shows the structure of the regression model. The input to the regression model is a pair of two people and an additional attribute which captures important feature of the pair. The additional attribute is same_race, which is a binary field set to 1 if the two people are of the same race and 0 if the two people are of different races. The regression model outputs the reciprocal score on this pair.

The regression model may also be understood as a classifier, since the given a threshold reciprocal score the output could be converted to a binary prediction on whether the input pair makes a match. Namely, given a pair of people (x, y), reciprocal score threshold k, reciprocal score on the pair $R(x, y)$, the match prediction using binary encoding 1/0 (where match is encoded as 1 and mismatch is encoded as 0) $Match(x, y)$ can be computed as

$$Match(x,y) = \begin{cases} 1, & \text{if } R(x,y) \geq k \\ 0, & \text{if } R(x,y) < k \end{cases}$$

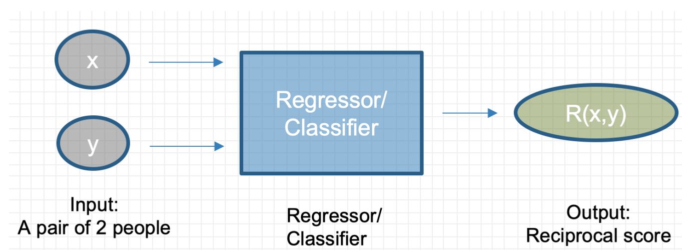

Fig. 8. The regressor model for the reciprocal recommender

V. ALGORITHM

This section discusses the algorithms and tools used to implement the recommender systems. The section is further divided into three subsections, oversampling, regression models, and tools. The oversampling subsection covers the oversampling technique used to solve the class imbalance in the dataset. The regression models subsection covers the four different models used for the regression model, including logistic regression model, neural network model, random forest model, and XGboost model. The tools subsection lists the tools used to implement the algorithms.

A. Oversampling

Due to the low match rate in speed dating, the original dataset has imbalance class ratio for the binary classes on the match attribute (match encoded as 1 and mismatch encoded as 0). The minority class accounts for only about 17% of the samples. The class imbalance results in poor performance of the models in terms of measures of F1 score, AUC, and Kohen Kappa. The accuracy measure is not meaningful given the high class imbalance.

The issue of class imbalance is not uncommon for classification tasks in machine learning. Resampling is a dataset preprocessing technique which addresses the binary class imbalance issue. Resampling approaches fall into three categories, oversampling the minority class, undersampling the majority class, and hybrid, which is the combination of oversampling minority class and undersampling majority class. Since the dataset has limited number of examples, to avoid throwing away too many training examples, the project uses oversampling of the minority class to tackle class imbalance.

Synthetic Minority Oversampling Technique, or SMOTE, is an oversampling technique that oversamples the minority class, which involves the creation of synthetic minority class examples rather than oversampling with replacement of minority class examples. Chawla et al. [4].

The synthetic minority class examples are generated in the "feature space" by taking existing minority class examples and their k nearest neighbors, and then generating synthetic minority class examples along the line joining the original example and any/all of the k minority class nearest neighbors by computing random weighted average, as illustrated in Fig. 9. Random choices on the k neighbors are made depending on the amount of oversampling required given the original dataset.

After SMOTE oversampling, with the synthetic minority class examples, the number of examples in the majority class and the number of examples in the minority class are about the same and the class imbalance is mitigated.

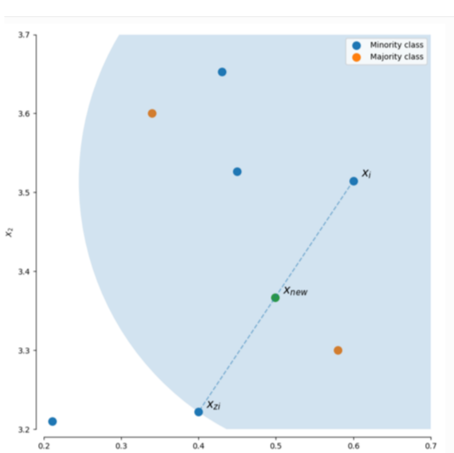

Fig. 9. SMOTE for oversampling

B. Regression Models

1) Logistic Regression Model: Let the training data be $X = \mathbb{R}^d$ and the label be $Y = \{0, 1\}$. A logistic regression model is a statistical model where the conditional probability function has a particular form [8]:

$$Y|X = x \sim Bern(\eta\beta(x)), x \in \mathbb{R}^d$$

with

$$\eta\beta(x) := logistic(x^T\beta), x \in \mathbb{R}^d$$

and

$$logistic(z) := \frac{1}{1+e^{-z}} = \frac{e^z}{1+e^z}, x \in \mathbb{R}$$

Parameters: $\beta = (\beta_1, ..., \beta_d) \in \mathbb{R}^d$
Bayes optimal classifier $f_\beta : \mathbb{R}^d \to \{0, 1\}$ in logistic regression model:

$$f_\beta(x) = \begin{cases} 1, & \text{if } x^T\beta \geq 0 \\ 0, & \text{if } x^T\beta < 0 \end{cases}$$

In our model, the output of the logistic regression model is the reciprocal score (may be interpreted as the probability that the input pair makes a match).

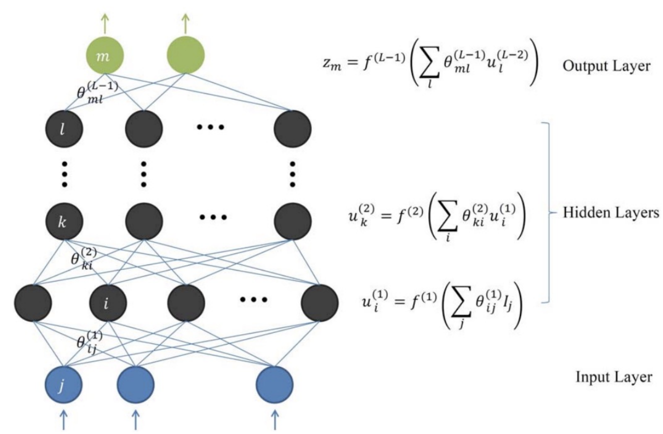

Fig. 10. An example of neural network model [13]

2) Neural Network: The structure of a neural network model is shown in Fig. 10. Each layer of the network consists of multiple neurons, each of which has an output that is a function (activation function) of a weighted sum of neurons of its preceding layer, as shown in Fig. 10. The output of the neural network z is a cascade of nonlinear transformation of input data I, mathematically expressed as

$$z = f(I, \theta) = f^{(L-1)}(f^{(L-2)}(\ldots f^{(1)}(I)))$$

where L stands for the number of layers and θ denotes the weights of the neural network Ye and Li [13]. The parameters of the model are the weights for the neurons. In supervised learning tasks, the optimal weights are learned on a training set, with known training labels.
In our model, the training label is the binary match label, where 1 denotes a match and 0 denotes a mismatch. The parameters for the training are listed below.

- loss: Mean Square Error (MSE)
- metrics: MSE, binary accuracy
- optimizer: adam
- activation function: ReLU
- number of training epochs: 300

Table I shows the neural network model structure in the project. The neural network model contains an input layer with 64 neurons, one hidden layer with 64 neurons, and an output layer with one neuron (corresponding to the reciprocal score, which is a single numeric value). To prevent overfitting the training data, two dropout layers were introduced in the model for regularization purposes.

TABLE I
NEURAL NETWORK MODEL STRUCTURE

Layer	Activation	Shape	# Param
Input	ReLU	64	4928
Dropout 1	None	64	0
Hidden 1	ReLU	64	4160
Dropout 2	None	64	0
Output	Sigmoid	1	65

3) Random Forest: Random Forest is a supervised learning algorithm which builds multiple decision trees and merges them together to make a more accurate and stable prediction. It can be used for both classification and regression problems. Random forest model often produces a good prediction result. Random Forest is trained by decision tree through the Bagging strategy.

$$Bagging + DecisionTree = RandomForest \quad [5]$$

The process is shown in Fig. 11.

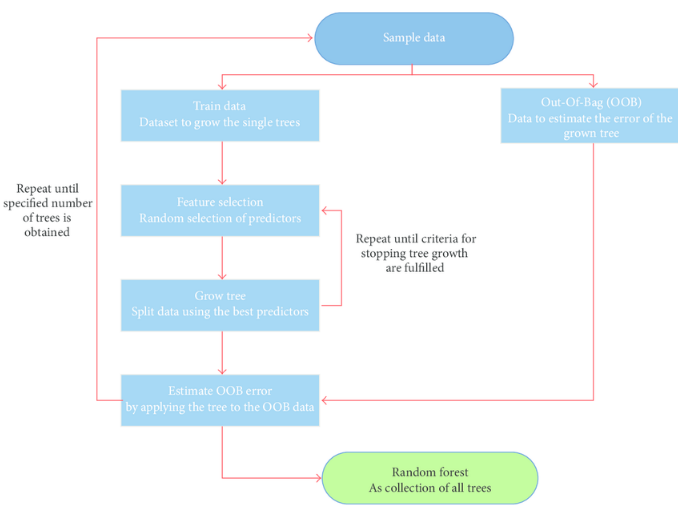

Fig. 11. The Process of Random Forest model

4) XGBoost: XGBoost implements machine learning algorithms under the Gradient Boosting framework. The algorithm reduces computing time efficiently and allocates an optimal usage of memory resources. It has many important features [1]:

1) Regularization: XGBoost has an option to penalize complex models through both L1 and L2 regularization which helps prevent overfitting.
2) Handling sparse data: XGBoost incorporates a sparsity-aware split finding algorithm to handle different types of sparsity patterns in the data.
3) Weighted quantile sketch: XGBoost has a distributed weighted quantile sketch algorithm to effectively handle weighted data.
4) Block structure for parallel learning: For faster computing, XGBoost can make use of multiple cores on the CPU which enables the data layout to be reused by subsequent iterations, instead of computing it again. This feature also appears useful for steps like split finding and column sub-sampling.
5) Out-of-core computing: This feature optimizes the available disk space and maximizes its usage when handling huge datasets that do not fit into memory.

The core algorithm is shown in Fig. 12.

Input : Data set $\mathcal{D}$.
A loss function L.
The number of iterations M.
The learning rate η.
The number of terminal nodes T

Initialize $\hat{f}^{(0)}(x) = \hat{f}_0(x) = \hat{\theta}_0 = \arg\min_{\theta} \sum_{i=1}^{n} L(y_i, \theta)$;

for m = 1,2,..,M **do**

$\hat{g}_m(x_i) = \left[\frac{\partial L(y_i, f(x_i))}{\partial f(x_i)}\right]_{f(x)=\hat{f}^{(m-1)}(x)}$;

$\hat{h}_m(x_i) = \left[\frac{\partial^2 L(y_i, f(x_i))}{\partial f(x_i)^2}\right]_{f(x)=\hat{f}^{(m-1)}(x)}$;

Determine the structure $\{\hat{R}_{jm}\}_{j=1}^{T}$ by selecting splits which maximize $Gain = \frac{1}{2}\left[\frac{G_L^2}{H_L} + \frac{G_R^2}{H_R} - \frac{G_{jm}^2}{H_{jm}}\right]$;

Determine the leaf weights $\{\hat{w}_{jm}\}_{j=1}^{T}$ for the learnt structure by $\hat{w}_{jm} = -\frac{G_{jm}}{H_{jm}}$;

$\hat{f}_m(x) = \eta \sum_{j=1}^{T} \hat{w}_{jm} \mathrm{I}(x \in \hat{R}_{jm})$;

$\hat{f}^{(m)}(x) = \hat{f}^{(m-1)}(x) + \hat{f}_m(x)$;

end

Output: $\hat{f}(x) \equiv \hat{f}^{(M)}(x) = \sum_{m=0}^{M} \hat{f}_m(x)$

Fig. 12. XGboost Algorithm

C. Tools

1) Data Analysis and Pre-processing: Packages used for the data analysis include Pandas, Numpy, and seaborn. The SMOTE algorithm is implemented using the Python imblearn package.

2) Models: The recommendation system is realized using Python3. We implemented two versions of the recommenders, one with scikit-learn and one with pyspark.ml in Spark. Keras and Tensorflow were used for the implementation of the neural network model.

3) Visualization: The PageRank is calculated with Graphframe in Spark. The visualization is implemented with d3.js.

VI. SOFTWARE PACKAGE DESCRIPTION

This section describes the software package.
The software package contains three parts, Exploratory Data Analysis, Model, and Visualization. The structure of the package is shown in Fig. 13

Package Structure

```
DatingRecommender
├── Exploratary Data Analysis
│   ├── AttributesAnalysis.py
│   ├── DataPreprocessing.ipynb
│   ├── CorrelationAnalysis.ipynb
│   └── profileComparision.ipynb
├── Model
│   ├── data_prep.ipynb
│   ├── Model-O.ipynb
│   ├── Model-S.ipynb
│   ├── regressor-spark.ipynb
│   ├── Recommender.ipynb
│   └── nn.py
└── Visualization
    ├── wave1.ipynb
    ├── original.ipynb
    ├── new.ipynb
    ├── datadealing.py
    ├── spark_wave1.py
    ├── spark_original.py
    ├── spark_new.py
    ├── graph_all_wave_1.html
    ├── graph_all_wave_ori.html
    └── graph_all_wave_new.html
```

Fig. 13. Software Package Structure

The dating recommendation system proposed, which is comprises of three parts (extraction of objective profile library from speed dating data, SMOTE oversampling on the minority class, regression-based reciprocal recommender), is contained in the Model repository.

The file data_prep.ipynb performs data engineering on the speed dating experiment dataset and outputs data files in csv format, which serves as input to the regression-based recommender. The process includes correlation-guided feature selection and the extraction of objective evaluation for a person. After the objective profile is constructed, data is prepared in the format of feature vector pairs, where each person is represented as a feature vector. The labeled training set and the objective profile library are output in separate csv files.

The file Recommender.ipynb file contains the random forest regression-based reciprocal recommender. This file conducts SMOTE oversampling on the training set to mitigate the imbalance in the dataset. After oversampling, the regression model is trained on the augmented training set. Evaluation of test accuracy is carried out. The last step is the recommendation, the recommender takes in pairs of people and then computes the reciprocal scores of the pairs to make recommendation.

VII. Experiment Results

The performance of the model is evaluated based on the test accuracy of the match prediction. The train-test split is 9:1.

The performance of all four models are compared. To verify the advantage of using the objective evaluation in building the profile library, all models are tested using both subjective profile library and objective profile library separately. The dataset used is preprocessed with SMOTE oversampling in the experiment setting. The experiment results are summarized in Table II and Fig. 15. From the figure, it is clear that the objective profile library results in better match prediction accuracy for all four models. The advantage of objective profile library is particularly evident in single models (Logistic Regression Model and Neural Network), compared to that in ensemble models (XGboost Model and Random Forest Model). This confirms the design rationale of the system and validates the empirical benefit of constructing objective profile library with speed dating data.

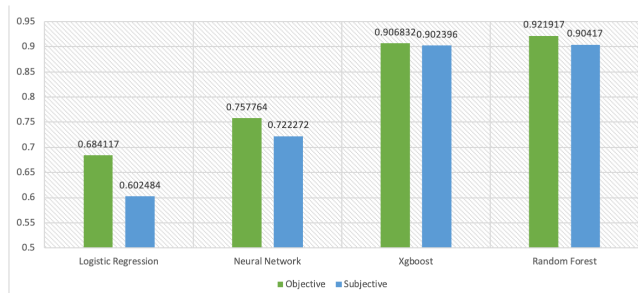

Fig. 14. Test accuracy of models with objective and subjective evaluation for the profile library

TABLE II
Test Accuracy of different models using subjective evaluation and objective evaluation

Model	subjective	Objective
Logistic Regression	60.2484	68.4117
Neural Network	72.2272	75.7764
XGboost	90.2396	90.6832
Random Forest	90.4170	92.1917

To benchmark the performance of our model, a comparison with the OpenML record on the classification task of match prediction using the same dataset is carried out. The best record on OpenMl [2] achieves test accuracy of around 0.871. Our random forest model achieves test accuracy of around 0.922. Our model outperforms the benchmark model and the performance is satisfactory.

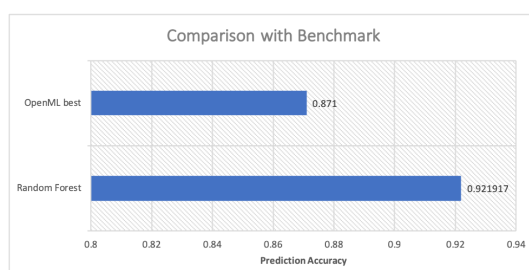

Fig. 15. Test accuracy of OpenML benchmark and our model

In order to better understand the effect of the dating recommender, we visualize the match network. In the match network, nodes denote people and edges denote an actual/predicted match. In speed dating experiment, a wave is a session in which people speed-date each other. People from different waves will not meet. We use spark to calculate PageRank of each person in every wave and visualize people and their match relationship with d3.js.

Fig. 16 shows the original match network within a single wave (wave 1). The radius of a node is proportional to the PageRank of the node. A node with larger radius corresponds to a person with more matches.

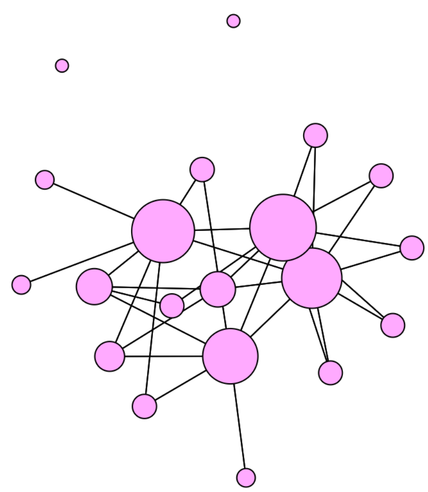

Fig. 16. Original match network of Wave 1

Fig. 17 shows the original match network of all the 21 waves, with wave 1 highlighted in purple pink. As people only speed-date within their own wave, edges in the original match network are all within the same wave. Not that there are two nodes in wave 1 without any edges, corresponding to two people who did not find a match in wave 1.

To visualize the benefit of the recommender, which makes recommendation using the entire profile library, we plotted Fig. 18, which shows the original match network of all waves together with the recommendations for the two isolated nodes in wave 1. With the profile library, people have access to a much larger pool of potential partners without having to meet all of them and are more likely to find a suitable match from the recommendations. For example, in Fig. 18, the two people in wave 1 who did not find a romantic partner (2 purple pink isolated points in Fig. 17) received recommendations of dating partners from other waves, whom the recommender predicts to be suitable for them with high reciprocity.

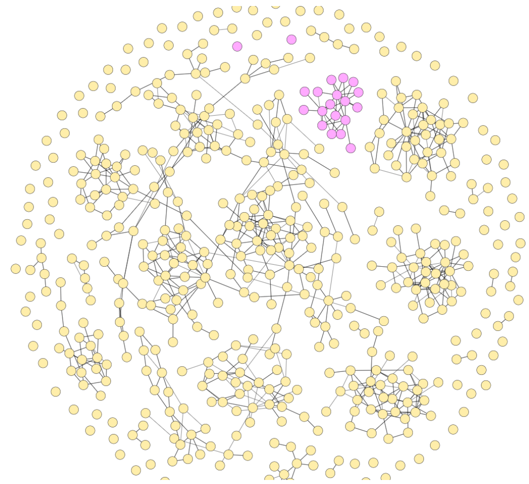

Fig. 17. Match Result of all waves without recommender

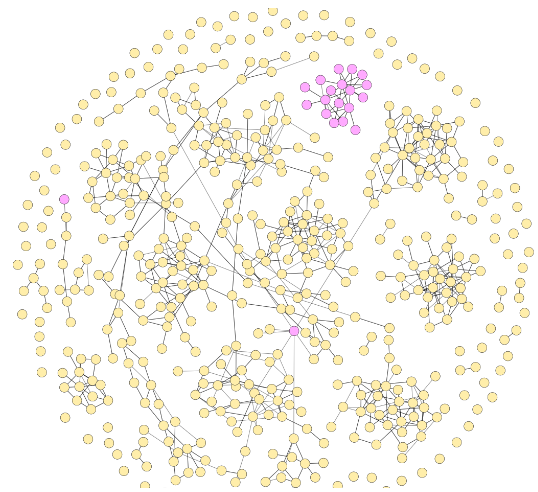

Fig. 18. Match Result of all waves with recommendations for the two isolated points in wave 1 in the original match network

VIII. Conclusion

This section draws a conclusion by summarizing the achievement of the project and outlining the future work and prospects that researchers may explore.

In summary, this project explores the research gap in dating recommendation systems by bringing the observations from speed dating into the design of dating recommender. The project proposes a novel dating recommender comprising of three major components. The first component is an approach of using objective evaluation from the speed dating experiment to construct an objective profile library of people, which leads to better performance of the recommender. The second component is the use of SMOTE oversampling technique for preprocessing. The synthetic minority class (match) examples are critical in mitigating the class imbalance issue in the original speed dating dataset. The third component is a random forest regression-based reciprocal recommdender, which makes recommendation based on the reciprocal score of potential pairs. The effectiveness of the proposed dating recommender has been validated with empirical experiment results, where the proposed dating recommender achieves a prediction accuracy of 92.19%, outperforming the 87.1% accuracy of the benchmark algorithm. The result not only validates the effectiveness of an objective profile library extracted from face-to-face speed dating events in recommender systems but also sheds light on the great potential of the study of speed dating being applied in dating recommender.

There may be other important insights from the speed dating studies such as the gender difference in choosing romantic partner illustrated in the Exploratory Data Analysis section that can be applied to improve dating recommender. Future studies may also explore alternative ways to obtain objective evaluation to construct the objective profile library. The prospect of the proposed method is that the recommendation system that combines insights from the real-life speed dating events may be incorporated into online dating sites to further improve existing online dating recommender, which is the core of perhaps the largest and fastest-growing segment of the dating service industry.

ACKNOWLEDGMENT

The authors would like to express deep gratitude to Dr. Ching-Yung Lin for his invaluable guidance throughout the project.

APPENDIX A
INDIVIDUAL CONTRIBUTION

Table III shows the contribution of the team members.

TABLE III
INDIVIDUAL CONTRIBUTION

Name	Tallis Shih-Ying Jeng	Yijia Xu	Min Fu
UNI	sj2909	yx2489	mf3200
Percentage(%)	40	30	30

REFERENCES

[1] An end-to-end guide to understand the math behind xgboost, 2018. `https://www.analyticsvidhya.com/blog/2018/09/an-end-to-end-guide-to-understand-the-math-behind-xgboost/`.

[2] Supervised classification on speeddating, 2018. `https://www.openml.org/t/146607`.

[3] Joshua Akehurst, Irena Koprinska, Kalina Yacef, Luiz Augusto Sangoi Pizzato, Judy Kay, and Tomasz Rej. Ccr-a content-collaborative reciprocal recommender for online dating. In *IJCAI*, pages 2199–2204, 2011.

[4] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002.

[5] Niklas Donges. The random forest algorithm. `https://towardsdatascience.com/the-random-forest-algorithm-d457d499ffcd`.

[6] Eli J Finkel, Paul W Eastwick, and Jacob Matthews. Speed-dating as an invaluable tool for studying romantic attraction: A methodological primer. *Personal Relationships*, 14(1):149–166, 2007.

[7] Raymond Fisman, Sheena S Iyengar, Emir Kamenica, and Itamar Simonson. Gender differences in mate selection: Evidence from a speed dating experiment. *The Quarterly Journal of Economics*, 121(2):673–697, 2006.

[8] Daniel Hsu. Slides: Logistic regression and linear classifiers. pages 22–24, 2018.

[9] Lucas Cadalzo James Hwang. The ugly truth of people decision in speed dating, 2016. `https://www.kaggle.com/jph84562/the-ugly-truth-of-people-decisions-in-speed-dating`.

[10] Anna Montoya. Speed dating experiment-what attributes influence the selection of a romantic partner?, 2015. `https://www.kaggle.com/annavictoria/speed-dating-experiment`, Last accessed on 2018-12-15.

[11] Luiz Pizzato, Tomek Rej, Thomas Chung, Irena Koprinska, and Judy Kay. Recon: a reciprocal recommender for online dating. In *Proceedings of the fourth ACM conference on Recommender systems*, pages 207–214. ACM, 2010.

[12] Peng Xia, Benyuan Liu, Yizhou Sun, and Cindy Chen. Reciprocal recommendation system for online dating. In *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015*, pages 234–241. ACM, 2015.

[13] Hao Ye and Geoffrey Ye Li. Deep reinforcement learning for resource allocation in v2v communications. In *2018 IEEE International Conference on Communications (ICC)*, pages 1–6. IEEE, 2018.

[14] Kang Zhao, Xi Wang, Mo Yu, and Bo Gao. User recommendation in reciprocal and bipartite social networks–a case study of online dating. *Intelligent Systems, IEEE*, 2014.