

# Subjectivity Grouping: Learning from Users' Rating Behavior

Hui Fang<sup>1,2</sup> Jie Zhang<sup>1</sup> Nadia Magnenat Thalmann<sup>2</sup>

<sup>1</sup>School of Computer Engineering <sup>2</sup>Institute for Media Innovation  
Nanyang Technological University, Singapore

{hfang1}@e.ntu.edu.sg, {zhangj,nadiathalmann}@ntu.edu.sg

## ABSTRACT

Considering opinions of other users (called advisors) has become increasingly important for each user in open and dynamic online environments. However, users might be subjectively different or strategically dishonest. Previous approaches on this problem generally suffer from the issue of limited (especially shared) historic experience when tracking each individual advisor's behavior. In this paper, instead, we model each advisor as part of groups by proposing a two-layered clustering approach. Specifically, in the first layer, the agent of each user clusters her advisors into different subjectivity groups and dishonest types, with respect to their rating behavior. In the second layer, each advisor is assigned to groups with respective membership degrees. Finally, each agent adopts an alignment approach to help its user align advisors' ratings to the ones of her own. Experimental results on both simulations and real data verify that our approach can better help users utilize ratings provided by advisors in opinion evaluation and recommender systems.

## Categories and Subject Descriptors

I.2.11 [Artificial Intelligent]: Distributed Artificial Intelligence—Intelligent agents

## Keywords

Subjectivity, Dishonest types, Clustering, Rating alignment

## 1. INTRODUCTION

In large and open online communities, users may often encounter other entities which they have no previous experience with or prior knowledge of. In this case, they usually rely on the experience or knowledge of other users (advisors), to choose which entities to interact with. However, in these environments, users could freely express their opinions with limited administration, and the quality of opinions may then vary. One common reason is that, users are subjectively different [5], which thus leads to discrepancy of users' opinions (ratings) towards same entities. For instance, a satisfactory experience of a user may turn to be an unsatisfactory one for another user. Moreover, some users might be dishonest and lie about their experience with entities. For example, to de-

mote an entity, a user may intentionally provide a negative rating to the entity while the real experience is successful.

Some approaches have been proposed to address the above mentioned problem by using *social networks* [9] where opinions coming from *friends* in a user's social network are greatly valued. However, being friends of the user does not imply to share the same subjectivity (i.e. preference) with the user. In addition, computational trust models in multi-agent systems area argue to evaluate the quality of opinions by designing agents for users to model the trustworthiness of advisors on the basis of their past activities. The underlying assumption here is that a more trustworthy advisor could provide opinions of higher quality. However, they mainly focus on the dishonesty problem (e.g. unfair ratings) [21] and merely on the user subjectivity difference [5], or could not well distinguish dishonesty and subjectivity from each other [17,20]. Besides, most of them strive to model an advisor's trustworthiness for a user on the basis of either the common experience between the user and the advisor [21] or the whole historic experience of them [22]. Consequently, these models, on the basis of machine learning techniques, might fail with great probability since 1) the user and advisor have limited past experience in a community, especially limited common experience towards same entities; and 2) a user's behavior is dynamic and evolving, which greatly increases the difficulty of tracking her behavior.

In view of the aforementioned problems, we propose a clustering-based method that categorizes users into different groups with respect to their rating behavior. We distinguish subjective users from dishonest ones. In other words, users with similar subjectivity in rating entities are grouped together, and dishonest users are labeled as outliers. We further examine these dishonest users by dividing them into three different types. Consequently, on the one hand, each user can not only directly employ other users' opinions in the same subjectivity group, but also effectively and rationally use the opinions of those in different subjectivity groups. On the other hand, the user might even take advantage of opinions provided by those dishonest advisors whose rating behavior follows definite patterns, while opinions of the dishonest advisors with no static patterns would be ignored. Accordingly, users can maximally and effectively adopt advisors' opinions with our method. We prefer unsupervised learning (i.e. clustering) over supervised learning since we have no prior knowledge about the size and number of the subjectivity groups. Besides, the number of groups would also be varied as the dynamic change of users (new users join, and users might leave). The advantages of exploring

**Appears in:** *Alessio Lomuscio, Paul Scerri, Ana Bazzan, and Michael Huhns (eds.), Proceedings of the 13th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2014), May 5-9, 2014, Paris, France.*

Copyright © 2014, International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). All rights reserved.

group information through clustering instead of personal information (for opinion evaluation) can mitigate the gap between limited data of an individual user and sufficient data required for accurately learning the user’s behavior. Moreover, we identify a set of features that can well capture the dynamic behavior of users.

Computationally, each user in the system is equipped with a software agent. Each agent clusters its user’s advisors according to their historic experience. More specifically, a two-layered clustering approach is proposed for each agent to cluster advisors of its user into subjectivity clusters and three dishonest types (i.e. direct dishonest, disguise dishonest and misguidance dishonest). The first layer employs the DENCLU algorithm [12] *twice* to identify advisors as either *subjective* or *dishonest*. Then, in the second layer, each advisor (not including misguidance dishonest ones) is assigned to two closest clusters with respective membership degrees (sum up to 1). Given the clustering results, each agent further adopts a simple but effective group alignment algorithm that helps its user align advisors’ ratings to the ones of her own. We conduct experiments on two different environments to validate the effectiveness of our approach: a distributed simulated e-marketplace for opinion evaluation, and three real datasets obtained from Epinions, Flixster and FilmTrust for rating prediction in recommender systems. Experimental results verify that our approach can help users better utilize ratings provided by advisors.

## 2. RELATED WORK

Social network based methods, e.g. trust-aware recommender systems where users’ information sources can be enriched by past experience or recommendations of trust neighbors (advisors), are designed with the assumption that an advisor in a user’s social network could provide more reliable opinions to the user. For example, Golbeck [9] introduces a trust-flow-based method (called TidalTrust) for rating predictions of target items. Guo et al. [10] propose to merge the ratings of trust neighbors and thus better model the preferences of users to resolve the data sparsity problems in recommendation systems. However, a generally agreed proposition states that people being friends of each other may not share similar preferences [14]. In other words, friends (trust neighbors) of a user might be intrinsically honest, but unnecessary to share the same subjectivity with the user.

Other than these memory-based approaches, the model-based approaches are also employed in recommender systems. Related approaches to our work include clustering-based approaches, and matrix-factorization ones. Clustering based methods reduce the search space in recommender systems by employing clustering techniques such as weighted co-clustering [8] method to cluster similar users or entities, and hence ratings of clustered users or entities are integrated to make predictions for corresponding users. However, they only employ information of users in the same cluster, while in our approach information of users from other clusters can also be effectively incorporated. Matrix factorization has become popular recently in recommender systems. It fits the user-entity rating matrix using low-rank approximations, i.e. user-feature matrix and entity-feature matrix, and employs low-rank matrices to make further predication. For example, Minh and Salakhutdinov [16] propose a probabilistic matrix factorization model by assuming Guassian observations on observed user-entity ratings. SocialMF, proposed by Jamali

and Ester, combines a basic matrix factorization method with a trust-based approach to further enhance recommendation [13]. However, they ignore the dishonesty problem when considering other users’ recommendations.

Different trust models have also been proposed in multi-agent systems to model the trustworthiness of advisors for evaluating their opinions. Some of them, such as [21, 22], focus on addressing opinions of low quality intentionally provided by dishonest advisors. Due to the ignorance of subjectivity difference between advisors and users, they may misuse some important information caused by subjectivity difference, rather than dishonesty. Some other approaches [5], although rare, only consider subjectivity difference between users and advisors, but ignore the dishonesty of advisors. They may mistakenly treat dishonest advisors as those having subjectivity difference with users.

Some recent approaches have also been proposed to model both dishonesty and subjectivity of advisors. BLADE [18] applies Bayesian learning to model the correlations between entities’ properties and ratings of users and advisors. However, if the correlations learned for users’ ratings are based on entities’ properties that are different from those for advisors, it is likely that advisors having subjectivity difference are treated as dishonest ones. HABIT [20] extends BLADE by additionally considering third party information, but it might suffer from the same problem as BLADE. ProbCog [17] is a two-layered behavioral modeling approach that firstly filters dishonest advisors and then discounts other advisors’ ratings according to their subjective trends. However, it has the assumption that advisors providing very different ratings with a user are dishonest to the user. Hence, advisors having large subjectivity difference with the user will be misclassified as dishonest. PRep [11] learns the advisors’ behavior using Bayesian learning and then adjusts their opinions (no matter biased or unbiased) according to the learned types of advisors. However, similar to ProbCog and BLADE, it would wrongly treat some dishonest users as subjective ones, or vice versa. Besides, almost all the aforementioned approaches have strict requirements on the number of available interactions between users. PGTM [4] explicitly distinguishes (dis)honesty and subjectivity difference by modeling them using different sources of rating information, and captures their relationships with trust through the influence of chains in a probabilistic graphical model. However, it ignores the fact that dishonesty and subjectivity are overlapped with each other to certain extent from the perspective of information sources.

On the contrary, our approach distinguishes subjective users from dishonest ones. Besides, it addresses the data sparsity problem by considering advisors as part of groups. Ratings of advisors are aligned to the ones of the user’s own. Moreover, the set of features used to represent advisors can capture their dynamic and evolving behavior.

## 3. SUBJECTIVITY AND DISHONESTY

In this section, we summarize the definitions and studies of subjectivity and dishonesty, respectively.

### 3.1 Subjectivity

People are subjectively different. Opinions (or ratings) of each user in online communities imply a certain degree of subjectivity of this user. For example, in a rating system, user A rates a comic book as “5”, while user B rates the

same book as “4”. In this case, we could conclude that A is a little more positive than B in this context of comic books. Hence, we choose to learn users’ subjectivity from their rating behavior. Subjectivity analysis has been actively studied in various applications such as customers’ opinion mining in online review forums and multi-document summarization [1]. Our research differs from those in the literature from two perspectives: 1) most previous research focuses on textual information, while we target at ratings; 2) in the literature, subjectivity analysis is often defined as a binary-classification task, i.e. subjective or non-subjective. This is opposed to the clustering task in our research, where users are clustered into multiple clusters (e.g. positive, neutral and negative in [17]), and users in the same subjectivity groups tend to provide similar ratings towards same entities. The number of subjectivity clusters is uncertain ( $\geq 2$ ).

### 3.2 Dishonesty

The current online environments open the door for dishonest users (i.e. attackers) to manipulate online rating systems by selfishly promoting or maliciously demoting certain entities [6]. According to the principle of veracity [15], users usually deceive for a reason, that motives producing deception is usually the same as that guides honesty. For example, in an e-commerce environment, ratings of an entity could not only reflect its popularity and reputation, but also greatly affect its sales. Hence, attackers tend to strategically manipulate their ratings in order to fulfill their goals (maximize their own profits or demote other competitors).

Extending from the work of Feng et al. [6], we mainly recognize three types of dishonest users: 1) *direct dishonest* users consistently provide dishonest ratings to all entities. This is the most naive attacker model; 2) *indirect dishonest* users behave honestly (by providing honest ratings) to entities of certain types, but perform dishonestly to those of other types. They can also be called as *disguise* users, since they disguise themselves as honest in some scenarios to gain trust of other users; 3) *misguidance dishonest* users provide dishonest and honest ratings to entities following no static pattern. Dishonest users of this type is very difficult to track since their behavior is extremely sophisticated.

## 4. OUR APPROACH

In this section, we present our approach in great details. Firstly, we genuinely introduce our research problem and procedural framework of the proposed method. Secondly, we identify the features employed in the clustering algorithm based on the intuitions and related work in the literature. Thirdly, we describe the details of the proposed two-layered clustering approach. Finally, we present a simple but effective group alignment algorithm, which shows how the results of our clustering approach can be effectively used in applications such as opinion evaluation and recommender systems.

### 4.1 Procedural Framework

In our approach, each user is equipped with a software agent being responsible for managing the ratings of its own user and other users (advisors) towards entities in the online communities. Each user has a set of past interactions with some entities and provides a rating in the range of  $[0, 1]$ <sup>1</sup>

<sup>1</sup>Multi-nominal ratings in online communities could be normalized into the rating scale of  $[0, 1]$ .

for each interaction. We assume that a user  $u$  (equipping with an agent  $a$ ) has previously interacted with a set of entities  $E$ . Based on these interactions (*rating behavior*), the user is described by a feature vector  $F^m = \{f_1, f_2, \dots, f_m\}$ , where  $m$  is the number of features. Similarly, agent  $a$  can also obtain the corresponding feature vectors for its user’s advisors (*feature extraction*). The set of advisors is denoted by  $U^u$ . For implementation,  $U^u$  mainly involves the users who previously interacted with the same entities as user  $u$ . If  $u$  has limited historic interactions, agent  $a$  could expand  $U^u$  by also actively propagating its requests of more information to the agents of other users until  $U^u$  is considerably large. Then, agent  $a$  conducts our proposed *cluster analysis* towards  $u$  and advisors in  $U^u$ , which are clustered into different groups: either subjectivity groups or dishonest groups (outliers). Consequently, agent  $a$  can effectively utilize ratings provided by advisors given the results of the cluster analysis. The basic process is presented in Figure 1. As shown in the figure, we also describe a simply alignment algorithm that employs the outcome of the cluster analysis into various real applications, such as trust models for opinion evaluations, and recommender systems.

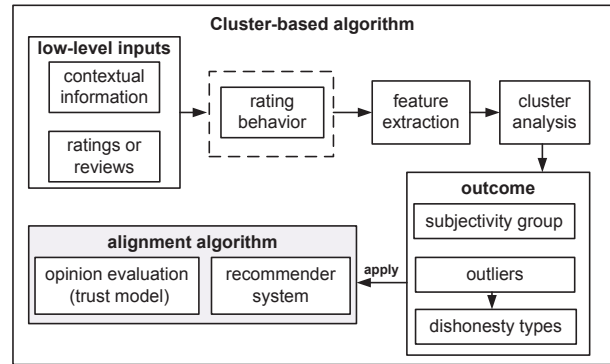


Figure 1: Procedural design of the framework

### 4.2 Feature Identification

In this part, we aim to identify feature vector  $F^m = \{f_1, f_2, \dots, f_m\}$ . Each feature for our research problem is expected to satisfy two objective requirements: 1) to maximize the distance between users of different subjectivity (including the dishonest ones); and 2) to minimize the distance between users of the same subjectivity (or dishonest type). To fulfil this goal, the most important issue is to understand the clues distinguishing the subjective behavior and dishonest behavior regarding to ratings (as presented in Section 3).

Following the reliability research in medical domain [19], we deem that ratings from different users could be mainly compared with respect to two perspectives: inter-user agreement and intra-user agreement. The former one indicates the scenario where different users rate a same entity, while the latter one refers to the scenario where a user provides ratings to different entities of same quality. The honest users of the similar subjectivity have higher values on the two perspectives –provide similar ratings to the same entities as the group members, and consistently provide similar ratings to the products of same quality. Accordingly, subjective users of significantly different types might have lower values on inter-user agreement, but higher values on intra-user agreement, while dishonest users have lower values on both per-

spectives. Additionally, another guideline for feature selection is: *rating behavior of subjective users is sort of consistent (might evolve with a low speed), while that of dishonest ones is comparatively unstable (strategically changing).*

With these guidelines in mind, in general, we first identify several propositions in the literature or by intuition, and then directly indicate some features related to these propositions for our research problem.

**PROPOSITION 1.** *The rating difference between users in the same subjectivity group is much smaller than that between users coming from different groups. In other words, users in the same subjectivity group tend to provide similar ratings. Hence, the related features for user  $u$  are variance of ratings,  $u$ 's average rating difference with other users regarding to same entities ( $r_d^u$ ), variance of  $u$ 's rating difference with other users regarding to same entities ( $v_d^u$ ), etc.*

$$r_d^u = \text{avg}_{i \in U^u, x \in E} (r_x^u - r_x^i) \quad (1)$$

$$v_d^u = \text{var}_{i \in U^u, x \in E} (r_x^u - r_x^i) \quad (2)$$

**PROPOSITION 2.** *A user's rating to an entity may be subjectively impacted by other users' ratings (positive or negative) to the entity. In this sense, average or midpoint of ratings to the entity could be treated as a (quality) benchmark for the entity [7]. The associated features are  $u$ 's average rating difference with the benchmark rating ( $r_b^u$ ), variance of  $u$ 's rating difference with the benchmark rating ( $v_b^u$ ), etc.*

$$r_b^u = \text{avg}_{x \in E} (r_x^u - \bar{r}_x) \quad (3)$$

$$v_b^u = \text{var}_{x \in E} (r_x^u - \bar{r}_x) \quad (4)$$

where  $\bar{r}_x$  is the average rating towards entity  $x$ , i.e. benchmark rating.

**PROPOSITION 3.** *The rating difference between two users would be relatively stable for two subjective users, but unstable for those where one or both of them are dishonest. Hence, corresponding features are the average of  $u$ 's rating difference with other users based on the commonly rated entities ( $r_c^u$ ), the variance of  $u$ 's rating difference with other users based on the commonly rated entities ( $v_c^u$ ), etc.*

$$r_c^u = \text{avg}_{i \in U^u, x \in E(u,i)} (r_x^u - r_x^i) \quad (5)$$

$$v_c^u = \text{var}_{i \in U^u, x \in E(u,i)} (r_x^u - r_x^i) \quad (6)$$

where  $E(u, i)$  is the common entities between user  $u$  and  $i$ .<sup>2</sup>

**PROPOSITION 4.** *For users having relatively fewer interactions, or lower interacting frequency with entities, they might be more reluctant to provide dishonest ratings. This is mainly because in this scenario, the dishonest ratings provided by them might be relatively less influencing than those provided by other dishonest users (with a larger number of historic interactions or higher interacting frequency). One possible exception would be dishonest users having sybil*

<sup>2</sup>Note that Proposition 1 considers each user's all rating history, whereas Proposition 3 considers only partial history.

identities. In this case, related features are *the total number of interactions, skewness of ratings ( $r_s^u$ ) [3], the interacting frequency, etc.*

$$r_s^u = \frac{\sum_{x \in E} (r_x^u - \bar{r}^u)^3}{(N-1) \times s^3} \quad (7)$$

where  $\bar{r}^u$ ,  $s$  and  $N$  are the mean, the standard deviation, and the number of the ratings provided by user  $u$ , respectively.

**PROPOSITION 5.** *A user might vary her behavior under different contexts. This proposition holds true for both dishonest users (e.g. the disguise dishonest users) and honest users (e.g. users might have different subjectivity in evaluating entities of different categories). Context information (e.g. category or time) is especially valuable in detecting dishonest behavior [2]. Each user's features of this type might be the variance of the user's rating for entities of different contexts (e.g. categories), the variance of the user's ratings in different time interval, etc.*

All the features are normalized to be in the range of  $[0, 1]$ . Note that our objective of this section is not to figure out all the features or the best candidate features to achieve the best performance, but, more importantly, to *provide hints on identifying features for our research problem.* As the value of these features can mirror the change of users' behavior with the change of time, we argue that the clustering algorithm on the basis of these features is capable of well addressing users' dynamic and evolving behavior.

### 4.3 Cluster Analysis

We propose a two-layered clustering approach for each agent to cluster its user's advisors into different subjectivity groups and dishonesty types (outliers). The first layer mainly employs a density based cluster algorithm – DENCLU [12] that crisply clusters users into different groups and outliers. The second layer is a fuzzy process which softly smooths and justifies the clustering results.

#### 4.3.1 The First Layer: DENCLU

We choose DENCLU mainly because it can well deal with the environments with lots of noise, which allows us to flexibly address different scenarios, even those with a great proportion of dishonest users. Besides, it has a firm mathematic basis [12] instead of only heuristic clues. Moreover, each agent can efficiently conduct the algorithm even towards a relatively large set of advisors since it is very fast. This is in conformity with our research scenario where users can easily obtain a considerably large amount of advisors as they interact with more entities.

DENCLU is based on the idea that the influence of each data point could be modeled formally using an influence function which describes the impact of the data point within its neighborhood. As mentioned in Section 4.1, for agent  $a$ , its user  $u$  and  $u$ 's neighborhood  $U^u$  (advisors) are described by the  $m$ -dimensional feature space  $F^m$ . For simplification, we denote the set of users including  $u$  and advisors in  $U^u$  as  $U^{u+}$ . The influence function is symmetric, continuous, and differentiable, such as Square Wave function and Gaussian function [12]. In our case, we choose Gaussian function since it could well represent most real-world cases. The influence of user  $x$  regarding to  $u$  ( $x \in U^{u+}$ ),  $f_B^u(x)$ , is defined as:

$$f_B^u(x) = e^{-\frac{d(u,x)^2}{2\sigma^2}} \quad (8)$$

where  $\sigma$  is a predefined threshold, and  $d(u, x)$  is defined as the Euclidean distance<sup>3</sup> between user  $u$  and  $x$ :

$$d(u, x) = \sqrt{\sum_{i=1}^m (f_i^u - f_i^x)^2} \quad (9)$$

Accordingly, the density function of user  $u$  is defined as the sum of the influence functions of  $u$ 's advisors. Thus,  $u$ 's density function  $f_B^D(u)$  is defined as [12]:

$$f_B^D(u) = \sum_{x \in U^{u+}} f_B^x(u) = \sum_{x \in U^{u+}} e^{-\frac{d(u,x)^2}{2\sigma^2}} \quad (10)$$

On the basis of the density function, we then need to find all the density attractors  $x_j^*$  ( $x_j^* \in U^{u+}$ , where  $j = 1, \dots, n_d$ , and  $n_d \leq \|U^{u+}\|$ ). Informally, density attractors are local maxima of the overall density function. If a point  $x \in U^{u+}$  is *density-attracted* to a density attractor  $x_j^*$ , and  $x_j^*$  has a relatively bigger influence ( $\geq \xi$ , where  $\xi$  is a predefined bound),  $x$  belongs to cluster with  $x_j^*$ , otherwise,  $x$  is an outlier.

The DENCLU algorithm mainly consists of two steps:

**Step 1** (pre-clustering stage): the representation of  $U^{u+}$  is divided into  $m$ -dimensional hypercubes, with an edge-length of  $2\sigma$ . Only hypercubes that contain at least one data point (i.e. one user) are determined, and called as *populated cubs*. The set of populated cubs is denoted by  $C^u$ , where the number of hypercubes  $\|C^u\|$  is sensitive to the choice of  $\sigma$ .

**Step 2** (main stage – clustering stage): only the highly populated cubs (contain a pre-defined number of data points) and cubs which are connected to a highly populated cube, denoted as  $C_q$ , are considered in determining clusters. The basic process is the same as [12].

The quality of DENCLU depends on a good choice of two parameters ( $\xi, \sigma$ ).  $\sigma$  impacts the influence of a user in her neighborhood, and  $\xi$  determines the minimum level for significant density-attractor. Appropriate ( $\xi, \sigma$ ) could help us to well deal with the environments with different noise levels (i.e. different levels of the proportion of dishonesty users in the online community). Specifically, it is better to set  $\xi$  to be bigger than the noise level (defined in [12]).

**The process of our first layer is as follows:** firstly, agent  $a$  conducts DENCLU towards  $U^{u+}$  with appropriate parameters ( $\xi_1, \sigma_1$ ), and then the output is subjectivity clusters  $C_s = \{c_1^u, c_2^u, \dots, c_{u_s}^u\}$  where  $u_s$  is the number of subjectivity clusters and  $u \in c_t$  ( $t \leq u_s$ ), and dishonest users  $U_d^u$  (i.e. outliers) regarding to  $u$ .

Secondly, agent  $a$  further conducts DENCLU with new appropriate ( $\xi_2, \sigma_2$ ) pair<sup>4</sup> towards  $U_d^u$  by controlling the cluster number to be 2. Thus, two types of dishonest users i.e. *direct dishonest* users ( $c_{d1}^u$ ) and *disguise dishonest* users ( $c_{d2}^u$ ), are identified.  $C_d^u = \{c_{d1}^u, c_{d2}^u\}$ . And, the outliers of this round are considered as *misguidance dishonest* ones, and their ratings are **discarded** (or filtered out) for the next layer and applications.

### 4.3.2 The Second Layer: Fuzzy Smoothing

For users in the border area of each cluster, it might be over-positive to determine them as belonging to only one

<sup>3</sup>The choice of distance functions is application-varied. For example, for the application where users have many features with 0 values, the hamming distance is more suitable.

<sup>4</sup>Here, the noise level is decided by the third type of dishonest users.

cluster. This is in accordance with the real-world scenario where some users have the mixed subjectivity of more than two distinct subjectivity groups. To resolve this problem, in the second layer, agent  $a$  conducts *fuzzy smoothing process* towards users in  $U^{u+}$  (particularly  $C_s^u$  and  $C_d^u$ ).

For each user  $u_c \in c_{c1}^u$  ( $c_{c1}^u \in C_s^u, c1 \leq u_s$ ), we first compute the distance of  $u_c$  with each of other clusters in  $C_s^u$  (not including  $c_{c1}^u$ ). The nearest cluster  $c_{c2}^u$  ( $c_{c2}^u \in C_s^u$ ) to  $u_c$  is selected. Afterwards,  $u_c$  is considered as belonging to set  $c_{c1}^u$  and  $c_{c2}^u$  with membership  $m_{c1}^{u_c}$  and  $m_{c2}^{u_c}$  using *piecewise membership function*, respectively:

$$\begin{cases} m_{c1}^{u_c} = \frac{d(u_c, \text{mean}(c_{c1}^u))}{d(u_c, \text{mean}(c_{c1}^u)) + d(u_c, \text{mean}(c_{c2}^u))} \\ m_{c2}^{u_c} = 1 - m_{c1}^{u_c} \end{cases} \quad (11)$$

Similarly, for each user  $u_c \in C_d^u$ , she is considered as belonging to set  $c_{d1}^u$  and  $c_{d2}^u$  with membership  $m_{d1}^{u_c}$  and  $m_{d2}^{u_c}$  respectively:

$$\begin{cases} m_{d1}^{u_c} = \frac{d(u_c, \text{mean}(c_{d1}^u))}{d(u_c, \text{mean}(c_{d1}^u)) + d(u_c, \text{mean}(c_{d2}^u))} \\ m_{d2}^{u_c} = 1 - m_{d1}^{u_c} \end{cases} \quad (12)$$

Note that each agent conducts cluster analysis based on the historic interactions of its own user as well as those of other users in the user's neighborhood (advisors). For a user with limited experience, the agent could request the information (feature vectors) of advisors from other agents about which the agent already has good knowledge (evaluation).

## 4.4 Group Alignment

Agent  $a$  manages its own user  $u$ 's rating system, including the ratings provided by  $u$ 's advisors in  $C_s^u$  and  $C_d^u$ . The ratings provided by advisors of the misguidance dishonest type are directly discarded.

We propose a simple alignment algorithm for each agent to effectively adopt the ratings provided by advisors. We call it *group alignment algorithm* mainly because we differ the rating difference between two users according to the ratings of the other users in the same clusters with them (see Equation 13). By doing so, each user could benefit from the collective knowledge of group users, and avoid the inaccuracy caused by limited information, which is the case for most users. Let us assume one user  $u$  belongs to two clusters  $c_{u1}^u$  and  $c_{u2}^u$  with respective membership degree  $m_{u1}$  and  $m_{u2}$ . And, there is another qualifying user  $u_c \in C_s^u \cup C_d^u$  belonging to two clusters  $c_{c1}^u$  and  $c_{c2}^u$  with corresponding membership  $m_{c1}$  and  $m_{c2}$ . In this case, each rating  $r$  provided by  $u_c$  to entity  $x$  would be aligned to that of  $u$ ,  $r_c^u$ , in the sense that adapts to  $u$ 's evaluation criterion:

$$r_u = r + \overbrace{\text{mean}(r_{c_{u1}^u}) * m_{u1} + \text{mean}(r_{c_{u2}^u}) * m_{u2}}^{u's \text{ group center}} - \overbrace{\text{mean}(r_{c_{c1}^u}) * m_{c1} + \text{mean}(r_{c_{c2}^u}) * m_{c2}}^{u_c's \text{ group center}} \quad (13)$$

where  $\text{mean}(r_{c_{u1}^u})$  is the average of ratings of users from  $c_{u1}^u$  to the entities of the same type as  $x$ . The right hand of Equation 13 (not including  $r$ ) indicates the rating difference between  $u$  and  $u_c$  to an entity of the same type as  $x$ .

Each agent depends on the cluster analysis and group alignment approach to manage its own user's rating system. Our approach can be directly used in opinion evaluation in online communities (where in most MAS cases, trust models are employed), and recommender systems.

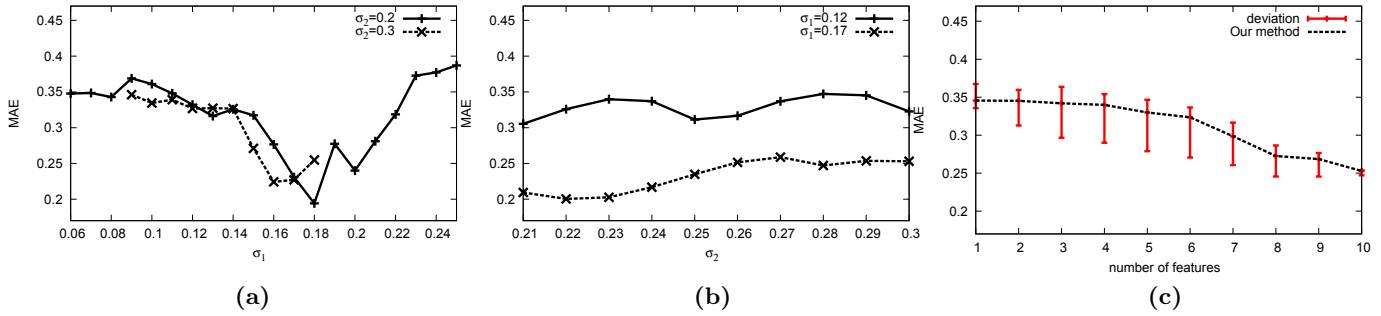


Figure 2: Performance change of our method by varying (a)  $\sigma_1$ ; (b)  $\sigma_2$ ; (c) number of features ( $\sigma_1 = 0.12$ ,  $\sigma_2 = 0.30$ ).

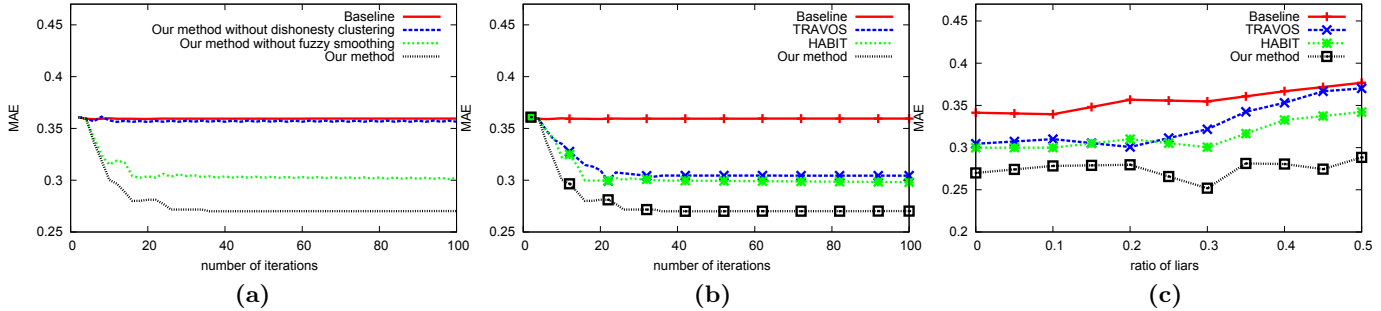


Figure 3: Performance comparison (a) and (b) as the change of iterations; (c) when varying the ratio of liars.

## 5. EXPERIMENTS

We conduct experiments in two different environments, a distributed simulated e-marketplace for checking the effectiveness of our approach in opinion evaluation (trust models), and three real data sets obtained from Epinions, Flixster and FlimTrust for validating our approach for rating prediction in recommender systems.

### 5.1 Simulated E-marketplace

We examine the effectiveness of our approach in a distributed environment by conducting comparisons with the competing trust models.

#### 5.1.1 Experimental Settings

We simulate an e-marketplace involving 55 sellers and 500 buyers. In our simulation, each seller is assigned a base reputation scaled from 0.5 to 1 with step of 0.05. For example, if base reputation of a seller is 0.5, the seller has a probability 0.5 of conducting transactions successfully. Buyers rate a seller with a value ranged in  $[0,1]$ . Buyers also have different subjectivity in rating their experience with same sellers. We define each buyer’s subjectivity as a constant, which follows a Gaussian Distribution across all buyers and is ranged in  $[-1,1]$ . For example, if a buyer’s subjectivity value is 0.5, it means that she would rate sellers as 1 if sellers’ base reputation is bigger than 0.5. In each iteration, whether a buyer would conduct a transaction is decided by a predefined probability value. A buyer always seeks to conduct transaction with the seller of the highest reputation value (computed using trust models) from her view.

Here, besides our approach, we implement a baseline approach without considering subjectivity and dishonesty, which computes the reputation of sellers by directly averaging the ratings collected from other buyers for the sellers. We also choose to implement the TRAVOS model [21], which is a representative approach in the set of filtering approaches.

As methods that consider both dishonest and subjective rating problem, the HABIT [20] method is chosen instead of PGMT [4] and PRep [11] because the latter two are complicated to implement for our scenario (refer to the details in Section 2).

We evaluate their performance in computing the reputation of sellers. The performance of an approach is measured as the mean absolute error (MAE) between the reputation of sellers computed for each buyer using the approach, and the reputation of sellers using the ratings according to each buyer’s own subjectivity (the ground truth about the reputation of sellers with respect to the buyer).

#### 5.1.2 Results and Discussion

Here, we first check the effectiveness of each part of our approach. Then, we present the performance of our approach and three benchmark approaches. We also examine these approaches in details by varying the ratio of dishonest buyers in the simulated environment.

**Model Effectiveness.** In Figures 2(a) and 2(b), we analyze the impact of  $\sigma$  used in the first DENCLU algorithm ( $\sigma_1$ ), and the second DENCLU algorithm ( $\sigma_2$ ), respectively. Note that  $\sigma_1 < \sigma_2$ . Specifically, Figure 2(a) presents the performance of our method by varying  $\sigma_1$  while fixing  $\sigma_2 = 0.2$  and 0.3, respectively. Figure 2(b) shows the performance of our method by changing  $\sigma_2$  while fixing  $\sigma_1 = 0.12$  and 0.17, respectively. As can be seen, our method is sensitive to  $\sigma_1$ , and relatively insensitive to  $\sigma_2$ . With regard to different  $\sigma_2$ , our method can consistently achieve the best performance when  $\sigma_1 \simeq 0.17$ . This implies that at that level of  $\sigma_1$ , we can obtain the best number of the subjectivity groups (granularity level) for our rating alignment. Given that value of  $\sigma_1$ , the MAE approximates to 0.18, which demonstrates almost 50% improvement over the baseline method (MAE  $\simeq$  0.35).

We also investigate the effectiveness of features used in our approach. We select 10 features to represent a buyer for

Table 1: The performance comparison of different approaches

Methods	Epinions		Flixster		FilmTrust	
	RMSE	MAE	RMSE	MAE	RMSE	MAE
<b>Our method</b>	<b>0.4058</b>	<b>0.1046</b>	<b>0.9057</b>	<b>0.6824</b>	0.8515	0.6566
TidalTrust	0.6759	0.5756	1.2449	0.9846	0.9687	0.7564
CoClustering	0.6398	0.3834	0.9263	0.6926	0.8472	0.6401
PMF	0.6498	0.4260	0.9264	0.6902	<b>0.8384</b>	<b>0.6405</b>
SocialMF	1.6427	1.3508	1.3749	1.0536	0.8602	0.6590

our model according to propositions in Section 4.2, including *mean*, *the number*, and *standard deviation* of the buyer’s ratings, *the rating different with other users* and *the variance, rating difference with other users with regard to commonly rated sellers* and *the variance, rating difference with the benchmark ratings and the variance, and skewness of the buyer’s ratings*. Figure 2(c) demonstrates the performance of our method as changing the number of features. The x-axis represents the number of features used in the implementation. Given a specific number of features, the line shows the average performance of different feature combinations, and the error bars demonstrate the respective best and worst performance. As illustrated, the overall performance of our method increases as more features are considered, especially when the number of features reaches 7. It partly indicates the reasonability of our propositions and effectiveness of the features identified in Section 4.2.

We further explore the effectiveness of our method by showing the performance of *our method without dishonesty clustering* (using DENCLU to cluster dishonest buyers in the first layer), and *our method without fuzzy smoothing*. The results are shown in Figure 3(a). As can be seen, we can conclude that both the fuzzy smoothing and the dishonesty clustering can contribute to the performance improvement of our method. This also represents that, some buyers (advisors) might be dishonest, but we still could extract valuable information from their opinions.

**Model Comparison.** Figures 3(b) and 3(c) show the performance comparisons between our approach and the other three approaches in both the basic and deceptive environments. For our approach, we set  $\sigma_1 = 0.12$  and  $\sigma_2 = 0.3$ . From the results shown in Figure 3(b), we can see that our method performs consistently the best no matter whether buyers have more or few interactions with sellers. HABIT performs better than TRAVOS, and both HABIT and TRAVOS perform much better than the baseline approach. Note that the marginal effect of new interactions on the performance of our method is very significant when there are only a few interactions for each buyer.

Based on the basic environment (number of iterations=100), we also examine the effect of deception as buyers lie about their past experience by conducting complementary lying behavior where if a true rating to a seller is  $r$  in the scale of  $[0,1]$ , the liar will modify the rating as  $1 - r$ . We vary the ratio of liars from 0 to 0.5, and plot the MAE results of different approaches in Figure 3(c). Our approach performs much better than the other approaches. Both our method and HABIT are not so much affected by lying buyers. This is mainly because in our simulations, the rating behavior of dishonest buyers is relatively static, and thus their ratings could still be used for our method. HABIT could also address this case as public information of advisors could be adopted to help infer the properties of the dishonest buyers.

## 5.2 Real Environments

In this section, we validate the effectiveness of our approach in centralized real-world environments by comparing our approach with the state-of-the-art recommendation algorithms in rating prediction. We conduct these experiments using real datasets to compare our model with several competing models, including TidalTrust [9], clustering-based recommender system (CoClustering) [8], probabilistic matrix factorization [16], and SocialMF method [13]<sup>5</sup>.

### 5.2.1 Experimental Settings

Three real-world datasets are used in the experiments, i.e. Epinions, FilmTrust and Flixster. Users provide numerical ratings in the range of  $[1, 5]$  on Epinions. Besides, users can also explicitly specify other users as trustworthy or not based on whether the ratings of others are consistently valuable or useless for the user. We adopt the extended Epinions data set<sup>6</sup> where trust value is labeled as 1. We sample a subset by randomly selecting 5,000 users. The other two datasets are FilmTrust and Flixster<sup>7</sup> where users can also indicate others as trustworthy, and provide item ratings ranging from 0.5 to 4.0 (5.0 in Flixster) with step 0.5. The statistics of the three datasets is given in Table 2.

Table 2: The statistics of three datasets

Features	Epinions	Flixster	FilmTrust
users	5,000	5,000	1,508
items	376,458	13,527	2,071
trust	744	2,898	2,853
ratings	968,467	264,540	70,998
avg rating	4.6964	3.6560	3.0028

The experiments are conducted by applying the leave-one-out technique, that is, each rating is iteratively hidden whose value will be predicted by applying our method, the TidalTrust, CoClustering, PMF, or SocialMF methods until all ratings in the data sets are tested. The performance is evaluated by two commonly used measures: the root mean square errors (RMSE) and mean absolute errors (MAE). They both refer to the differences between the predictions and the ground truth, but differ from each other as indicated by their names. Generally, smaller RMSE and MAE values indicate better predictive accuracy.

### 5.2.2 Results and Discussion

Table 1 summarizes the performance comparisons between our approach and other approaches on three real datasets. As shown in Table 1, our approach achieves better performance than others in terms of both RMSE and MAE on

<sup>5</sup>We use the source codes of the latter three methods provided by MyMediaLite Recommender System Library ([www.mymedialite.net](http://www.mymedialite.net)), and adopt the corresponding settings suggested by the papers.

<sup>6</sup>[www.trustlet.org/wiki/Epinions\\_datasets](http://www.trustlet.org/wiki/Epinions_datasets)

<sup>7</sup>[www.cs.sfu.ca/~sja25/personal/datasets](http://www.cs.sfu.ca/~sja25/personal/datasets)



Epinions and Flixster datasets, and a little worse but still comparably competitive to other approaches on FilmTrust dataset. This validates the effectiveness of our approach with respect to rating prediction. The lack of good performance of our approach on FilmTrust dataset is possibly because: 1) each item in FilmTrust has a great deal of interactions with users (around 34), which is suitable for the implementation of other approaches; and 2) users are quite controversial (see Table 2). Hence, a considerable amount of users is considered as *misguidance dishonest ones* and filtered out by our method. The performance of SocialMF and TidalTrust is much worse than PMF and CoClustering, which may be due to: 1) there exists noisy trust relationship on the three datasets; and 2) trust neighbors of a user might not share the same preference of the user.

## 6. CONCLUSIONS AND FUTURE WORK

This paper proposes a two-layered clustering approach to address the advisors' subjectivity difference and dishonesty problem in providing opinions. Specifically, the agent of each user firstly clusters the advisors of its users into different groups, with respect to their rating behavior. And then, each advisor is assigned to two groups with respective membership degrees. Lastly, each agent adopts an alignment algorithm to help its user align advisors' ratings to the ones of her own. We conduct experiments on both a simulated distributed environment and real data (considered as centralized environments) collected from Epinions, Flixster and FilmTrust, respectively. Experimental results verify that: 1) our approach can better help users utilize ratings (opinions) provided by advisors, and is relatively robust to deceptive environments; 2) the identified features are validated to be effective for our research scenario, and each part of our approach contributes to performance improvement; and 3) our approach can achieve competitive performance in rating prediction with respect to other recommender systems.

For the future work, we will conduct experiments to further validate the robustness of our approach by evaluating its effectiveness on resisting different kinds of attacks such as whitewashing and Sybil attacks.

## 7. ACKNOWLEDGEMENTS

This work has been made possible thank to the Institute for Media Innovation at Nanyang Technological University who has given a scholarship to the first author, and supported by the MOE AcRF Tier 2 Grant M4020110.020 awarded to Dr. Jie Zhang.

## 8. REFERENCES

- [1] P. Biyani, S. Bhatia, C. Caragea, and P. Mitra. Thread specific features are helpful for identifying subjectivity orientation of online forum threads. In *Proceeding of the 24th COLING*, pages 295–310, 2012.
- [2] J. P. Blair, T. R. Levine, and A. S. Shaw. Content in context improves deception detection accuracy. *Human Communication Research*, 36(3):423–442, 2010.
- [3] D. P. Doane and L. E. Seward. Measuring skewness: a forgotten statistic. *Journal of Statistics Education*, 19(2):1–18, 2011.
- [4] H. Fang, Y. Bao, and J. Zhang. Misleading opinions provided by advisors: Dishonesty or subjectivity. In *Proceeding of the 23rd IJCAI*, 2013.
- [5] H. Fang, J. Zhang, M. Sensoy, and N. Thalmann. SARC: subjectivity alignment for reputation computation. In *Proceedings of the 11st AAMAS*, pages 1365–1366, 2012.
- [6] Q. Feng, L. Liu, and Y. Dai. Vulnerabilities and countermeasures in context-aware social rating services. *ACM Transactions on Internet Technology (TOIT)*, 11(3):11, 2012.
- [7] F. Garcin, L. Xia, and B. Faltings. How aggregators influence human rater behavior? 2013.
- [8] T. George and S. Merugu. A scalable collaborative filtering framework based on co-clustering. In *Proceedings of the 5th ICDM*, pages 4–pp, 2005.
- [9] J. A. Golbeck. Computing and applying trust in web-based social networks. 2005.
- [10] G. Guo, J. Zhang, and D. Thalmann. A simple but effective method to incorporate trusted neighbors in recommender systems. In *Proceedings of UMAP*, pages 114–125. 2012.
- [11] Y. Haghpanah and M. desJardins. PRep: a probabilistic reputation model for biased societies. In *Proceedings of the 11th AAMAS*, pages 315–322, 2012.
- [12] A. Hinneburg and D. A. Keim. An efficient approach to clustering in large multimedia databases with noise. In *Proceedings of the 4th SIGKDD*, pages 58–65, 1998.
- [13] M. Jamali and M. Ester. A matrix factorization technique with trust propagation for recommendation in social networks. In *Proceedings of the 4th ACM RecSys*, pages 135–142, 2010.
- [14] A. Jøsang, W. Quattrociocchi, and D. Karabeg. Taste and trust. In *Trust Management*, pages 312–322. 2011.
- [15] T. R. Levine, R. K. Kim, and L. M. Hamel. People lie for a reason: Three experiments documenting the principle of veracity. *Communication Research Reports*, 27(4):271–285, 2010.
- [16] A. Mnih and R. Salakhutdinov. Probabilistic matrix factorization. In *Proceeding of the 21st NIPS*, pages 1257–1264, 2007.
- [17] Z. Noorian, S. Marsh, and M. Fleming. Multi-layered cognitive filtering by behavioural modelling. In *Proceedings of the 10th AAMAS*, pages 871–878, 2011.
- [18] K. Regan, P. Poupart, and R. Cohen. Bayesian reputation modeling in E-marketplaces sensitive to subjectivity, deception and change. In *Proceedings of the 21st AAAI*, pages 1206–1212, 2006.
- [19] Y. I. Sheline, M. Sanghavi, M. A. Mintun, and M. H. Gado. Depression duration but not age predicts hippocampal volume loss in medically healthy women with recurrent major depression. *The Journal of Neuroscience*, 19(12):5034–5043, 1999.
- [20] W. T. L. Teacy, M. Luck, A. Rogers, and N. R. Jennings. An efficient and versatile approach to trust and reputation using hierarchical bayesian modelling. *Artificial Intelligence*, 193:149–185, 2012.
- [21] W. T. L. Teacy, J. Patel, N. R. Jennings, and M. Luck. TRAVOS: Trust and reputation in the context of inaccurate information sources. *Autonomous Agents and Multi-Agent Systems*, 12(2):183–198, 2006.
- [22] J. Zhang and R. Cohen. A comprehensive approach for sharing semantic web trust ratings. *Computational Intelligence*, 23(3):302–319, 2007.