Forming Opinions via Trusted Friends: Time-Evolving Rating Prediction Using Fluid Dynamics

Wenjun Jiang, Jie Wu, *Fellow, IEEE*, Guojun Wang, *Member, IEEE*, and Huanyang Zheng

Abstract—Trust-based recommendation systems study how people form opinions via trusted friends, so as to predict unknown ratings based on the ratings expressed by trusted friends. Most of the existing work only considers the ratings at the current time slot. In real life, a user's opinion evolves over time, since he receives the influence of different opinions sequentially. In addition, existing work usually targets a single user at a time; there is a need to predict multiple ratings for multiple connected users. To reach these ends, we propose a novel multiple-rating prediction scheme, *FluidRating*, which uses fluid dynamics theory to reveal the *time-evolving* formulation process of human opinions. In this scheme, each user corresponds to a container, and several containers are connected through single directional pipes, corresponding to influence relations. We identify three features of human personality in the opinion formulation and propagation process: "persistency" represents how much one insists on his opinion, "persuasiveness" represents the ability to impact others, and "forgetting" reflects the common truth that people have limited memory. The recommendation (or influence) is modeled as fluid with two dimensions: its temperature is taken as the "opinion/rating," and its height is deemed as the persistency. When new opinions emerge, each person refines his opinion through a round of fluid exchange with neighbors. Opinions of multiple rounds are aggregated to gain a final prediction. Experimental evaluation in a real data set validates the feasibility and the effectiveness of the proposed model.

Index Terms—Fluid dynamics theory, personality feature, rating prediction, time-evolving, trust-based recommendation system

1 Introduction

High-quality and personalized recommendations are a key feature in many online systems [1], [2], [3], [4], [5], [6], [7]. Before recommending an item to a user, we need to predict if he likes it or not. Much work has been done for this end [8]. Among them, the trust-based recommendation system has gained much attention [1], [9], [10], [11], which explores trusted friends' opinions as the evidence of recommendation. We study trust-based recommendation in this paper. Besides trust relations, features of user personalities will be identified, to better model the opinion formulation process. We also emphasize the time-evolving feature of opinion formulation, using fluid dynamics theory.

We consider the setting in which there is a single item of interest (e.g., a product). A subset of users (*raters*, denoted as R) have prior opinions about this item. The remaining users, (*non-raters*, denoted as N) have not formed their opinions, and thus their ratings need to be predicted. In addition, we convert the trust relations between two users, say a and a', into the influence relations [12], which are based on the following intuition: the more a' trusts a, the higher the probability that a can influence a'. Then, all the users, $R \bigcup N$, and the influence relations among them, are used to construct the *rating network*. Fig. 1 shows an example of a rating network, where $R = \{a_1, a_2\}$, and $N = \{a_3, a_4, a_5, a_6\}$. The number associated with a node in R corresponds to a *rating*. The higher the number, the higher the rating. The number associated with each edge represents the *influence value* from one user to another, which is determined by the trust between them. We identify three key issues in users' opinion formulation, and we map them with three features of human personality, as follows.

The first key issue in the rating prediction is: *Upon receiving new opinions, how will a user refine his opinion*? We observe that, in real life, whether the user changes his opinion or not depends on how much he insists on his own opinion, as well as how much the others insist on theirs. We call this feature the *persistency*.

The above issue is suitable for the *direct influence* from connected friends. A natural successive question is *how much will the user pass the influence on to the next user*, which is the *indirect influence* from friends of friends. As in real life, it depends on the user's ability to convince others. We model this feature as the *persuasiveness*.

Moreover, people usually have a limited memory, indicating that the recent influence usually has the main effect. This also indicates that the former influence, which he received long ago, has been either forgotten or blended into his current opinion. To reflect this point, we abstract this

- *W. Jiang is with the School of Computer Science and Electronic Engineering, Hunan University, Changsha, Hunan Province 410082, P. R. China. E-mail: jiangwenjun@hnu.edu.cn.*
- *G. Wang is with the School of Information Science and Engineering, Central South University, Changsha, Hunan Province 410083, P. R. China. E-mail: csgjwang@gzhu.edu.cn.*
- *J. Wu and H. Zheng are with the Department of Computer and Information Sciences, Temple University 1805 N. Broad Street, Philadelphia, PA 19122. E-mail: {jiewu, huanyang.zheng}@temple.edu.*

Manuscript received 7 Mar. 2014; revised 1 June 2015; accepted 3 June 2015. Date of publication 11 June 2015; date of current version 17 Mar. 2016.
Recommended for acceptance by V. Eramo.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TC.2015.2444842

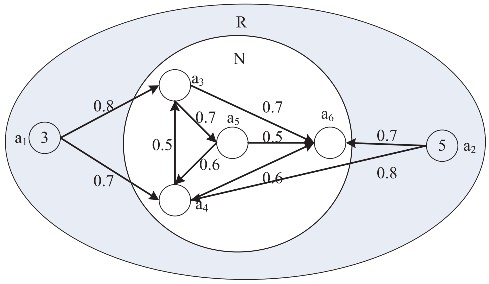


Fig. 1. An example of a rating network: nodes represent users; weighted edges represent influence relations; numbers on nodes represent ratings on a given target item.

feature as the *forgetting*, which is a necessary mental activity of human beings [13].

1.1 Motivation

Several models have been proposed to aggregate the trust information among trusted friends [2], such as TidalTrust [14], MoleTrust [15], FlowTrust [16], and RN-Trust [17]. These models work in one round, i.e., only the *current* trust information is considered. However, in real life, a user's opinion evolves with time. This is because he receives the influence of different opinions at different times, either directly from his friends, or indirectly from friends of his friends. Therefore, going one step further, we propose a rating prediction scheme, *FluidRating*, to simulate the *time-evolving* opinion formulation process as fluid flows, using fluid dynamics theory.

In addition, most of the existing work targets a single user at a time. Taking a different view, we consider a multiple-user marketing scenario: service providers or marketers desire to know the opinions/ratings of multiple users in a group, particularly those who are the ones that think more highly of the target item.

1.2 Main Ideas

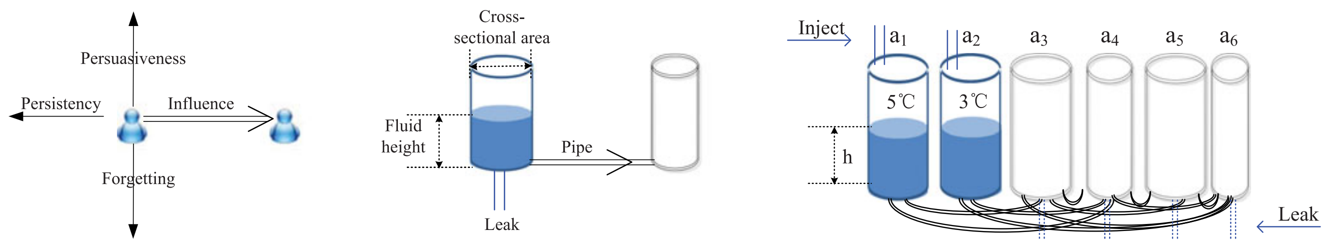
In FluidRating, a rating network is modeled as a fluid dynamics system: each node corresponds to a *container* with enough volume. Each influence edge corresponds to a single-direction *pipe* connecting two containers; Pipes are installed at the bottom of the containers. The recommendation (or opinion influence) from friends is captured as *fluid*, which has two dimensions: the *temperature* is taken as the "opinion/rating," and its *height* is deemed as the "persistency." The cross-sectional area of a container is used to reflect the "persuasiveness," where a larger area indicates less persuasiveness. In addition, each container has a small plug at its bottom, through which fluid will leak a little, reflecting the *forgetting* feature. Fig. 2 shows the mapping from features to the resulting fluid dynamics system.

In FluidRating, fluids originate from raters and pass through non-raters. When there exists a fluid height difference between two connecting containers, fluid will flow from one container to another, according to fluid dynamics theory. When fluid flows, the fluids flowing into a container will mix with the existing fluids. Then, the fluid temperature and volume in this container will change, reflecting the time-evolving properties of opinion formulation. Eventually, each container will have some fluid. We adopt a discretized approach to computing the temperature change of each container over the *round*. The final predicted rating of a non-rater is a collection of sampled temperatures.

1.3 Contributions

Our contributions are threefold: (1) We present a clean-slate computational model, FluidRating, based on fluid dynamics theory that can capture many subtle properties in a time-evolving recommendation system. Instead of only considering a static influence at the current time slot, the model takes opinion refinements collected over time into consideration; this clearly reveals the time-evolving formulation process of human opinions. In addition, our model provides two-dimensional information as the final prediction: the fluid temperature (i.e., predicted rating) and volume. The latter can be deemed as the confidence of the predicted rating. The larger the volume is, the more difficult it is to change the corresponding rating. (2) We identify three features of human personality, i.e., the persistency, the persuasiveness, and the forgetting. We also differentiate direct influence from directly connected friends and indirect influence from friends of friends. Both features and influence values are captured through a simple system consisting of containers and pipes only. Moreover, multiple ratings of users in a group can be predicted simultaneously. (3) We conduct extensive experiments in a real data set (Epinions), which validates the feasibility of the proposed model, and also demonstrates its effectiveness.

The remainder of this paper is organized as follows: Section 2 surveys related work. Section 3 formulates the problem. Sections 4 and 5 present the overview of the model and the algorithm details, respectively. Section 6 analyzes the properties of FluidRating. Section 7 describes the experimental evaluation. Section 8 concludes this paper and suggests future work.



(a) Features and influence (b) The mapping (c) The fluid system

Fig. 2. (a) shows three features of persistency, persuasiveness, and forgetting, in opinion formulation and propagation, (b) shows three corresponding mappings of fluid height, cross-sectional area of a container, and leak; in addition, influence relation is modeled as a pipe, (c) shows a FluidRating system for the rating network in Fig. 1.

2 Related Work

Based on how recommendations are made, recommendation systems are classified into three categories [18]: the content-based recommendation, where the user will be recommended items similar to the ones the user has preferred in the past; the collaborative recommendation [19], [20], where the user will be recommended items that people with similar preferences have liked in the past; and the hybrid of the above two. Neighborhood-based recommendation [21] is very popular for its simplicity, efficiency, and high quality recommendation. Our work focuses on trust-based recommendation, which can be taken as a collaborative and also neighborhood-based approach, since trust relations are created based on past similarities. In this section, we briefly review the literature of trust models, user opinions, and social influence that can be used in recommendation.

2.1 Trust Model

Trusted graphs [22] usually are directed acyclic graphs (DAGs), where trust calculation is simplified as node/edge-disjointed paths [23], serial-parallel graphs [24], shortest and strongest paths [14], or paths within a restricted depth [15]. RN-Trust [17] adopts circuit theory to model the trust process. FlowTrust [16] considers flow as a trust measure, where the overall trust is calculated using the network flow theory. All of the above models calculate trust values in one round at the current time slot. Personalized PageRank [1], [25] is time-evolving, based on the Markov chain model. FluidRating uses fluid dynamics theory to relate time-evolving temperature changes to opinions/rating refinements. It is more general than other trust models, in that it can be incorporated as the first step to construct the subgraph, based on which the fluid dynamics system can be set up.

2.2 User Opinion/Rating

User opinion is usually represented as a numeric value on an online web site. Anderson et al. [1] uses a finite integer set with {+, -, 0} representing positive, negative, and no (neutral) ratings. In FluidRating, opinion is measured by fluid temperature, which can easily be updated based on the volume and temperature of the new fluid. In addition, people can be associated with both an innate opinion and an expressed opinion [26] for a given topic. The former is formed independent of social interactions, while the latter could be shaped by others [27]. In FluidRating, the two kinds of opinions can be easily treated with the initial and mixed fluid.

2.3 Social Influence

Zhu et al. [28] find that a person's opinion is significantly swayed by others' opinions. Bakshy et al. [29] validate that stronger ties are individually more influential, while weak ties are responsible for the propagation of novel information. Ma et al. [30] proposed using heat diffusion to simulate influence propagation, and they aimed to solve the influence maximization problem [31]. The proposed FluidRating scheme takes the finding in [28] as a foundation: when new opinions come, each person will refine his opinion (through a round of fluid exchange with neighbors).

TABLE 1
Notations

Symbol	Description
$G=(V,E)$	rating network
R	rater set $\{a_1, a_2, ..., a_m\}$
N	non-rater set $\{a_{m+1}, ..., a_{n-1}, a_n\}$
$V = R \bigcup N$	node set with a total of n nodes
$a/a'/a''$	node a, outgoing/incoming neighbor of a
$N_a/N_a^{out}/N_a^{in}$	whole/outgoing/incoming neighbor set of a
$e_{aa'}/w_{aa'}/v_{aa'}$	edge/weight/flow velocity
i/Δ	sample index/sample interval
$h_a(i)/s_a(i)/t_a(i)$	height/volume/temperature in a
$s_{aa'}(i)/t_{aa'}(i)$	volume/temperature from a to a'
b	cross-sectional area of a container
l	the leak
θ	the initial maximum fluid height of non-raters

3 Problem Formulation

We first describe the settings of a trust-based recommendation system. Then, we formulate the problem. Notations used in this paper are described in Table 1. We first define the rating network as:

Definition 1. *A rating network is a directed graph $G=(V,E)$, where V is a set of nodes, and $E \subseteq V^2$ is a set of directed edges. Each edge $e_{aa'}$ has the direction from node a to a', associated with a weight $w_{aa'}$ indicating the influence value from a to a'.*

The node set $V=\{a_1, a_2, ..., a_m, a_{m+1}, ..., a_{n-1}, a_n\}$ consists of two types of nodes: *rater*s who have formed their opinions, $R=\{a_1, a_2, ..., a_m\}$, and *non-rater*s who have not formed their opinions, $N=\{a_{m+1}, ..., a_{n-1}, a_n\}$.

Two types of users take different roles: raters (R) have formed their opinions, and thus serve as the source of opinions. Non-raters (N) connect raters and other non-raters, and take the roles of receiving and propagating influence, i.e., they will be influenced by some raters, and then, will propagate the influence to other non-raters.

In addition, we have the following observations: (1) The opinions of non-raters evolve with time, while those of raters do not. (2) Each non-rater can refer to multiple opinions from friends (raters or non-raters), and form his own opinion. (3) All the influence from raters, or non-raters, to others are independent.

Next, we formally define three features of human personality in the recommendation system, as follows:

Definition 2. *The* persistency *of a user's opinion represents the force or the degree to which one insists on his own opinion.*

Definition 3. *The* persuasiveness *of a user represents the force or the degree to which one can convince others to accept his opinion.*

Definition 4. *The* forgetting *of a user represents the degree/proportion with which one forgets (or gives up) his former opinions, in order to accept new opinions.*

Here, the influence is directional from the user with higher persistency to the user with lower persistency. For two connected users a and a', a can only influence a' if a has a higher persistency.

In fact, the persistency and the persuasiveness are not independent. Generally speaking, people who are more persuasive are more persistent in regard to their own opinion. For two users a and a', suppose the persuasiveness of a is larger than that of a'. Then, receiving the same amount of influence, a should be able to generate a larger persistency than a'.

Based on this, we define the problem as follows:

Time-evolving rating prediction problem. Given a rating network $G = (V, E)$ with $V = R \bigcup N$, R is the set of raters and N is the set of non-raters whose ratings on a target item need to be predicted. The task is to design a scheme to predict the ratings of non-raters in N efficiently, and to capture the time-evolving opinion formulation process, as well as to refine opinions using the features of users, such as the persistency, the persuasiveness, and the forgetting.

4 Overview of the Model

We first describe the basic social and physical principles that our model should obey. Then, we describe the overview of FluidRating, which uses a novel approach of applying fluid dynamics theory in trust-based recommendation systems.

4.1 Basic Social Principles

Since we are focusing on how the current user will refine and form his opinion in a recommendation system, we consider the scenarios in our real life experience. In most of the rating-based system, users can register and express their own personal opinions on products. These reviews will influence future customers' impressions on the item being discussed. The first rating a user reads will give him the first impression of the related item, which we call *the first influence*. This mainly depends on the way that the system displays the ratings. In real examples including eBay (eBay.com), Amazon (Amazon.com), and Taobao (Taobao.com), the default setting is that most new reviews are displayed first. Users can also manually choose to display the most helpful reviews first. In Epinions (Epinions.com), every member maintains a "web of trust", which consists of other members who are trusted, and by which the Epinions Website displays reviews by trusted friends first. Our work is considering a system such as Epinions.

In addition, some users insist more on their opinions. It is natural that people takes those opinions more seriously, which we call *stronger influence*. We extract the following two ground truths from real life. They serve as general rules for the model design:

Principle 1 (First Influence Dominates). *The first influence makes more of an impact on a user's opinion.*

Principle 2 (Stronger Influence Dominates). *The recent influence makes more of an impact on a user's opinion.*

Principle 1 is also called the *first impressions* [32] phenomenon in psychology. As mentioned in a proverb, "first impressions are lasting impressions." In our container-pipe setting, early fluids in a container will have a higher likelihood of determining the final rating than will later ones. This is because the fluid from neighboring containers will not flow into the current container unless their fluid levels are higher (i.e., more certain). For Principle 2, people usually have a limited amount of storage for memory, indicating that the recent influence usually has more of an impact, with a possible exception of first impressions. To reflect this point, we abstract this feature as *forgetting*.

Principle 1 will be used to guide the selection of aggregation sequence. Principle 2 will be used to model the refinement of opinion.

4.2 Basic Physical Principles

Given a FluidRating system, fluid will flow from raters to non-raters in N. There are two basic physical principles that FluidRating should obey as a closed system:

Principle 3 (Mass Conservation [33]). *The mass of an closed system must remain constant over time.*

Principle 4 (Energy Conservation [33]). *The total energy of an isolated system cannot change over time.*

FluidRating takes a single type of fluid which has a constant density. Hence, fluid volume is proportional to the mass. Principle 3 will be used for maintaining the fluid volume conservation when we conduct fluid updates. Principle 4 will be applied to calculate the updated temperature when fluids are mixed.

4.3 Model the Recommendation

We view the time-evolving formulation process of human opinions as follows: each user first receives the influence from directly connected friends, and updates his own opinion accordingly; he then propagates his opinion to other friends. In this way, for each user, the first influence he receives is the direct influence, while the later ones are mixtures of direct and indirect influences. The process can be done iteratively.

4.3.1 The Model

A rating network is modeled as a fluid dynamics system (Algorithm 1): each user corresponds to a container with enough volume so that fluid will never overflow. Containers are connected through single-direction pipes, which correspond to the influence edges in the rating network. Recommendation is modeled as fluids, which originate from raters, and pass through non-raters. The ratings of users are modeled as the fluid temperature, and the persistency of the opinion of the corresponding user is measured as the fluid height. The persuasiveness of a user is reflected by the cross-sectional area b. The total amount of influence (or recommendation) is the volume of the fluid.

Algorithm 1. Initialization(G, N)

Input: G, a rating network.
Output: G', a FluidRating system for N.
For each rater/non-rater, set up a container with enough volume, so that fluid will not overflow.
for each rater in G **do**
  Set the fluid temperature in its container to be equal to its rating, and height be h.
for each non-rater in G **do**
  Randomly set the fluid temperature in its container to fall in the range of $[1, 5]$, and height in $[0, \theta]$.
for each influence edge from a to a' in G **do**
  Set up a single-direction pipe from a to a' in G'.

Both the direct and indirect influences are modeled through fluid exchanges among connecting containers. When there exist fluid height differences, we can obtain the speed of efflux by using basic fluid dynamics theory, and thus, can cope with the fluid updating.

We adopt a discretized approach to computing the temperature change over the round (or time slot), with each slot having a duration of Δ. A total of k samples of the fluid temperatures in a non-rater's container are collected, and aggregated to get the final temperature (i.e., the ultimate opinion). Then, the process of time-evolving rating prediction is converted into the fluid temperature and volume updating through multiple rounds.

4.3.2 Assumptions

We make the following assumptions: (1) We assume the whole system to be a closed one. Besides the containers and pipes converted from a rating network, it has the outer part, consisting of very large containers that have enough fluid to supply each rater's container, respectively; and an additional large container that can hold all the fluid leaks from each rater or non-rater's container. (2) We assume that the fluid temperature in each container will not change until there is some incoming fluid. That is, the container, the pipes, the one-way valve, and the plug at the bottom are associated with temperature insulating material. In addition, we neglect the effect of natural cooling. In this way, connecting containers can keep their own fluid temperatures. (3) Similar to [1], we assume that raters' ratings and persistency will not change. The insight behind this phenomenon is that, when a user has enough first-hand experience, he will not listen to other second-hand opinions. (4) We assume that the leakage by the plug is relatively small compared to all of the fluid in a container. That is, there will always be some fluid in a container (it will never be completely empty). (5) To predict the rating and provide a proper recommendation for a given user, it is almost impossible to take all nodes into consideration. Therefore, a subgraph is expected to improve the efficiency. It can be constructed as follows: beginning at the target group of non-raters, add in neighbors who are raters or who can reach raters within a given maximum length; Next, add in all edges among the added nodes. Without loss of generality, we assume that there is already a subgraph, based on which, the FluidRating system can be set up. (6) We assume the rating falls in the range of [1, 5].

4.4 FluidRating System Setup

The FluidRating system consists of three parts: the containers, the pipes between containers, and the fluid flowing among the containers and pipes. Algorithm 1 shows the initialization process. Fig. 2c illustrates an example of the FluidRating system corresponding to the rating network in Fig. 1.

The containers. We relate each node to a container with enough volume (i.e., large enough so that fluid can never overflow). Moreover, all containers are put on the same level, to make sure that the fluid in all containers have the same atmospheric pressure on the surface. The cross-sectional area b essentially reflects the persuasiveness of the corresponding node, i.e., how effectively he can persuade neighbors.

The pipes. Each edge $e_{aa'}$ is related to a directional pipe from the container of a to a', the cross-sectional area of which is equal to $w_{aa'}$, and the direction of which is consistent with the influence relations. The directional pipe is implemented through installing a one-way valve. Again, it is worth noting that, the one-way valve is associated with temperature-insulating material, so that fluid temperature is isolated in each container. All the pipes are installed at the bottom of connecting containers. Based on Principle 2, stronger influence dominates; only when a container has a larger fluid height (indicating larger persistency), will the fluid flow to its neighbors and mix with their fluids.

The fluid. We assume that there is a single type of fluid in FluidRating. As shown in line 2 in Algorithm 1, we maintain the fluids in the container of each rater to be a height of h by injecting fluid continuously, indicating that their persistency is stable. Their ratings are initialized as the fluid temperatures.

As for the non-raters, although they have not expressed their ideas, due to the external influence [34] or the intuition, they may have some innate opinions on the target [27]. However, how to get the initial opinion is out of the scope of this paper. [27] provides a method to induce the innate opinion through debiasing the expressed opinion. Here, we set the initial fluid height of the non-raters randomly in the range of $[0, \theta]$, with $\theta <= h$; meanwhile, the initial fluid temperature is in the range of $[1, 5]$.

In Algorithm 1, each rater and non-rater are considered once (lines 1-5), with the time complexity of being $O(|V|)$. Each edge is transformed into a pipe, with time complexity $O(|E|)$. Therefore, the final time complexity of Algorithm 1 is $O(|V| + |E|)$.

5 FluidRating: Algorithm Details

In this section, we introduce the details of FluidRating, which consist of three steps (Algorithm 2): fluid updating preparation, fluid updating execution, and sample aggregation. First, we present how the fluid will flow and how the flowed fluids will mix up, according to fluid dynamics theory. Then, sampled temperatures of multiple rounds are aggregated to obtain a final opinion.

We show how fluid flows among containers and pipes, from the view of a *discrete time system*. Without loss of generality, we consider that the fluid updating is done synchronously at the end of each time slot.

Algorithm 2. FluidRating(G', N)

Input: G', a FluidRating system.
Output: $t_{a_{m+1}}, \ldots, t_{a_n}$, temperature of non-raters in N.

```
Let k be the total number of samples (time slots).
for i = 0 to k do
    for each pipe from a to a' do
        if h_a(i) > h_a'(i) then
            Record the volume/temperature of the flowed fluid
            (Eqs. (1) and (2)).
    for each rater's container do
        Fill fluid to maintain its height/temperature.
    for each non-rater's container do
        Update fluid height/temperature (Eqs. (6) and (7)).
    Record fluid temperatures in non-raters' containers.
Aggregate the fluid samples (Section. 5.3).
```

5.1 Fluid Updating Preparation

First, let us consider a single pipe, say the pipe connecting a and a', with the cross-sectional area $w_{aa'}$. At the beginning of the ith time slot, if fluid height in a is higher than that in a' (i.e., $h_a(i) > h_{a'}(i)$), then the fluid will flow from a to a' with a duration of Δ. The basic theory here is Torricelli's law [33]. It states that the speed of efflux, v, of a fluid through a sharp-edged plug at the bottom of a tank filled to a depth h is the same as the speed that a drop of fluid would acquire in falling freely from a height h, i.e., $v = \sqrt{2gh}$, where g is the gravitational acceleration. As an application of this law, the speed of the fluid flows in our case will be $v_{aa'} = \sqrt{2g(h_a - h_{a'})}$.

5.1.1 The Updating Volume

Considering the cross-sectional area $w_{aa'}$ and the duration of time slot Δ, the volume of flowed fluid in ith Δ can be calculated as:

$$s_{aa'} = v_{aa'} \cdot w_{aa'} \cdot \Delta = \sqrt{2g[h_a(i) - h_{a'}(i)]} \cdot w_{aa'} \cdot \Delta, \quad (1)$$

where Δ is set to be small enough so that $s_{aa'} < s_a$. That is, the amount of the outgoing fluid cannot be larger than the total fluid in a container. It is worth noting that, according to the basic physical theory, when only considering the two connected containers a and a', the fluid height relation will not be reversed, as follows.

Lemma 1. *Suppose a physical system has only two connecting containers a and a' (with the same surface pressures). If the fluid heights at the beginning meet the condition of $h_a > h_{a'}$, then, after any time, it would not happen that $h_a < h_{a'}$.*

The insight behind Eq. (1) is that, the influence received by a person from a friend is proportional to the square root of their persistency difference, the influence value from this friend to him, and the time length. Moreover, in FluidRating, as time passes, some fluid flows from a a to a'. Then, the height difference will become smaller; this leads to the decrease of fluid speed. The process is very similar to the influence process in real life: at the beginning, two friends may be very different from one another; they influence each other with interactions; then, their differences become fewer.

5.1.2 The Updating Temperature

As for the temperature of the flowed fluid from a to a', we consider it to be the same as that of a, as follows:

$$t_{aa'}(i) = t_a(i). \quad (2)$$

In addition, we define the uniform *leakage proportion* to reflect the forgetting feature, as in the following:

$$l(i) = l, l \in [0, 1]. \quad (3)$$

That is, at the end of each time slot, fluid in a container will leak proportionally to its volume.

5.2 Fluid Updating Execution

In this section, we describe how the flowed fluids mix up with the remaining fluid in the containers.

5.2.1 The Updated Volume

According to the law of mass conservation, the fluid in the amount of $s_{aa'}$, will flow out from a, and flow into a'. For a given container a, at the end of the ith time slot, the volume of fluid in a, before it leaks (denoted as $\tilde{s}_a(i+1)$), will be:

$$\tilde{s}_a(i+1) = s_a(i) - \sum_{a' \in N_a^{out}} s_{aa'} + \sum_{a'' \in N_a^{in}} s_{a''a}, \quad (4)$$

where N_a^{out} and N_a^{in} represent the outgoing and incoming containers of a, respectively.

To reveal the forgetting feature, we let the fluid leak a little. Then, the final volume will be:

$$s_a(i+1) = \tilde{s}_a(i+1) \cdot (1 - l), \quad (5)$$

where l is the leak proportion.

5.2.2 The Updated Height

Since the height (representing the persistency) will impact whether fluid will flow, we calculate it as follows:

$$h_a(i+1) = s_a(i+1)/b, \quad (6)$$

where b is the cross-sectional area of a container a, and it represents the persuasiveness of the corresponding node.

5.2.3 The Updated Temperature

Since we use a single type of fluid, then the specific heats are the same and can be overlooked. According to Principle 4, the law of energy conservation, the fluid temperature after mixing up is:

$$t_a(i+1) = \frac{t_a(i) \cdot \left[s_a(i) - \sum_{a' \in N_a^{out}} s_{aa'}\right] + \sum_{a'' \in N_a^{in}} \left[t_{a''a} \cdot s_{a''a}\right]}{\tilde{s}_a(i+1)}, \quad (7)$$

which is essentially $\sum(\text{volume} \cdot \text{temperature}) / \sum \text{volume}$. The first part of the numerator is the remaining fluid in container a, while the second part is the fluid that has flowed from other containers.

For the fluid system in Fig. 2c, part of its fluid updating process is shown in Fig. 3 ($k = 100, \Delta = 0.04$). The known conditions are: $h_{a_3} = h_{a_4} = h$, $t_{a_3} = 5$ and $t_{a_4} = 3$. The trends of fluid heights and temperatures are shown in Fig. 4.

5.3 Sample Aggregation

We aggregate the fluid temperature of a non-rater in different time slots to gain a final temperature (i.e., a final opinion or rating). We design four aggregation methods and compare their effects in experiments. It is worth noting that, all the methods consider the time-evolving effects, since a new round of fluid exchange is based on the results of all past rounds.

5.3.1 FluidRating I

Uniform aggregation. That is, each sample is given the same weight, i.e., $t_{a_n} = \sum_{i=1}^{k} q \cdot t_{a_n}(i)$, where q is the weight for each sample, and $\sum_{i=1}^{k} q = 1$. Then, the aggregation sequence is $\{q, q, q, ...\}$.

5.3.2 FluidRating II

Descending aggregation. That is, the earlier samples are given more weight, and vice versa. An example of descending

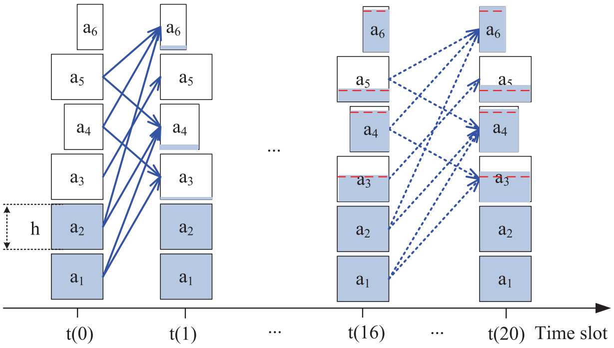
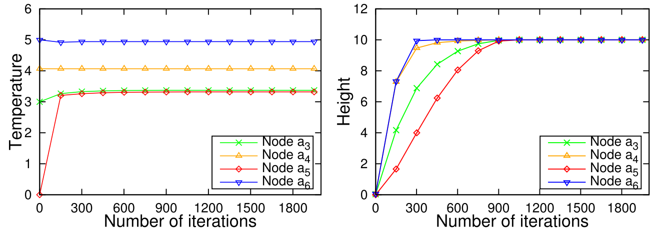


Fig. 3. The illustration of fluid updating from the 0th to 1th Δ; and fluid heights change from 16th to 20th Δ.



(a) Temperature (b) Height

Fig. 4. The fluid temperature and height for the example scenario in Fig. 1, $k = 2{,}000, \Delta = 0.003, h = 10$, b equals to 0.75 for a_3, 0.5 for a_6), and 1 for others.

aggregation can be: $t_{a_n} = \sum_{i=1}^{k} q^i \cdot t_{a_n}(i)$, where q^i is the weight for the $i\text{th}$ sample, and $\sum_{i=1}^{k} q^i = 1$. Then, the aggregation sequence is $\{q, q^2, q^3, \ldots\}$.

5.3.3 FluidRating III

Ascending aggregation. That is, the later samples are given more weight, and vice versa. An example of ascending aggregation can be: $t_{a_n} = \sum_{i=1}^{k} q^{k-i+1} \cdot t_{a_n}(i)$, where q^{k-i+1} is the weight for the $i\text{th}$ sample, and $\sum_{i=1}^{k} q^{k-i+1} = 1$. Then, the aggregation sequence is $\{q^k, q^{k-1}, \ldots, q^2, q\}$.

5.3.4 FluidRating IV

No aggregation. That is, only the last sample is taken as the final temperature.

The time complexity of Algorithm 2 can be calculated as follows: in a single time slot, each container and pipe is considered once, the time complexity of which is $O(|V| + |E|)$; there are a total of k time slots, so the final time complexity is $O(k|V| + k|E|)$. Over a time period, the updating of fluid temperature and volume in containers can be deemed as a state transition from one to the next. At a specific time slot, say $t(i)$, only one array is used to record the current fluid state in pipes; two arrays are needed to store the current state (temperature/height) and the next state of containers, respectively. The space cost is $O(|E|)$ for pipes, and $O(|V|)$ for containers. $n - m$ additional arrays are used for recording the k samples of fluid temperatures in $a_j \in N$ ($j \in [m+1, n]$), with space cost $O(k|N|)$. Therefore, the total space complexity is $O(|V| + |E| + k|N|)$.

6 The Analysis

In this section, we analyze the properties of FluidRating on two aspects: its convergence and its conformity with social and physical principles. We also analyze its explainability of recommendation and make deep discussions in the Appendix.

6.1 Convergence Analysis

We first summarize two properties on the fluid heights in FluidRating. Then we analyze the convergence.

Theorem 1. *In FluidRating, non-raters' fluid heights will not be larger than raters' fluid heights, h.*

Proof. We can proof this by contradiction. Suppose there exists a non-rater a, whose fluid height h_a is larger than h. There are two possible cases about the fluid in a.

Case 1: fluid in a comes directly from some rater b, which indicates that, in the beginning, $h < h_a$. In this case, according to Lemma 1, it will never happen that $h_a > h$. Therefore, it contradicts with the statement.

Case 2: fluid in a comes directly from some non-rater c (whose fluid height is h_c), which indicates that $h_c > h_a$. Since $h_a > h$, we have $h_c > h$. Then, we can study the same cases, from which c's fluid comes. Iteratively doing this eventually results in some non-rater whose fluid comes directly from a rater. Then, the case is converted into case 1, which has a contradiction. □

Theorem 2. *In FluidRating, after a sufficient time period, all non-raters' fluid heights will be equal to h.*

Proof. Suppose there exists a non-rater a, whose fluid height h_a is not equal to h. According to Theorem 1, it cannot be $h_a > h$. Then, it must be $h_a < h$. Again, there are two possible cases about the fluid in a.

Case 1: fluid in a comes directly from some rater b. In this case, because $h_a < h$, and there is a pipe from b to a, the fluid will flow from b to a, until $h_a = h$.

Case 2: fluid in a comes directly from some non-rater c (whose fluid height is h_c). If $h_c > h_a$, there will be flow from c to a until $h_c = h_a$. Then, we can study the same two cases from which c's fluid directly comes. Iteratively doing this will eventually result in some non-rater whose fluid directly comes from a rater. Then, this case is converted into case 1. □

As shown in Fig. 4, the height and temperature become stable after a certain time period. Moreover, after enough rounds, the fluid heights turn to be equal to $h = 10$, which is consistent with Theorems 1 and 2. We further test the result when considering the leakage function in Eq. (3), and the result is very similar.

The insight behind this phenomenon is the opinion formulation process. At the very beginning, a person has no idea of the given target item. Upon receiving opinions from others, he formulates and refines his own opinion. In addition, the opinion of a person becomes more and more mature, indicating increased persistency. Moreover, the opinion matures quickly at the beginning, and slows down later. The amount of increased persistency decays with time. The simulation result is consistent with our real-world experiences.

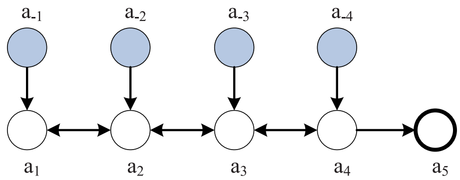


Fig. 5. A linear network.

6.2 Conformity with Basic Principles

The proposed FluidRating model is consistent with both the basic social principles and physics principles.

6.2.1 Conformity with Principle 1

We use a simple linear configuration of container-pipe (see Fig. 5) to show that FluidRating preserves Principle 1. In this configuration, each of four linearly connected non-raters (a_i, $i = 1, 2, 3, 4$) is connected to a distinct rater (a_{-i}). Sink a_5 is only connected to a_4. In the configuration of Fig. 5, we conducted an experiment to calculate the percentages of fluids from four raters in the sink. Each rater selects a fluid height randomly from [1], [10]. The higher the fluid level is, the more chances it has to reach the sink. The results from 10,000 random samples show the following percentages: 0.73 for a_{-4}, 0.22 for a_{-3}, 0.04 for a_{-2}, and 0.01 for a_{-1}, a clear indication of first impressions (i.e., the closer one gets to have more chances of reaching the sink). Table 2 shows the fluid heights and fluid source percentages from four raters after different iterations. A total of 100 rounds are tested for initiating rater heights of 2, 5, 9, and 4 for a_{-1}, a_{-2}, a_{-3}, and a_{-4}, respectively. The 100-round simulation shows the first impression for this case at a_5, where 0.62 is from a_{-4}, and 0.38 is from a_{-3} (even though a_{-3} has a larger height).

6.2.2 Conformity with Principle 2

In FluidRating, a larger height indicates stronger persistency. To update fluid in containers, we look at each pipe. For the pipe from a to a', according to Torricelli's law, only when $h_a > h_{a'}$ will fluid flow and mix. It is consistent with Principle 2, i.e., only when another's persistency is larger, will the current user take the advice and refine his own opinion.

6.2.3 Conformity with Principle 3

Since we consider a single type of fluid in FluidRating, the conservation of volume is equivalent to that of mass. The fluid volume in FluidRating remains unchanged, because for each step of updating, the fluid flowing in a pipe is always equal to the fluid flowing out of that pipe. Therefore, the total amount of fluid remains unchanged.

TABLE 3
The Coverage with Different Lengths

Length	1	2	3	4	5	6
Coverage(%)	62.49	78.84	85.74	89.76	96.4	100

6.2.4 Conformity with Principle 4

According to physics, the fluid energy is equivalent to the product of fluid mass, temperature, and specific heat. In FluidRating, with several parts of the fluid being mixed, the final temperature is calculated according to the energy before being mixed (Eq. (7)). Therefore, Principle 4 holds.

7 Experimental Evaluation

In this section, we evaluate the performance of FluidRating with experiments in a real social network data set.

7.1 Experimental Design

7.1.1 Data Set and Preprocess

As far as we know, Epinions is a good testbed, which is widely used in the research of trust-based recommendation. The main reason is that it includes both the information of user trust relationships and user/item ratings. Users can review items and assign them numeric ratings in the range of [1, 5]. They can also build their own trust network by adding the people whose reviews they think are valuable. We use the data set of Epinions.com published by Massa and Avesani [9]. It consists of 49,290 users who rated a total of 139,738 different items at least once. The total number of reviews is 664,824. The total number of issued trust statements is 487,181. Since we focus on the time-evolving opinion formulation, we do not run experiments on the whole data set. Alternatively, we extract a subset: we randomly choose 1,000 users; for each user, we choose at most six items that he has given ratings, and that rating can be predicted by the rating network. Finally, we restrict the maximum length to be 6, and a total of 5,548 pairs of users/items that can be predicted are selected. Table 3 shows the coverage of the selected sub data set, measuring the percentage of user/item pairs that can be predicted.

7.1.2 Evaluation Method

We use the leave-one-out method to evaluate the performance [9], [10]. If there is a user providing a rating to an item, the rating is masked and predicted through algorithms based on the rating network. Then, we compare the calculated value with the masked value.

For a given test pair, we first construct a rating network, by identifying raters and non-raters: the former are the users' friends within a given hop who have rated the item, while

TABLE 2
The Height (First Column) and Percentage of Different Raters (Second Column) in Containers $a_1 - a_5$ in Fig. 5

round	a_1		a_2		a_3		a_4		a_5	
1	0.06	(1,0,0,0)	0.10	(0,1,0,0)	0.13	(0,0,1,0)	0.09	(0,0,0,1)	0.00	(0,0,0,0)
2	0.13	(0.94,0.06,0,0)	0.20	(0,0.96,0.04,0)	0.25	(0,0,1,0)	0.17	(0,0,0.05,0.95)	0.01	(0,0,0,1)
3	0.20	(0.90,0.09,0,0)	0.29	(0,0.94,0.06,0)	0.35	(0,0,1,0)	0.25	(0,0,0.08,0.92)	0.03	(0,0,0.03,0.97)
50	2.02	(0.40,0.49,0.11,0)	2.29	(0,0.73,0.27,0)	2.50	(0,0,1,0)	1.91	(0,0,0.47,0.53)	1.58	(0,0,0.28,0.72)
100	2.46	(0.33,0.52,0.15,0)	2.47	(0,0.57,0.43,0)	2.46	(0,0,1,0)	2.37	(0,0,0.71,0.29)	2.34	(0,0,0.38,0.62)

TABLE 4
Parameter Settings

Parameter	Description	Value
h	fluid height in rater's container	10
b	cross-sectional area of containers	$\{1, 0.75, 0.5\}$
k	number of rounds	$[1, 250]$
Δ	time slot	0.04
θ	maximal initial height	$[0, 8]$
l	leak proportion	$[0, 0.01]$
q	uniform aggregation	$1/k$
	nonuniform aggregation	$[0.1, 0.9]$

Note: b represents the persuasiveness of a person. Currently, we set it to be 3 levels for simplicity. The value is set based on an intuition that, the more items that a person has known, the more persuasive he is. When he has rated more than five items, $b = 0.75$; when the number comes over 10, $b = 0.5$; otherwise, $b = 1$.

the latter are those who have not. Based on this, we relate users with containers and ratings as fluid temperatures, and then conduct multiple rounds of fluid updating using the FluidRating scheme. The temperatures of non-raters are collected and aggregated as their final predicted ratings.

7.1.3 Accuracy Metrics

We mainly consider the following four metrics for *rating prediction accuracy* (similar to [22]), representing the ability of predicting whether a predicted rating is consistent with the real rating.

- Precision. $P_h = A_h \cap B_h / B_h$, $P_l = A_l \cap B_l / B_l$, where A_h is the number of users who give a rating higher than 3 (the default threshold for the range [1, 5]), in the data set; and B_h is the number of that by prediction through the algorithm. A_l and B_l are of similar meaning, but with a rating lower than 3.

  Precision is the ratio of both the predicted and real higher (or lower) ratings over the predicted higher (or lower) ratings. A higher precision indicates a higher prediction accuracy.
- Recall. $R_h = A_h \cap B_h / A_h$, $R_l = A_l \cap B_l / A_l$. Recall is the ratio of both the predicted and real higher (or lower) ratings over the real higher (or lower) ratings. A higher recall indicates a higher prediction accuracy.
- FScore. $F_h = 2R_hP_h/(R_h + P_h)$, $F_l = 2R_lP_l/(R_l + P_l)$. Usually, there is a tradeoff between precision and recall. Therefore, FScore is used to measure the accuracy using Recall and Precision jointly.
- The root mean squared error (RMSE) [10] is used to measure the error in rating prediction: $RMSE = \sqrt{\sum(r_{u,i} - \hat{r_{u,i}})^2/D}$, where D is the total number of user/item pairs that can be predicted, and $r_{u,i}$ and $\hat{r_{u,i}}$ denote the real and predicted ratings, respectively. A smaller RMSE indicates a higher prediction accuracy.

7.1.4 Algorithms for Comparison

We select the following algorithms for comparison: (1) TidalTrust [14]. It finds all trusted raters with the shortest path distance from the sink, and aggregates their ratings, weighted by the trust between the sink and these raters. (2) MoleTrust [15]. It considers all raters up to a maximum-depth, which is given as an input, and is independent of any specific user and item. (3) Random Walk. Similar to [10], we set different thresholds on the number of steps in a random walk. (4) Personalized PageRank [1]. We take the result when it converges (e.g., the variation between two rounds is less than 0.001), with a minimum of 100 rounds. Table 4 shows the parameter settings.

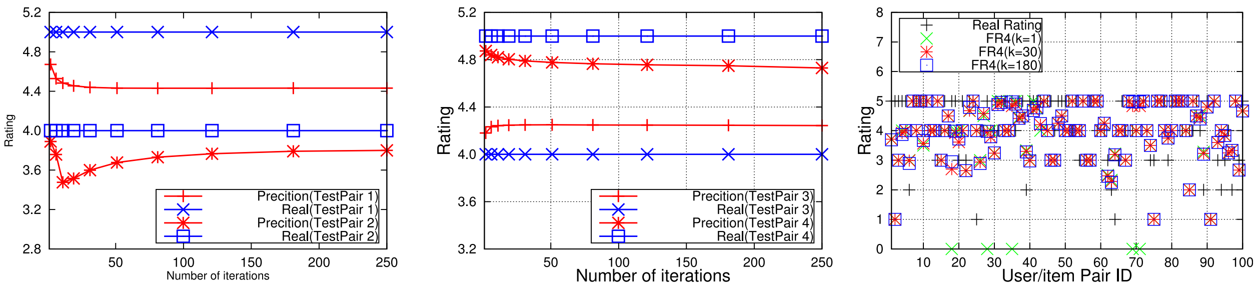
7.2 Experimental Results and Analysis

In this section, we present the results of our experiments. First we describe the findings of "first influence," then we analyze the effects of each impacting factor.

7.2.1 The Existence of the "First Influence"

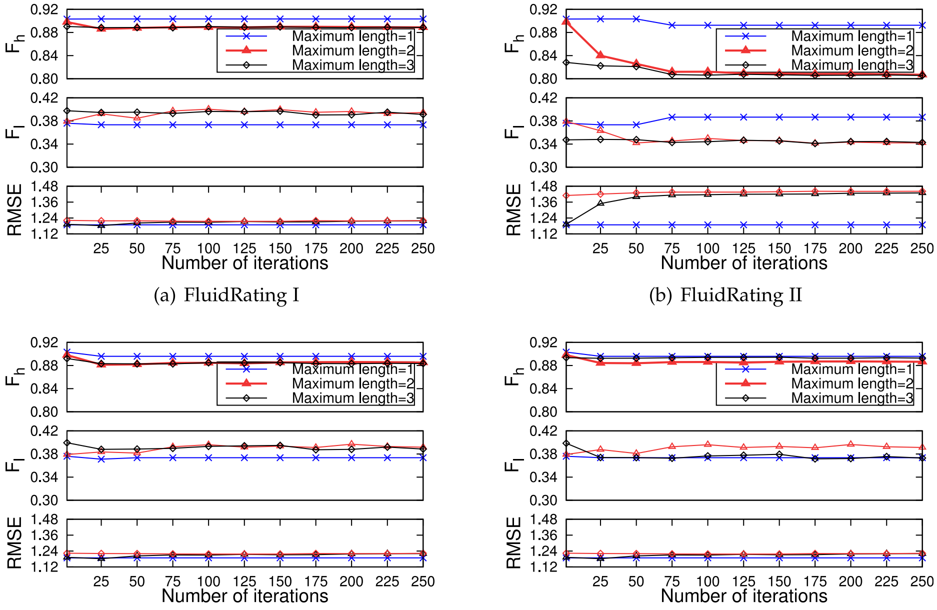
We observe the first influence phenomenon through all experiments, as shown in Fig. 6. Figs. 6a and 6b show four different patterns of four user/item pairs, where the real ratings of TestPairs 1, 2, and 4 are higher than the predicted rating, and that of TestPair 3 is lower. For TestPair 1, the real rating is 5, and the predicted rating first decreases when the number of samples, that is k, increases from 1 to 6; it then remains stable during the period afterward. For TestPair 2, there is a fluctuation when k varies from 2 to 6, then to 11 and 19. It first decreases, then increases, and finally stabilizes. TestPairs 3 and 4 show other patterns. The point is, in all of the four patterns, the first samples give predictions more close to the real truth.

In fact, this finding is a general phenomenon in the data set; for the sub-data set we use, the average rating when k is small is very close to the the real average rating, and when k becomes larger, the gap between real and predicted ratings



(a) Test pairs 1 and 2 (b) Test pairs 3 and 4 (c) 100 test pairs

Fig. 6. The phenomenon of the first influence. (a) and (b) show four different patterns of four user/item test pairs. (c) shows the predicted rating of 100 test pairs.



(a) FluidRating I (b) FluidRating II (c) FluidRating III (d) FluidRating IV

Fig. 7. The accuracy of using FluidRating I, FluidRating II, FluidRating III, FluidRating IV.

decreases gradually. This also indicates the refinements of users' opinions.

In addition, we record the predicted ratings and find that, real higher ratings tend to be predicted to be lower, while real lower ratings tend to be predicted as higher. It indicates that, our FluidRating model essentially takes into consideration another important phenomenon in social life: *conformity*. That is, a user's opinion usually tends to become closer to that of his friends. Moreover, it also shows that the real opinion formulation process is very complicated, and is accompanied with self opinion, peer influence, and group conformity.

Fig. 6c displays the results of 100 randomly chosen user/item pairs. It shows that, in some cases, the predicted rating is very close to the real rating, while some other cases are not. We analyze the meta-results, and find the reason is: for the latter test pairs, it usually happens that those users have few raters in their subgraph, or the raters' opinions are largely different from one another. In fact, this is what usually happens in real life. FluidRating is proposed exactly for modeling how a user formulates his opinion by combining the influence from different opinions over time. The impact factors and their effects are discussed in detail below.

7.2.2 The Effects of Aggregation Methods

As mentioned before, we design four aggregation methods: FluidRating I, FluidRating II, FluidRating III, and FluidRating IV. The accuracy results are shown in Fig. 7. The sub figures on the top show the Fscore of high rating prediction (rating ≥ 3), while those in the middle show that of low rating prediction (rating < 3), and the sub figures at the bottom show the RMSE. We have several findings.

1) The accuracy results of $k = 1$ are very close to the later ones, which validates the existence of "first influence" again. It also indicates that the algorithm converges quickly, and it gets steady after 50 rounds. For instance, the meta results using FluidRating II ($q = 0.5$) shows that: $|RMSE^{i+1} - RMSE^{i}| < 0.008$, when $i = 25$; and $|RMSE^{i+1} - RMSE^{i}| < 0.0001$, when $i = 50$.

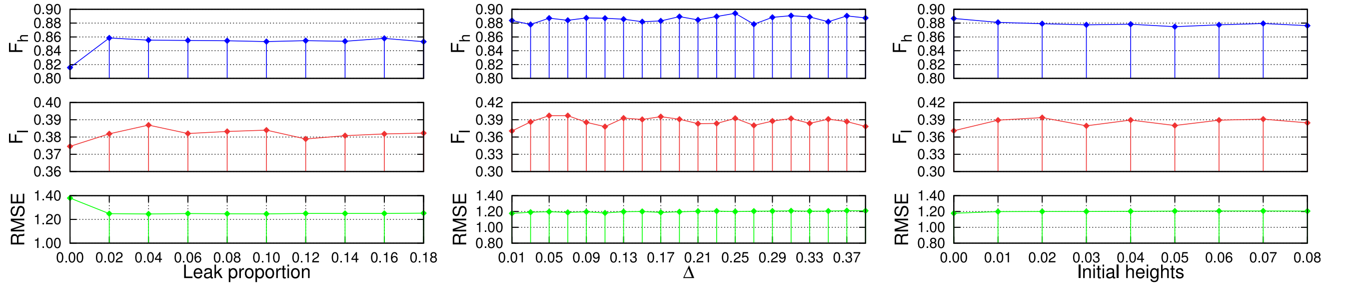
2) The accuracy of predicting high ratings is much higher. That is, F_h is at least over 80 percent. However, the accuracy is lower when predicting low ratings, with F_l being less than 40 percent. We argue that it is because there are more high ratings (than low ratings) in the data set: (a) Epinions is a friendly web community where the average rating is about 4 (the maximum is 5) [9]; (b) online users tend to avoid negative ratings, because of fear of retaliation from other parties. The meta result in Table 5 validates this, in which $A_h > 2{,}400$, while $A_l < 450$.

We further analyze why it is easier to predict high ratings. Since there are far more high ratings in the data set, it is common that for any target non-rater whose rating is being predicted, the surrounded raters give high ratings; or, there are more high ratings than low ratings among the surrounded raters. Then it is natural that the target's predicted rating is not low.

3) The accuracy of FluidRating I, FluidRating III, and FluidRating IV, are better and more stable than FluidRating II ($q = 0.5$). The reason is that, the sampling temperature in a later time is actually the accumulation of the earlier ones. FluidRating II gives more weight to earlier

TABLE 5
The Meta Accuracy Result of FluidRating IV

k	A_h	B_h	$A_h \cap B_h$	A_l	B_l	$A_l \cap B_l$
1	2,442	2,441	2,181	433	434	173
25	2,470	2,428	2,167	442	484	181
50	2,484	2,441	2,179	444	487	182
75	2,487	2,439	2,179	443	491	183
100	2,490	2,446	2,187	442	486	183
125	2,488	2,444	2,185	443	487	184
150	2,490	2,448	2,189	443	485	184
175	2,494	2,451	2,188	444	487	181
200	2,495	2,451	2,188	445	489	182
225	2,495	2,450	2,189	445	490	184
250	2,496	2,453	2,190	445	488	182



(a) Changing leak proportion (b) Changing Δ (c) Changing θ

Fig. 8. The effects of impact factors, taking FluidRating IV for instance: (a) leak proportion for reflecting the "forgetting" feature; (b) Δ for the time slot; and (c) θ for the initial fluid height.

TABLE 6
The Meta Accuracy Result of FluidRating IV, with Respect to Initial Opinion

θ'	$\theta \leq 0.01\theta'$			$\theta \leq 0.1\theta'$			$\theta \leq \theta'$		
	F_h	F_l	RMSE	F_h	F_l	RMSE	F_h	F_l	RMSE
0	0.8867	0.3709	1.1775	0.8867	0.3709	1.1775	0.8867	0.3709	1.1775
1	0.8811	0.3893	1.1987	0.8785	0.3899	1.1904	0.8766	0.3761	1.2011
2	0.8792	0.3937	1.2001	0.8794	0.3953	1.1952	0.8733	0.3847	1.2348
3	0.8775	0.3795	1.1999	0.8823	0.3996	1.1992	0.8698	0.3732	1.2686
4	0.8784	0.3895	1.2032	0.8792	0.3882	1.2074	0.8599	0.3638	1.2882
5	0.8749	0.38	1.2047	0.8756	0.3748	1.2137	0.8561	0.3563	1.2992
6	0.8775	0.3892	1.207	0.8791	0.4057	1.2241	0.8494	0.3576	1.3122
7	0.8794	0.3911	1.206	0.8772	0.3947	1.2424	0.8475	0.3382	1.3213
8	0.8764	0.3846	1.2057	0.8753	0.3851	1.2368	0.8449	0.3336	1.3323

samples, which may lead to information reuse. In addition, the performance of FluidRating III and FluidRating IV are very close to each other. That is because, FluidRating III gives more weight to the current samples (i.e., kth sample); meanwhile, FluidRating IV can be taken as an extreme case of FluidRating III, which gives full weight (i.e., 1) to the current sample. These findings validate the feasibility of our sample aggregation approaches; it also suggests that a decreasing weight sequence is better.

Since the four aggregation approaches show similar patterns, in the following experiments (if not specified), we take the results of FluidRating IV as the default.

7.2.3 The Effects of Leak Proportion

We set uniform leakage at the end of each time slot. The result of using FluidRating 3 is shown in Fig. 8a. We can see that the RMSE is reduced significantly when the leak proportion changes from 0 to 0.04, and then becomes smooth after that. The change of F_h and F_l seems insignificant compared to RMSE. This finding indicates that, proper settings of leak proportion can reduce the deviation between the predicted rating and real rating, with little changes of relative accuracy (i.e., Fscore).

7.2.4 The Effects of Time Slot and Sample Number

We test the effects of the time slot, Δ, as shown in Fig. 8b (other parameters use default set: $k = 250$, maximum length $= 3$). The accuracy shows a smooth change with respect to Δ. The effects of sample number (i.e., the total number of rounds), are shown in Fig. 7. The results validate two important things: "the first influence", and the convergence of our proposed algorithm.

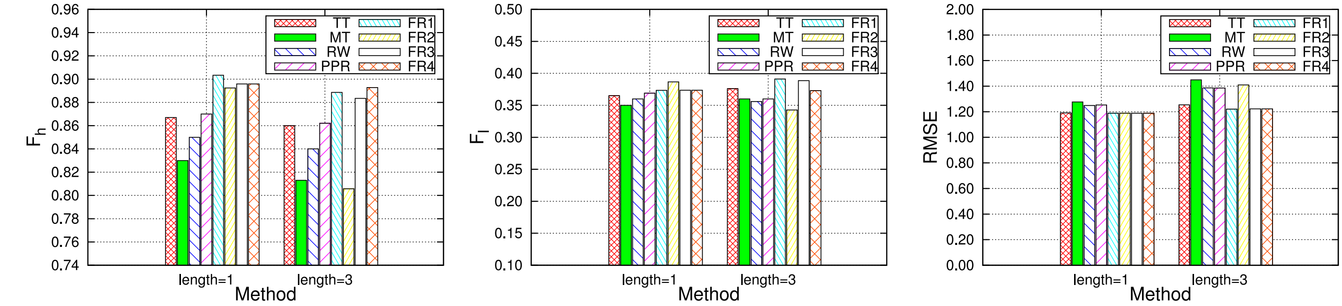
7.2.5 The Effects of Initial Opinion

To test the effect of the initial opinion, we set the initial fluid height of each non-rater randomly in the range of $(0, \theta)$, and the fluid temperature randomly in the range of $[1, 5]$. Fig. 8c shows the result of $\theta \in [0, 0.08]$. The most top sub figure shows the Fscore for high ratings (F_h), the middle for low ratings (F_l), and the bottom for RMSE. The results show that both F_h and RMSE reduce smoothly and insignificantly along with the maximum initial height; meanwhile, F_l is improved relatively significantly. Taking the result of $\theta \in [0.01, 0.08]$ for instance, F_h is reduced from 0.63 percent ($\theta = 0.01$) to 1.33 percent ($\theta = 0.05$), and RMSE is reduced from 1.8 percent ($\theta = 0.01$) to 2.5 percent ($\theta = 0.06$). F_l is increased from 2.33 percent ($\theta = 0.03$) to 5.63 percent ($\theta = 0.07$).

As shown in Table 6 (where θ' is a temporary variable), we change the initial fluid heights to fall into the range of $[0, 0.8]$, and the result shows a similar pattern (with a larger reduction, i.e., F_h is reduced by at most 1.28 percent , F_l by 9.39 percent and RMSE by 5.51 percent); when the range is set to be $[0, 8]$, the accuracy decreases sharply (i.e., F_h is reduced by at most 4.7 percent , and RMSE by 13.15 percent;

TABLE 7
The Effects of Considering Persuasiveness: Non-Uniform Container Size (Y) or Not (N)

Length	F_h		F_l		RMSE	
	N	Y	N	Y	N	Y
1	0.8955	0.8959	0.3977	0.3735	1.1889	1.1889
2	0.8051	0.8864	0.3427	0.3913	1.4221	1.2203
3	0.7807	0.8928	0.3467	0.373	1.5252	1.2226



(a) F_h (b) F_l (c) RMSE

Fig. 9. The comparison with other methods.

F_l first increases then reduces by 10.05 percent , with the changing point at $\theta = 3$). It indicates that the initial opinion does affect the prediction accuracy. In addition, it cannot be taken as too large of a portion.

7.2.6 The Effects of Persuasiveness

We compare the accuracy of considering persuasiveness or not. When considered, the cross-sectional area b is set to be non-uniform. Currently, we set three different levels. When the user gives ratings to more than 10 items, $b = 0.5$; when he gives ratings to less than 10, but more than five items, $b = 0.75$; when he gives ratings to less than five items, $b = 1$. When not considering persuasiveness, all cross-sectional areas are set to be $b = 1$. Table 7 shows the resulting accuracy. We can see that non-uniform container size produces a higher Fscore, and less RMSE. Particularly when the length is increased, the performance of uniform size decreases sharply, while that of non-uniform remains rather stable. The finding indicates the advantage of considering the feature of persuasiveness.

7.2.7 Comparison of Multiple Algorithms

We compare the Fscore and RMSE of using several trust-based recommendation methods. As shown in Fig. 9, FluidRating beats MoleTrust, RandomWalk, and Personalized Pagerank; meanwhile, TidalTrust performs almost as well as FluidRating. We analyze the reason to be that, TidalTrust takes the shortest and strongest recommendation path for rating prediction, which is exactly taking the first influence. In addition, the RMSE of using FluidRating I is 2.65 percent less than that of using TidalTrust (the best among other approaches) when maximum length=3, while the improvement of F_h is 3.29 percent, and that of F_l is 4.17 percent. That is to say, the proposed method shows slightly better precision. The achievement indicates its reasonability.

Moreover, the main contribution of this paper is to provide a novel and natural approach to understand and to model the process of opinion formulation in trust-based recommendation systems.

7.3 Summary of Experiments

Experimental results show that users' opinions do evolve with time, and verify the existence of the first influence phenomenon. The FluidRating model can flexibly handle those key points. The effects of multiple factors are tested. The proper settings of persuasiveness, forgetting, and initial opinions can benefit the prediction.

8 Conclusion and Future Work

Recommendation systems aim to predict the opinions of users on a target item, in order to determine whether or not to recommend the item to them. However, existing work focuses on the static rating prediction at the current time, and the prediction is usually conducted on a single user. To overcome the problems, we identify three features of human personality in forming and propagating recommendation: persistency, persuasiveness, and forgetting. The first two features can address the two challenges of forming opinions and refining them, respectively. The last feature can reflect the common truth of limited memory. Based on this, we propose a novel time-evolving rating prediction scheme using fluid dynamics theory, *FluidRating*. Fluid bears two dimensions of information: the temperature is taken as "opinion/rating," and the height is deemed as the "persistency" of the opinion. In this way, the two challenges of forming and refining opinions are solved naturally and gracefully. The experiments in Epinions, validate the reasonability and the effectiveness of the proposed model.

In future work, we will consider the evolution of the circle from a user's predicted rating to his real experience, and then to his trusted friends' rating.

Acknowledgments

This work is supported by US National Science Foundation (NSF) grants ECCS 1231461, ECCS 1128209, CNS 1138963; NSFC grants 61272151 and 61472451, ISTCP grant 2013DFB10070, the China Hunan Provincial Science & Technology Program under Grant Number 2012GK4106; and the Chinese Fundamental Research Funds for the Central Universities 531107040845. Some work of Wenjun Jiang was conducted at Temple University. G. Wang is the corresponding author.

References

[1] R. Andersen, C. Borgs, J. Chayes, U. Feige, A. Flaxman, A. Kalai, V. Mirrokni, and M. Tennenholtz, "Trust-based recommendation systems: An axiomatic approach," in *Proc. ACM 17th Int. Conf. World Wide Web*, 2008, pp. 199–208.

[2] A. Srinivasan, J. Teitelbaum, H. Liang, J. Wu, and M. Cardei, "Reputation and trust-based systems for ad hoc and sensor networks, " in, *Algorithms and Protocols for Wireless, Mobile Ad Hoc Networks*. New York, NY, USA: Wiley, 2008.

[3] X. Amatriain,"Mining large streams of user data for personalized recommendations," *SIGKDD Explorations*, vol. 14, no. 2, pp. 37–48, 2012.

[4] Y. Dai and J. Wu, "Sense in order: Channel selection for sensing in cognitive radio networks," in *Proc. 8th Int. Conf. Cognitive Radio Oriented Wireless Netw.*, 2013, pp. 74–79.

[5] Y. Zhao and J. Wu, "Socially-aware publish/subscribe system for human networks," in *Proc. Wireless Commun. Netw. Conf.*, 2010, pp. 1–6.

[6] Y. Zhao and J. Wu, "B-sub: A practical bloom-filter-based publish-subscribe system for human networks," in *Proc. IEEE 30th Int. Conf. Distrib. Comput. Syst.*, Jun. 2010, pp. 634–643.

[7] N. Koenigstein, G. Dror, and Y. Koren, "Yahoo! music recommendations: Modeling music ratings with temporal dynamics and item taxonomy," in *Proc. 5th ACM Conf. Recommender Syst.*, 2011, pp. 165–172.

[8] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 6, pp. 734–749, Jun. 2005.

[9] P. Massa and P. Avesani, "Trust-aware recommender systems," in *Proc. ACM Conf. Recommender Syst.*, 2007, pp. 17–24.

[10] M. Jamali and M. Ester, "Trustwalker: A random walk model for combining trust-based and Item-based recommendation," in *Proc. 15th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2009, pp. 397–406.

[11] X. Yang, H. Steck, and Y. Liu, "Circle-based recommendation in online social networks," in *Proc. 15th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2012, pp. 1267–1275.

[12] G. Wang, W. Jiang, J. Wu, and Z. Xiong, "Fine-grained feature-based social influence evaluation in online social networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 9, pp. 2286–2296, Sep. 2014.

[13] L. J. Bannon,"Forgetting as a feature, not a bug: The duality of memory and implications for ubiquitous computing," *CoDesign: Int. J. CoCreation Des. Arts*, vol. 2, no. 1, pp. 3–15, 2006.

[14] J. Golbeck, "Computing and applying trust in Web-based social networks," PhD thesis, Univ. of Maryland, College Park, MD, USA, 2005.

[15] P. Massa and P. Avesani, "Trust metrics on controversial users: Balancing between tyranny of the majority and echo chambers," *Int. J. Semantic Web Inf. Syst.*, vol. 3, pp. 39–64, 2007.

[16] G. Wang and J. Wu, "FlowTrust: Trust inference with network flows," *Frontiers Comput. Sci. China*, vol. 5, no. 2, pp. 181–194, 2011.

[17] M. Taherian, M. Amini, and R. Jalili, "Trust inference in web-based social networks using resistive networks," in *Proc. 3rd Int. Conf. Internet Web Appl. Serv.*, 2008, pp. 233–238.

[18] M. Balabanović and Y. Shoham, "Fab: Content-based, collaborative recommendation," *Commun. ACM*, vol. 40, no. 3, pp. 66–72, 1997.

[19] G. Linden, B. Smith, and J. York, "Amazon.com recommendations: Item-to-item collaborative filtering," *IEEE Internet Comput.*, vol. 7, no. 1, pp. 76–80, Jan. 2003.

[20] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-based collaborative filtering recommendation algorithms," in *Proc. ACM 10th Int. Conf. World Wide Web*, 2001, pp. 285–295.

[21] C. Desrosiers and G. Karypis. (2011). A comprehensive survey of neighborhood-based recommendation methods. in Recommender Systems Handbook, pp. 107–144. [Online]. Available: http://dblp.uni-trier.de/db/reference/rsh/rsh2011.html#DesrosiersK11

[22] W. Jiang, G. Wang, and J. Wu, "Generating trusted graphs for trust evaluation in online social networks," *Future Generation Comput. Syst.*, vol. 31, pp. 48–58, 2014.

[23] M. K. Reiter and S. G. Stubblebine, "Resilient authentication using path independence," *IEEE Trans. Comput.*, vol. 47, no. 12, pp. 1351–1362, Dec. 1998.

[24] P. Flocchini and F. L. Luccio, "Routing in series parallel networks," *Theory of Comput. Syst.*, vol. 36, no. 2, pp. 137–157, 2003.

[25] L. Page, "PageRank: Bring order to the web," *Stanford Digtial Libraries Working Paper*, 1997.

[26] S. Goel, W. Mason, and D. J. Watts, "Real and perceived attitude agreement in social networks," *J. Personality Social Psychol.*, vol. 99, no. 4, pp. 611–621, 2010.

[27] A. Das, S. Gollapudi, R. Panigrahy, and M. Salek, "Debiasing social wisdom," in *Proc. 19th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2013, pp. 500–508.

[28] H. Zhu, B. Huberman, and Y. Luon, "To switch or not to switch: Understanding social influence in online choices," in *Proc. ACM Conf. Human Factors Comput. Syst.*, 2012, pp. 2257–2266.

[29] E. Bakshy, I. Rosenn, C. Marlow, and L. Adamic, "The role of social networks in information diffusion," in *Proc. ACM 21st Int. Conf. World Wide Web*, 2012, pp. 519–528.

[30] H. Ma, H. Yang, M. R. Lyu, and I. King, "Mining social networks using heat diffusion processes for marketing candidates selection," in *Proc. 17th ACM Conf. Inf. Knowl. Manage.*, 2008, pp. 233–242.

[31] D. Kempe, J. Kleinberg, and E. Tardos, "Maximizing the spread of influence through a social network," in *Proc. 9th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2003, pp. 137–146.

[32] E. R. Smith and D. M. Mackie, *Social Psychology*. Philadelphia, PA: Psychology Press, 2007.

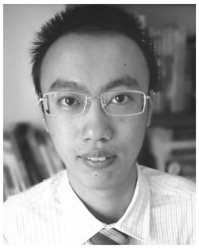
[33] P. G. Hewitt, *Conceptual Physics*, 11th Ed. Reading, MA, USA: Addison-Wesley, 2009.

[34] S. A. Myers, C. Zhu, and J. Leskovec, "Information diffusion and external influence in networks," in *Proc. 18th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2012, pp. 33–41.

Wenjun Jiang received the bachelor's degree in computer science from Hunan University, P. R. China, in 2004, the master's degree in computer software and theory from the Huazhong University of Science and Technology, P. R. China, in 2007, and the doctorate degree in computer software and theory from Central South University, P. R. China, in 2014. She has been a visiting PhD student at Temple University for two years. Currently, she is an assistant professor at Hunan University. Her research interests include trust and social influence evaluation models and algorithms in online social networks, recommendation systems, and social network analysis.

Jie Wu is the chair and a Laura H. Carnell professor in the Department of Computer and Information Sciences at Temple University. He is also an Intellectual Ventures endowed visiting chair professor at the National Laboratory for Information Science and Technology, Tsinghua University. Prior to joining Temple University, he was a program director at the National Science Foundation and was a distinguished professor at Florida Atlantic University. His current research interests include mobile computing and wireless networks, routing protocols, cloud and green computing, network trust and security, and social network applications. He regularly publishes in scholarly journals, conference proceedings, and books. He serves on several editorial boards, including *IEEE Transactions on Service Computing* and the *Journal of Parallel and Distributed Computing*. He was a general co-chair/chair for IEEE MASS 2006, IEEE IPDPS 2008, IEEE ICDCS 2013, and ACM MobiHoc 2014, as well as a program co-chair for IEEE INFOCOM 2011 and CCF CNCC 2013. He was an IEEE Computer Society distinguished visitor, ACM distinguished speaker, and a chair for the IEEE Technical Committee on Distributed Processing (TCDP). He is the recipient of the 2011 China Computer Federation (CCF) Overseas Outstanding Achievement Award. He is a CCF distinguished speaker and a fellow of the IEEE.

Guojun Wang received the BSc degree in geophysics, MSc degree in computer science, and the PhD degree in computer science from Central South University, China. He is now a chair and a professor of the Department of Computer Science and Technology at Central South University. He is also a director of the Trusted Computing Institute of the University. He has been an adjunct professor at Temple University; a visiting scholar at Florida Atlantic University; a visiting researcher at the University of Aizu, Japan; and a research fellow at the Hong Kong Polytechnic University. His research interests include network and information security, Internet of things, and cloud computing. He is a distinguished member of CCF, and a member of the IEEE, ACM, and IEICE.



Huanyang Zheng received the BEng degree in telecommunication engineering from the Beijing University of Posts and Telecommunications, Beijing, China, in 2012. He is currently working towards the PhD degree in the Department of Computer and Information Sciences, Temple University, Philadelphia, PA. His current research focuses on the delay tolerant networks, social networks, and cloud systems.