Inf Syst Front (2013) 15:533–551
DOI 10.1007/s10796-012-9377-6

Web 2.0 Recommendation service by multi-collaborative filtering trust network algorithm

Chen Wei · Richard Khoury · Simon Fong

Published online: 2 September 2012

Abstract Recommendation Services (RS) are an essential part of online marketing campaigns. They make it possible to automatically suggest advertisements and promotions that fit the interests of individual users. Social networking websites, and the Web 2.0 in general, offer a collaborative online platform where users socialize, interact and discuss topics of interest with each other. These websites have created an abundance of information about users and their interests. The computational challenge however is to analyze and filter this information in order to generate useful recommendations for each user. Collaborative Filtering (CF) is a recommendation service technique that collects information from a user's preferences and from trusted peer users in order to infer a new targeted suggestion. CF and its variants have been studied extensively in the literature on online recommending, marketing and advertising systems. However, most of the work done was based on Web 1.0, where all the information necessary for the computations is assumed to always be completely available. By contrast, in the distributed environment of Web 2.0, such as in current social networks, the required information may be either incomplete or scattered over different sources. In this paper, we propose the Multi-Collaborative Filtering Trust Network algorithm, an improved version of the CF algorithm designed to work on the Web 2.0 platform. Our simulation experiments show that the new algorithm yields a clear improvement in prediction accuracy compared to the original CF algorithm.

Keywords Recommendation system · Collaborative Filtering · Social network · Web 2.0

1 Introduction

Social networking sites, such as Facebook for instance, have been growing at an unprecedented rate in recent years (Korvenmaa 2009). This new platform, commonly known as Web 2.0, is characterized in part by cultivating open communication and information sharing between users and their friends. Many companies have already established their virtual presence on social networking sites and started reaching out to potential customers through a network of friends of friends (Massari 2010). In this distributed environment, customer relationship management (CRM) faces some new difficulties. The basic marketing principle remains the same as those of CRM for Web 1.0; namely to recommend products to customers according to their personal preferences (Hung 2005). However, the information used to make these recommendations is social, distributed, and not always available to the recommendation algorithm. Collaborative Filtering (CF) is a well-known technique to implement a recommendation service (RS) that is based on peer user information and on product information.

A Web 2.0 recommendation service can be seen as a system service provided by the social network host to display relevant advertisements on each user's account based

C. Wei (✉)
Lenovo China,
No.6 ChuangYe Road, Haidian District,
Beijing 100085, China
e-mail: weichen1@lenovo.com

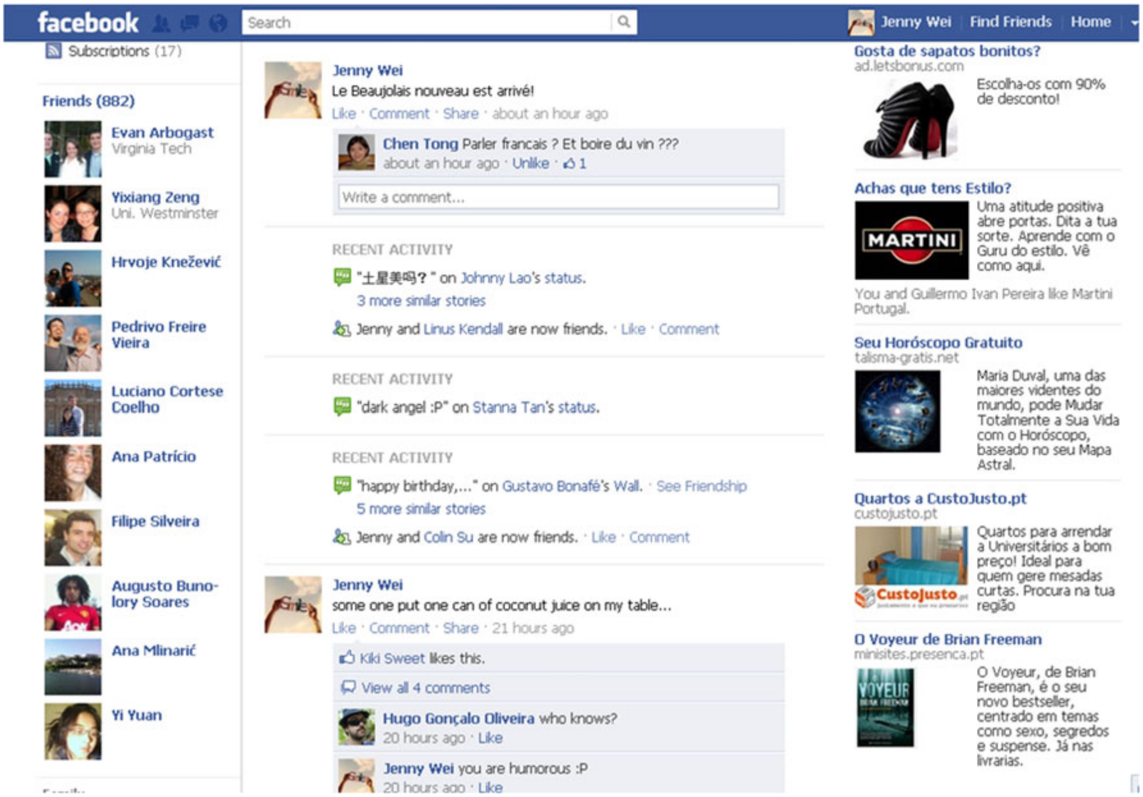
S. Fong
Faculty of Science and Technology, University of Macau,
Av. Padre Tomás Pereira Taipa,
Macau, S.A.R, China
e-mail: ccfong@umac.mo

R. Khoury
Department of Software Engineering, Lakehead University,
955 Oliver Road,
Thunder Bay, ON P7B 5E1, Canada
e-mail: richard.khoury@lakeheadu.ca

on their personal tastes and demographic data. This system service can potentially develop into a business product itself, which is offered by the host to businesses to propagate their advertisements to accurately-targeted groups of users. The service aims to recommend customer items (such as books, news, web pages, movies, bands, etc.) or social components (such as events, groups, friends, etc.) that are likely to be of interest to the user. It determines which recommendations to offer to a user by filtering information from similar users in a social network. Figure 1 shows a real-world example of a recommendation system. On the right side of that figure, there are advertisements for shoes, wine, and other products. The user in this figure and some of her friends indicate a preference for the wine Martini by clicking a “Like” button some time ago, and as a result an advertisement recommending Martini Portugal appears on the user’s page.

RS can use a number of different technologies. Typically, they use one of two types of filtering techniques, namely content-based filtering or collaborative filtering. Content-based filtering systems track the properties of items preferred by a user in order to recommend similar items. For instance, if a YouTube user has listened to several Mariah Carey songs, then the system would recommend movies in which she starred as well as other musical artists of the same genre. On the other hand, collaborative filtering systems recommend items based on the preferences of similar users. If a user and several of his friends list Mariah Carey among their preferred artists, and the friends all list a second artist as well, this artist will be recommended to the user.

The workflow illustrated in Fig. 2 shows how a general RS operates by aggregating information from different sources, analyzing it, and inferring recommendations to offer to online users. The first step is to connect the information collected from different online sources. Facebook proposed the Open Graph protocol at the 2010 F8 Conference as a way to connect different social networking sites that have overlapping information. For example, using this framework a news post a Facebook user makes about having a business lunch at a specific restaurant could be enriched with his LinkedIn job information and an UrbanSpoon restaurant review. Alternatively, the information could come from the sensors of a smart home, such as the one suggested in Song and Kim (2009), which can recognize the house inhabitants and their current context automatically. Using Open Graph, or an equivalent technology if one is available, it thus becomes possible to match up the diverse sources of social information and product reviews available on the internet. After a few steps of data preprocessing and cleaning, the information can be stored in a data warehouse for future use. This corresponds to the “data extraction” step in Fig. 2. The next step in that figure is “data



Subscriptions (17)

Friends (882)

Evan Arbogast
Virginia Tech

Yixiang Zeng
Uni. Westminster

Hrvoje Knežević

Pedrivo Freire Vieira

Luciano Cortese Coelho

Ana Patrício

Filipe Silveira

Augusto Bunolory Soares

Ana Mlinarić

Yi Yuan

Jenny Wei
Le Beaujolais nouveau est arrivé!
Like · Comment · Share · about an hour ago

Chen Tong Parler francais ? Et boire du vin ???
about an hour ago · Unlike · 1

Write a comment...

RECENT ACTIVITY
"土星美吗? " on Johnny Lao's status.
3 more similar stories
Jenny and Linus Kendall are now friends. · Like · Comment

RECENT ACTIVITY
"dark angel :P" on Stanna Tan's status.

RECENT ACTIVITY
"happy birthday,..." on Gustavo Bonafé's Wall. · See Friendship
5 more similar stories
Jenny and Colin Su are now friends. · Like · Comment

Jenny Wei
some one put one can of coconut juice on my table...
Like · Comment · Share · 21 hours ago

Kiki Sweet likes this.

View all 4 comments

Hugo Gonçalo Oliveira who knows?
20 hours ago · Like

Jenny Wei you are humorous :P
20 hours ago · Like

Gosta de sapatos bonitos?
ad.letsbonus.com
Escolha-os com 90% de desconto!

Achas que tens Estilo?
Uma atitude positiva abre portas. Dita a tua sorte. Aprende com o Guru do estilo. Vê como aqui.
You and Gullermo Ivan Pereira like Martini Portugal.

Seu Horóscopo Gratuito
talisma-gratis.net
Maria Duval, uma das maiores videntes do mundo, pode Mudar Totalmente a Sua Vida com o Horóscopo, baseado no seu Mapa Astral.

Quartos a CustoJusto.pt
custojusto.pt
Quartos para arrendar a Universitários a bom preço! Ideal para quem gere mesadas curtas. Procura na tua região

O Voyeur de Brian Freeman
minisites.presenca.pt
O Voyeur, de Brian Freeman, é o seu novo bestseller, centrado em temas como sexo, segredos e suspense. Já nas livrarias.

Fig. 1 Example of a recommendation service on Facebook

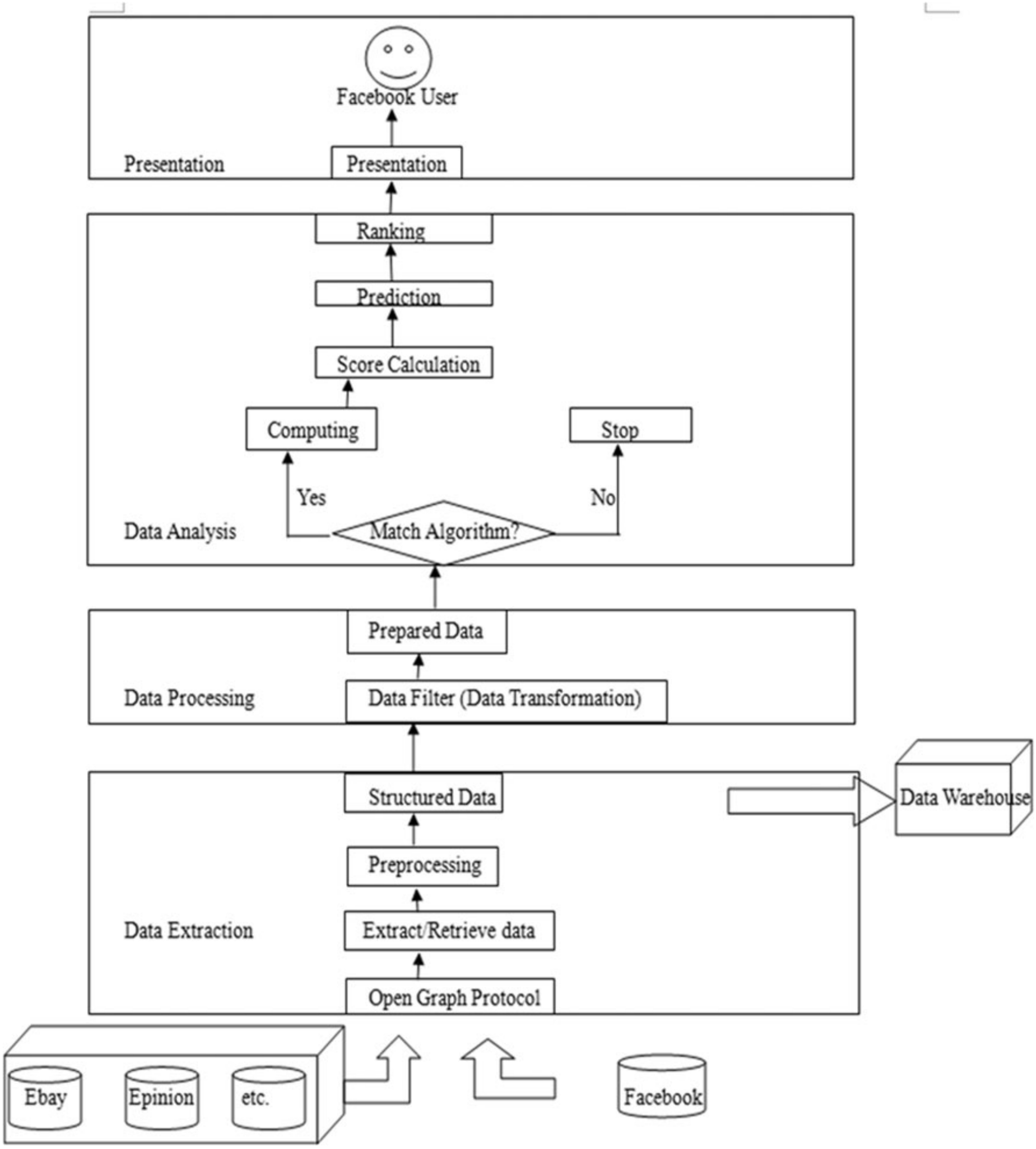


Fig. 2 Workflow of a recommendation service

processing", where the data is prepared for the specific RS task it will be needed for. This may involve for example filtering out the types of social relationships between users to only consider recommendations from close relatives and friends. The "data analysis" step is the crucial data mining stage of the RS system, where the information is used to predict what type of product a given user may be interested in. Finally, the information is given to the user in the "presentation" stage using the interface of whatever website he is using, such as the advertisement bar shown in Fig. 1. In this paper, we focus on the "data analysis" step, and we develop a Multi-Collaborative Filtering Trust Network (MCFTN) as a method to mine recommendations from the information collected from multiple different source websites.

The rest of this paper is organized as follows. We will give an overview of related work in RS system in Section 2, and position our work with respect to the literature. In Section 3 we will further highlight the main challenges that our work tackles. Our MCFTN algorithm is presented in mathematical details in Section 4, and thoroughly tested and analyzed in Section 5. Finally, we give some concluding remarks in Section 6.

2 Related work

A common problem faced by recommendation systems is the Cold Start problem, sometimes also called the Cold Boot-up problem. In essence, given a user with a small circle of friends and a limited amount of activity on the social network (such as a newly-registered user for example), a recommendation system will have very little data to use to infer good recommendations, and moreover the chances that the data available will help target a specific product are very small. To deal with this problem, Sandholm, Ung, Aperjis, and Huberman proposed a geo-aware rating and incentive mechanism using the Kendall Rank Correlation Coefficient algorithm (Sandholm et al. 2010), which improves the ranking and aggregate rating of a series of location-dependent Web pages. Desarkar, Sarkar, and Mitra proposed a memory-based algorithm (Desarkar et al. 2010) by comparing and aggregating four existing approaches, namely: User based Pearson Correlation (UPCC), Item based Pearson Correlation Coefficient (IPCC), Random Walk Recommender (RWR) and Somers Coefficient based algorithm.

Massa studied the issue of trust between users in social networks (Massa and Bhattacharjee 2004). He proposed building a trust matrix by letting users explicitly rate their level of trust of other users, in essence creating a “web of trust”. By propagating a trust matrix in addition to the product ratings matrix, his work extended recommendation systems into Trust-aware recommendation systems (Gursel and Sen 2009). However, his design is based on epinion.com and Page Rank (Page et al. 1998), which are not exactly social platforms. By contrast, our CF framework considers the activity between users on Facebook and uses this information to estimate the trust level between a pair of users without the need for them to explicitly express or rate their trust for each other. Based on Massa’s research (Massa and Avesani 2007), we measure the trust both by relation and by reputation.

Several variations of Massa’s trust-aware recommendation systems emerged in recent years. For example, the Reputation-based Trust-Aware Recommendation System (Desarkar et al. 2010) includes social factors (e.g. users past behaviors and reputation) as a component of trust. Another variant is the trust-aware recommendation model (TARM) (Sun et al. 2008), which does not handle all users as equals but rather tracks trustworthy experts on a given topic. These experts’ experiences are used to generate recommendations for laymen users with similar profiles and preferences. The authors later extended their model (Wu et al. 2009) by replacing the similarity weight with a trust weight obtained by propagation over the trust network. They innovated in that aspect by introducing a penalty measure that decreases the trust based on the length of propagation.

The next major innovation has been the measure of temporal relations, or of how social relationships and trust changes over time. Campos, Bellogin, Diez, and Chavarriaga proposed a Time-Biased KNN-based recommendation algorithm (Campos et al. 2010), which considers some simple strategies based on variations of the K-Nearest Neighbor (KNN) algorithm to take into account temporal context in its recommendations. Brenner, Pradel and Usunier proposed a time-dependent collaborative personalized recommendation algorithm (Brenner et al. 2010), which modeled the short-term evolution of the probability and predicted the behavior of these probabilities in the future. Rendle and Schmidt-thieme proposed a Time-aware Matrix Factorization Model (Time SVD) (Rendle and Schmidt-thieme 2010), which extended the widely-used neighborhood-based recommendation algorithms by incorporating temporal information. They also developed an incremental algorithm to update the neighborhood similarities when new data is acquired.

A general framework for recommendation systems specifically designed for online social networking sites was proposed in Kazienko and Musial (2006). It integrates several sources of data in order to generate relevant personalized recommendations for the users. These sources include information taken from the users’ profile pages and the relationships that can be inferred from the level of activity between users. The users’ profiles measure their activity within the community, the number and duration of the users’ relationships with their peers, and other features that characterize them. The framework thus considers of most if not all social elements; however it lacks weight (or significance) scales for these social elements to represent how much each element contributes to the trust between users.

To summarize, we present in Tables 1 and 2 a sample of recent work in this area, and indicate whether it deals with the cold boot-up problem, handles data from multiple sources, and if it considers temporal relation and trust factors. For comparison, we included in this table the algorithm we propose in this paper. It appears that our proposed algorithm is the only one to deal with all four points.

3 Critical challenges

As we mentioned earlier, CF is a well-known technique for implementing a recommender system that is makes use of both peer user information and product information. There are nonetheless open challenges when it comes to applying this technique on Web 2.0, related namely to the availability of data required in the CF computation. We can highlight two specific challenges.

The first challenge comes from the fact that the user information needed by the algorithm may not be fully available in the social network. In a normal Web 1.0 e-business, user information is stored as well-structured customer records in a relational database. The information is thus complete, trusted and usable. Web 2.0 social networks, however, are partially-connected graphs where each node is a user with their own information and the connections between the nodes

Table 1 Comparison of related work by paper

Authors	Title	Area of work			
		Cold boot-up problem	Multiple sources	Trust factor	Temporal relation
Sandholm et al. (2010)	Global Budgets for Local Recommendations	✓	x	x	x
Rendle and Schmidt-thieme (2010)	Factorization Models for Context/Time-Aware Movie Recommendations	x	x	x	✓
Campos et al. (2010)	Simple Time-Biased KNN-based recommendations	x	x	x	✓
Desarkar et al. (2010)	Aggregating Preference Graphs for Collaborative Rating Prediction	✓	x	x	x
Brenner et al. (2010)	Predicting Most Rated Items in Weekly Recommendation with Temporal Regression	x	x	x	✓
Liu et al. (2010)	Online Evolutionary Collaborative Filtering	x	x	x	✓
Massa and Avesani (2007)	Trust-aware Recommender Systems	✓	x	✓	x
Wei, Khoury and Fong	MCFTN	✓	✓	✓	✓

are the social connections between the users. Two nodes are not connected if the two users do not know each other, and the number of connections separating two users represents the level of trust between them. We may further see a section of the graph encompassing all nodes a set number of connection steps away from a given central node as the circle of friends of a given user. However, information about a given product may not exist in a section of the graph, meaning that no one within the circle of friends has experience with that product, or the information may come from too far away in the network (too distant a relation) to be considered trustworthy.

The second challenge is the Cold Boot-up problem we mentioned previously. This challenge is inherent from the nature of partially-connect graphs and from the fact that product information is extracted from social interactions, or posts and comments made by users in the social network. A user who has a small circle of friends and a small history of interaction will yield very little data to mine for information. The chances that the interactions involve, explicitly or implicitly, a specific product are extremely small.

Both of these problems will be exacerbated when the product to recommend is highly specialized or uncommon. For example, suppose that an accountant wishes to pick up a new hobby playing the piccolo. His circle of Facebook friends, including trusted ones like family members and work colleagues, may have little or no experience with piccolos, and thus cannot be mined for useful information and recommendations.

Table 2 Comparison of related work by algorithms

Reference	Algorithm	Area of work			
		Cold boot-up	Multiple sources	Trust rank	Temporal relation
(Sandholm et al. 2010)	Kendall Rank Correlation Coefficient	✓	x	x	x
(Campos et al. 2010)	Combining Prediction	✓	x	x	x
	Item-KNN	✓	x	x	x
(Campos et al. 2010)	Time-Biased KNN	x	x	x	✓
(Campos et al. 2010)	Time-Periodic-Biased KNN	x	x	x	✓
(Desarkar et al. 2010)	User-based Pearson Correlation Coefficient (UPCC)	✓	x	x	x
(Desarkar et al. 2010)	Item-based Pearson Correlation Coefficient (IPCC)	✓	x	x	x
(Desarkar et al. 2010)	Somers Coefficient based approach (Somers)	✓	x	x	x
(Desarkar et al. 2010)	Random Walk Recommender (RWR)	✓	x	x	x
(Desarkar et al. 2010)	Random Walk Restart Recommender (RWRR)	✓	x	x	x
(Brenner et al. 2010)	Time-Dependent Collaborative Personalized RS	x	x	x	✓
(Rendle and Schmidt-thieme 2010)	Time-Aware Matrix Factorization Model (Time SVD)	x	x	x	✓
(Massa and Avesani 2007)	Trust Aware	✓	x	✓	x
	Multiple Collaborative Filtering Trust Network (MCFTN)	✓	✓	✓	✓

It can be seen that both challenges stem at a fundamental level from the same problem: the lack of information in a user's circle of friends, be it because of the partially-connected nature of the graph in the first case or because of the limited size of the user's circle of friends in the second case. By design, our system tackles both challenges at that fundamental level, by making more trustworthy information usable for recommendations to that user. Indeed, while the experiences available in a user's circle of friends may be limited, there is abundant information to be mined from social networks, especially those on specialized topics such as Pandora for music, Epinions for consumer goods, Urban-Spoon for restaurants, and so on. This would deal with the Cold Boot-up problem, but the fact this information is too distant from a user node to be trusted, as highlighted in the first challenge, still remains. To deal with this, we developed trust metrics between users, which we will present in the next section. For example, the distance in number of connections between two users, which was mentioned previously, is considered in the "trust by relation" metric. The "trust by reputation" refines this measure by notably taking into account the activity and interaction between the users, such as the number of wall-to-wall posts they made to each other. This allows our recommendation system to use trustworthy information taken from anywhere in graph rather than be limited to a user's circle of friends, and to know exactly how trustworthy this information is.

In theory, the more sources of information the CF algorithm can tap into, the better its recommendations will be. This solution to the scarcity of information in social networks is the motivation behind our algorithm, which combines social networks, which are good at expressing the social relationships between users needed for our trust metrics, with customer review websites that contain the information needed to make specialized recommendations.

4 MCFTN algorithm

In this section, we will present our MCFTN algorithm. As we will show, our algorithm combines recommendations from two sources to in order to make a superior recommendation. It uses, on the one hand, the recommendations made by trusted social-networking-site friends and acquaintances of a user who have similar tastes to the user, and on the other hand the recommendations made by the trusted experts of specialized websites discussing similar products.

In order to make our explanation clearer and easier to understand, we have divided it thusly. Section 4.1 will introduce the mathematical foundations of similarity between users and between items. Moreover, it was shown in Brenner et al. (2010) that this similarity was a time-dependent metric. This time-decay element is introduced into our similarity equations in Section 4.2, and we use it to develop a recommendation score prediction equation. In addition to similarity, our MCFTN system takes into account the level of trust the user has in the source of each recommendation. After an introduction to the notion of trust in Section 4.3, we give the mathematical foundations to computer trust between relations on a social network in Section 4.4 and trust in the reputation of an expert in Section 4.5. In Section 4.6 we update our recommendation score prediction equation from Section 4.2 to take into account the trust metrics. It will become apparent from the mathematical development we will present that several weights are needed to balance the multiple sources of information that can be included in the trust equation. This weakness is not unique to our system but present in many CF systems, and in many cases they are arbitrarily set by the system designers. By contrast, in Section 4.7 we propose instead a novel method to estimate these weights from the social network data. Finally, we bring all these notions together in Section 4.8 and give the complete view of our system along with the complete recommendation score prediction equation.

4.1 Similarity

There are several different ways to calculate the similarity between two users. However, the two most commonly-used methods in the literature are to assign weights to the attributes or to use Pearson Correlation, a measure of the degree to which the variables are related. In Debnath et al. (2008), the similarity between users a and u is aggregated by a series of difference functions $f(A_{na},A_{nu})$ with respect to each attribute A_n in the users' profiles:

$$S(a,u) = \sum\nolimits_{n=1}^{N} w_n f(A_{na},A_{nu})$$

where w_n is the relative weight given in the system to attribute A_n, and $\sum\nolimits_{n=1}^{N} w_n = 1$. The function $f(A_{ni},A_{nj})$ is a binary comparison function:

$$f(A_{na},A_{nu}) = \begin{cases} 1 & \text{if } A_{na} = A_{nu} \\ 0 & \textit{otherwise} \end{cases}$$

Alternatively we could define $f(A_{na}, A_{nu})$ to return a result in the interval [0,1] in order to handle ordinal or numeric data, using for example Pearson function.

For example, assume we have two users with the attributes listed in Table 3, and that we have predefined the interest weights as in Table 4. The system would compute their similarity as $S(u_1,u_2) = 0.1 \times 0 + 0.39 \times 0 + 0.14 \times 1 + 0.23 \times 1 + 0.12 \times 1 = 0.49$.

Table 3 Users and attributes table for calculating similarity

Users attributes	*a*	*u*
Sex	Female	Male
Music	no	Classic Music
Reading	Novels	Novels
Relationship	Making Friends	Making Friends
Travel	Yes	Yes

In Desarkar et al. (2010), the authors proposed the User-based Pearson Correlation Coefficient (UPCC) and the Item-based Pearson Correlation Coefficient (IPCC) to calculate the similarity between users and items respectively. In user-user collaborative filtering, the PCC defines the similarity between two users a and u based on the items they rated or related in common:

$$Sim(a,u) = \frac{\sum_{i \in I(a) \cap I(u)} \left(r_{a,i} - \overline{r_a}\right)\left(r_{u,i} - \overline{r_u}\right)}{\sqrt{\sum_{i \in I(a) \cap I(u)} \left(r_{a,i} - \overline{r_a}\right)^2} \sqrt{\sum_{i \in I(a) \cap I(u)} \left(r_{u,i} - \overline{r_u}\right)^2}}$$

where $Sim(a,u)$ stands for the similarity between user a and user u, $I(a)$ is the set of items user a has rated, $I(u)$ is the set of items user u has rated, and i is an item in the intersection of both sets. The value $r_{a,i}$ is the rating user a gave to item i, and $\overline{r_a}$ represents the average rating given by user a. From this definition, it can be seen that the user similarity $Sim(a,u)$ is in the range [0,1], and a larger value means user a and user u give similar ratings to the same items.

In item-item collaborative filtering, the similarity between two items i and j can be expressed as:

$$Sim(i,j) = \frac{\sum_{u \in U(i) \cap U(j)} \left(r_{u,i} - \overline{r_i}\right)\left(r_{u,j} - \overline{r_j}\right)}{\sqrt{\sum_{u \in U(i) \cap U(j)} \left(r_{u,i} - \overline{r_i}\right)^2} \sqrt{\sum_{u \in U(i) \cap U(j)} \left(r_{u,j} - \overline{r_j}\right)^2}}$$

where $Sim(i,j)$ is the similarity between item i and item j, $U(i)$ and $U(j)$ are the set of users who rated items i and j respectively, and u belongs to the subset of users who rated both items. The value $r_{u,i}$ is the rating user u gave item i, and $\overline{r_i}$ represents the average rating of item i. Like for user similarity, item similarity is in the range [0,1] and a higher value represents that users give similar ratings to both items.

Table 4 Different weights assigned to each attributes

w_{Sex}	w_{Music}	w_{Reading}	$w_{\text{Relationship}}$	w_{Travel}
0.1	0.39	0.14	0.23	0.12

4.2 Similarity with temporal relation

The intuition behind the temporal version of CF is that the strength of a relationship should fade gradually over time. The more recently a relationship activity occurred, the more important it is. For example, a comment made today should have stronger weight than one made last month when evaluating the relationship between two users. We deal with the temporal nature of the relationships using a modified version of the temporal relation from Brenner et al. (2010). The similarity between items i and j over time can be expressed as follow:

$$s_{ij}(t) = \frac{\sum_{u \in U_i^t \cap U_j^t} \left(f_{ui}^{\alpha}(t) \cdot r_{ui}\right)\left(f_{uj}^{\alpha}(t) \cdot r_{uj}\right)}{\sqrt{\sum_{u \in U_i^t} \left(f_{ui}^{\alpha}(t) \cdot r_{ui}\right)^2 \sum_{u \in U_j^t} \left(f_{uj}^{\alpha}(t) \cdot r_{uj}\right)^2}}$$

where U_i^t is the set of users who rated item i at time t. In this equation, the temporal relevance is defined as $f_{ui}^{\alpha}(t) = e^{-\alpha(t - t_{ui})}$ and the parameter α controls the decay rate:

$$f_{ui}^{\alpha}(t+1) = \gamma \cdot f_{ui}^{\alpha}(t)$$

where the constant $\gamma = e^{-\alpha}$ denotes a constant decay rate. Next, we find the k items that are most similar to i according to S_i to form the neighborhood N_i. The Score Prediction $r_{ui}^{S}(t)$ that we expect user u to give to item i based on the temporal similarity of i to other items that the user rated can then be computed as follow:

$$\widehat{r}_{ui}^{S}(t) = \overline{r}_u(t) + \frac{\sum_{j \in N_i^t \cap \tau_u^t} S_{ij}(t) \cdot f_{uj}^{\beta}(t) \cdot r_{uj}}{\sum_{v \in N_i^t \cap \tau_u^t} S_{ij}(t) \cdot f_{uj}^{\beta}(t)}$$

where $\overline{r}_u(t)$ represents the average rating given by user u with time decay.

4.3 Trust

Massa's contribution to the area of RS systems was to add a trust matrix to represent the users' trust in each other's ratings (Massa and Avesani 2007). This essentially turns the system into a trust-aware recommendation system. Several other authors have since developed their own versions of a trust metric (Kitisin and Neuman 2006; Sun et al. 2008; Wu et al. 2009). However, trust metrics are still a very recent development and there haven't been thorough analyses of which metrics perform better in different scenarios. In this work, we develop trust metrics that are characterized by the features of Facebook. The trust matrix we propose is compatible with Massa's architecture and could be integrated into his system in future work. The following two sections present our trust metrics.

4.4 Trust by relation

Measuring trust in a social network environment is a challenging task due to the wide range and mostly indiscriminate nature of relationships on the network. In Facebook, for instance, a pair of users who have added each other as friends could be family members, friends, friends-of-friends, work colleagues, members of a same hobby group, activity partners, or even strangers who met once on a trip. The nature of the relationship does impact the level of trust between the individuals. For instance, a recommendation made by a family member is perceived as more trustworthy than that made by an online friend who has seldom or never been met in person before. This is the notion of "trust by relation". Indeed, this is the way that people infer trust in real life as well (Gursel and Sen 2009).

Inferring trust from people with whom a user has no direct connection is a key research challenge in trust-aware recommender systems. It has led to the development of referral systems based on the underlying concept of a trust network, or a Web-of-Trust (Massa and Avesani 2007). A trust network infers the level of trust between two people based on the degree of connectedness between them (Massa and Bhattacharjee 2004). Trust is propagated along the network, and decays with each propagation step. In other words, a shorter distance between a source node and a destination node in the network represents a closer friendship and hence a higher level of trust between two users.

In our previous work (Wei and Fong 2010), we used a JUMP counter to symbolize the relationship between two users in the trust network. A suggested mapping from the JUMP count to the implied type and tier of relationship is shown in Table 5.

On many social networking sites, when a user adds new friends into their social network, they are asked to specify the type of relationship between them. This relationship can be mapped to one of the six tiers in Table 5. Different weights can then be set to differentiate the level of trust assigned to each relationship. The following equation shows how to compute the trust between users a and u, who are separated by $n+1$ JUMPs through n intermediate users in the social network:

$$T_{a,u} = \prod_{u}^{a} T_{a,b,c,\ldots,n,u} = T_{a,b} \times T_{b,c} \times \ldots \times T_{n,u} = T_1 \times T_2 \times \ldots \times T_5 \times T_6^{n-5}$$

where T_1 to T_6, represent the trust weight of relationship tiers 1 to 6 respectively. In this equation, the first jump from the source user a to the next user b represents a tier-1 relationship and has weight T_1. The second jump from user b to user c is a tier-2 relationship for user a and multiplies the trust by weight T_2. The same holds for jumps three, four and five. Jumps six and following are all tier-6 relationships, so each jump multiplies the trust by the same weight T_6. Since there could be multiple paths between two nodes in the network, the minimum value of $T_{i,j}$, corresponding to the shortest possible path between the nodes, is chosen as an optimistic choice (Berners-Lee et al. 2001).

Table 5 An example mapping of JUMP counts to relationships

JUMP count between two users	Tiers	Relationships
1	1	Family members (parents, children, close relatives)
2	2	Colleagues, distant relatives
3	3	Former schoolmates (used to have the same preferences)
4	4	Friends
5	5	Friends of friends
6 and more	6	Public

4.5 Trust by reputation

In addition to relationships, trust can be derived from the reputation of a user. Indeed, the recommendation of a known expert on a given topic will carry noticeable weight with another user, even if there are no social relationships between the user and the expert. Sociological definitions of trust generally have two major components (Sztompka 1999): a belief and a willingness to take action based on that belief. In a virtual community, this definition of trust translates into a belief that an information producer will create accurate information, and a willingness to commit some time to reading and processing this information.

On an online social networking site, users perceive each other by their reputations, including components such as how well-known they are among their peers in some groups, or their activity history in the social community. We call this kind of trust "trust by reputation". Since it is based on public information, such as the user's profile and their public activity history, it can be observed and evaluated to determine the level of social trust a person merits.

The idea of developing metrics to quantify trust by reputation on social networks has been studied by several researchers (Dwyer et al. 2007; Gilbert and Karahalios 2009; Golbeck 2008). For instance, Dwyer et al. (2007) applied statistical methods, including ANOVA analysis and correlation analysis, to measure the levels of trust and privacy in a comparison of Facebook and MySpace. Dependent variables such as the information shared on the users' profiles and the level of social communication were evaluated.

In our research, we assume that there are six attributes that can be observed from a user's account on Facebook and that contribute to a user's reputation. These attributes are presented in Table 6.

The trust by reputation between user *a* and user *u* can be evaluated based on the measured values of these attributes using Multiple Attribute Utility Theory (MAUT) as follows:

$$T_{a,u} = \alpha \frac{\sum PI}{totpi} + \beta \frac{\sum WTWP}{totwtwp} + \gamma \frac{\sum LF}{totlf} + \delta \frac{\sum NF}{totnf} + \varepsilon \frac{\sum T}{tott} + \theta \frac{\sum_{i=1}^{n} GIC_i}{totgic}$$

where $\alpha+\beta+\gamma+\delta+\varepsilon+\theta=1$, and where *PI* means personal information, *WTWP* means wall-to-wall posts, *LF* means the number of links among friends, *NF* means the number of friends, *T* means the number of tags between user *a* and user *u*, and *GIC* means the number of groups in common. In each term, the denominator *tot* represents the total quantity of that particular attribute of this user.

4.6 Updated score prediction and relationship strength

The score prediction equation we presented in Section 4.2 can now be updated to make use of the trust values between users that we developed in the previous two sections. In this equation, the trust $T_{a,u}$ could be either the Trust-by-Relation, the Trust-by-Reputation, or a combination of both. Compared to Section 4.2, it can be seen that the difference in the equation is to replace the similarity between users with the trust between users.

$$\widehat{r}^{T}_{ui}(t) = \bar{r}_u(t) + \frac{\sum_{j\in N_i^t\cap\tau_u^t} T_{a,u}(t)\cdot f_{uj}^{\beta}(t)\cdot r_{uj}}{\sum_{v\in N_i^t\cap\tau_u^t} T_{a,u}(t)\cdot f_{uj}^{\beta}(t)}$$

We can also define the strength of the relationship *sr*(*a*,*u*) between user *a* and user *u* in the social network as a combination of both the similarity and trust between these users:

$$sr(a,u) = w_s S(a,u) + w_t T_{a,u}$$

where the weights w_s and w_t are used to control the relative importance of similarity vs. trust between the two users, and $w_s+w_t=1$. This equation is flexible enough that one can add or remove other factors as needed by adding terms into the summation and updating the weights.

Table 6 Trust by reputation attributes

Attribute	Metric
Personal information	Completeness
Wall-to-wall posts	Quality and Count
Links among friends	Count
Number of friends	Count
Tags	Count
Groups in common	Count

4.7 Estimating the weights

A number of weights are used in CF equations, including in the trust equations we have presented previously. In many existing CF systems, these weights are defined arbitrarily by the system designer. In this research, we will rather estimate the weight values from their corresponding factors in a Facebook survey dataset.

We should begin by pointing out, however, that our selection of factors to account for and quantify in the trust equation is not definitive. In fact, there is no definite selection of trust factors in the literature. A growingly popular set of factors is the one proposed in Massa and Bhattacharjee (2004), which takes into account the number of emails exchanged between users, the number of mutual readings and comments on each other's blogs, and the number of common chats within a specified period. Some of these attributes are included in our list in Table 6. However, it should be clear that this selection is tightly dependent on the functionalities that each specific social networking site provides to its users, and thus can vary greatly from site to site. A recent study (Gilbert and Karahalios 2009) generalized the trust factors into seven categories (called time strength dimensions), and has statistically shown how useful they are as predictive variables for trust in social networks. These factors are illustrated in Fig. 3.

It can be seen from Fig. 3 that the three strongest dimensions are related to the interactive messaging activities between two users. We therefore defined some messaging and contact activities whose data can normally be extracted from a typical social network. We then generalized these activities hierarchically into sub-groups and super-groups of trust factors to obtain the hierarchical trust metrics model shown in Fig. 4. These factors contribute to the trust a user has for

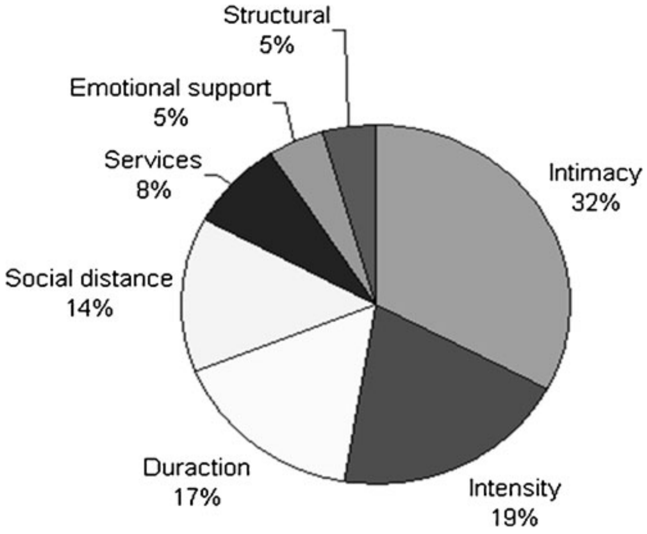
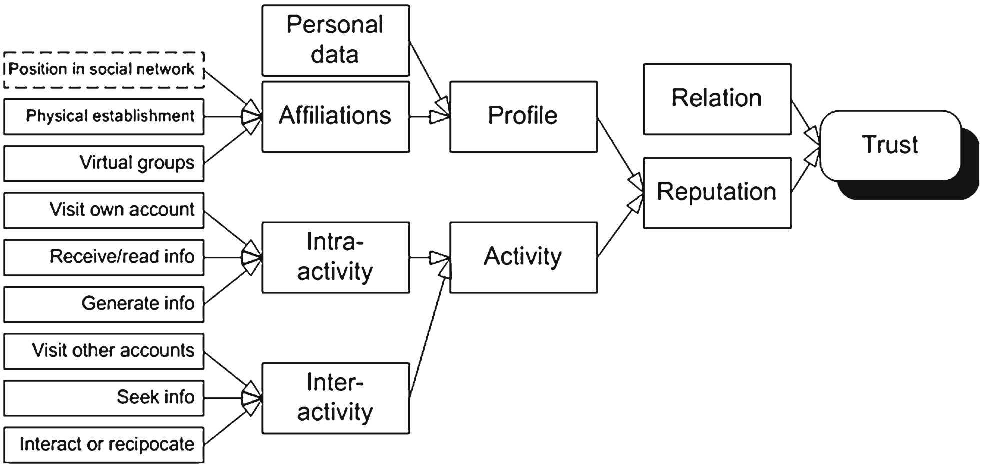


Fig. 3 The distribution of the predictive power of the seven time strength dimensions as part of the how-strong model. Source: (Gilbert and Karahalios 2009)

Fig. 4 Hierarchical trust metrics model

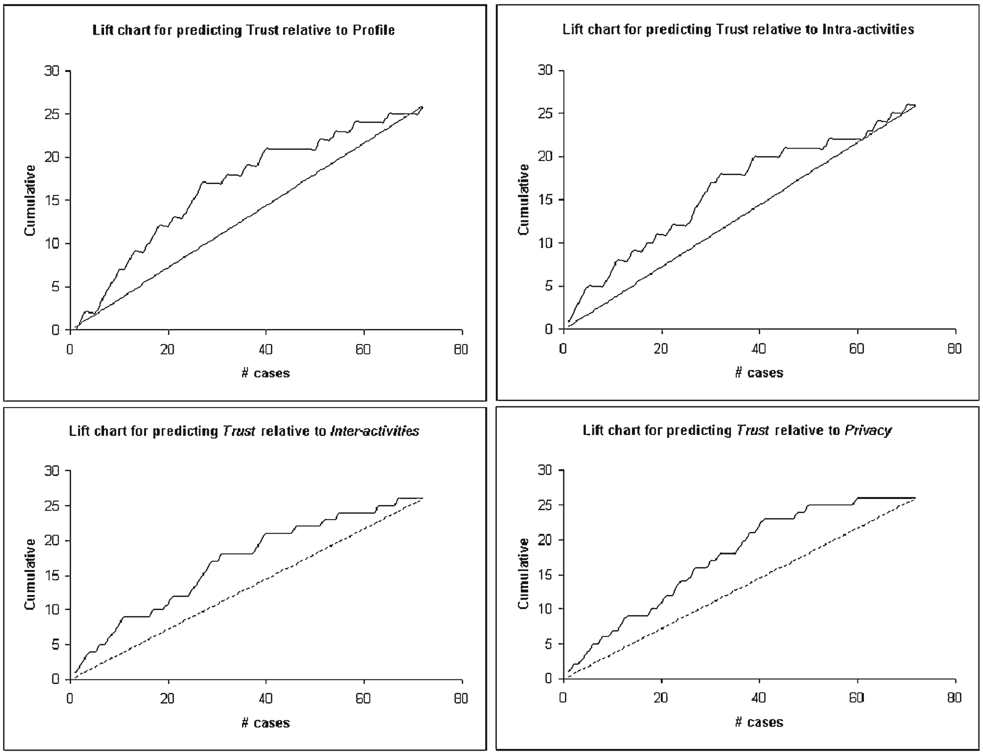


another person in the social network. At the top of the hierarchy, trust is comprised of the relation and the reputation of that person to the user. As discussed in the previous sections, the relation is computed in function of the number of jumps

Table 7 Attributes used in dataset

Category	Survey questions
Profile	3. How often do you update any aspect of your profile on Facebook
	31.Do you list your significant other as such on Facebook
	32. College students to display information
	33. High school students to display information
	34. Faculty and staff to display information
	35. Alumni to display information
	71. When browsing through profiles will you investigate profiles of people
Intra-activity	4. Investigate view profiles or pictures
	5. Investigate view groups or events
	6. Investigate view notes or posted items
	7. View news feeds personal or general
	11. Read wall posts
	15. Create groups
	16. Create events
	17. Post pictures
	18. Check out advertisements
Inter-activity	8. Search for people profiles or pictures
	9. Search for groups or events
	10. Check reply-to or send messages
	12. Make or respond to wall-posts
	13. Poke others initiate
	14. Return pokes reciprocate
Privacy	30. What is your comfort level indicating your relationship status on Facebook
	40. Do you think Facebook is invasive or not invasive into your privacy
	52. Do you adjust who can see your contact information
	53. Do you adjust what information the news feed can publish about you
	54. Do you adjust who can see your pictures
	56. Do you display your relationship status on Facebook
	62. Do you display address or contact information
	66. Do you display the interested in category on Facebook
	67. Do you feel your Facebook picture is important

Fig. 5 Accumulative gain charts of the decision tree models by different groups of factors



between the two users, while the reputation is based on the person's profile and activity history, both of which are based on other factors, as detailed in Fig. 4.

In this paper, we use Facebook as a representative case study of a social networking site. This choice naturally affects the selection of attributes we can measure. Some attributes are not possible; for example, Facebook does not keep track of who views a user's profile. On the other hand, some attributes can be further detailed; for example, we can distinguish between solo activities (e.g. posting a message on one's own wall) and interactions (e.g. posting a message on someone else's wall).

Having defined the hierarchical trust metrics model, we can now statistically estimate the relative weights of the trust factors in our model using data taken from Facebook. In this paper, we use a set of real-life survey data taken from The

Fig. 6 A two-layer model that shows interactivity relations between users and products in a site internally and also shows external relations of users and products across other websites

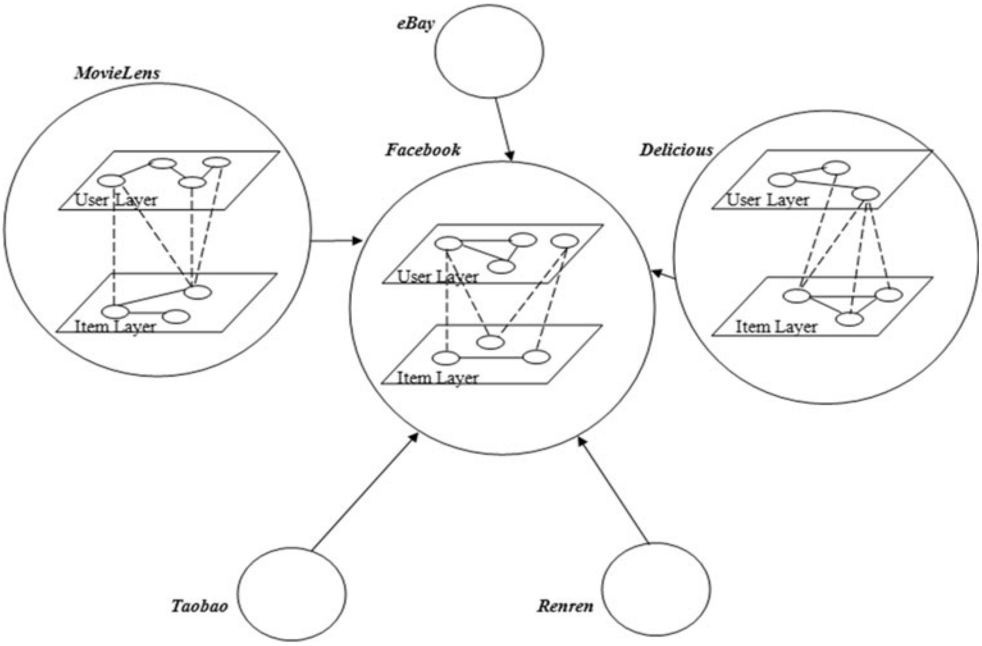


Table 8 MAE performance comparisons

Absolute difference	Percentage of predictions	Without temporal relation	With temporal relation	Performance improvement
3.0 & above	100 %	1.21537251	1.024978499	0.190394011
2.5	97.79 %	1.159584498	1.002693694	0.156890804
2.0	93.36 %	1.064817381	0.908702809	0.156114572
1.5	76.75 %	0.818810549	0.667862074	0.150948475
1.0	41.33 %	0.533974054	0.442009977	0.091964077
0.5	23.25 %	0.212732869	0.171091167	0.041641702

Facebook Project, by Jeff Ginger of the University of Illinois (Facebook Project 2011). The survey dataset was collected in April and May 2006 and gathered responses from 124 students (73 undergraduates after filtering). The dataset contains responses to questions pertaining to perceptions of trust and privacy, meeting people and relationships, identity management, messaging, pictures, and groups. We filtered the dataset to only keep questions that are relevant to our hierarchical trust metrics model. The selected questions are listed in Table 7. The responses in the dataset that were originally ordinal data were normalized. We then used the C4.5 decision tree learning algorithm (Quinlan 1993) to discover apriori association rules from the data. Four decision trees were built to classify the data to the groups of trust factors in the four categories of our hierarchy, namely Profile, Privacy, Intra-activities and Inter-activities. Each of these four decision trees can be used to predict the trust a user has for another person given that category's subset of the attributes. By observing and comparing the predictive power of the four decision trees, we can determine how much each group of trust factors contributes to the perceived trust between users, and thus quantitatively estimate the relative importance (and the weights) of these groups of trust factors.

To illustrate this process, we retained the performances of the four decision trees and represented them as Accumulative Gain Charts (Vuk and Curk 2006). The accumulative gain is a measure of the effectiveness of a predictive model calculated as the ratio between the results obtained with and without the predictive model. The diagonal lines in the graphs of Fig. 5 represent the baseline in which the model has zero predicting power. The cumulative gains, the curves over the diagonal lines in the graphs, are visual representations of the model's performance. The higher above the diagonal the curve of the gains is, the better the model is at predicting the perceived trust. In other words, the greater the area between the curve of the cumulative gain and the baseline is, the better the model is. It can be seen visually in Fig. 5 that the Profile and Privacy sets of attributes give the best predictions. We measured the area under the accumulative gain curves for the decision tree of each trust factors group, and we computed their ratio to use as weights. The ratios of area between Profiles, Privacy, Intra-activities and Inter-activities are 0.3265, 0.2449, 0.2041 and 0.2245 respectively. We should note that this result is for the Facebook set of attributes and dataset. However, the technique we developed here can be applied to measure the relative weights of any other set of trust factors available on other social networking sites.

We also applied a data mining algorithm to discover apriori association rules from the Facebook Project dataset. The program used is XLMiner (FrontlineSolvers 2011). The results allowed us to verify that the decision trees learned

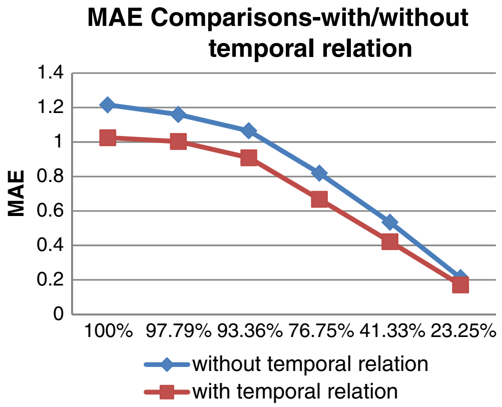


Fig. 7 MAE Performance comparisons-with/without temporal relation

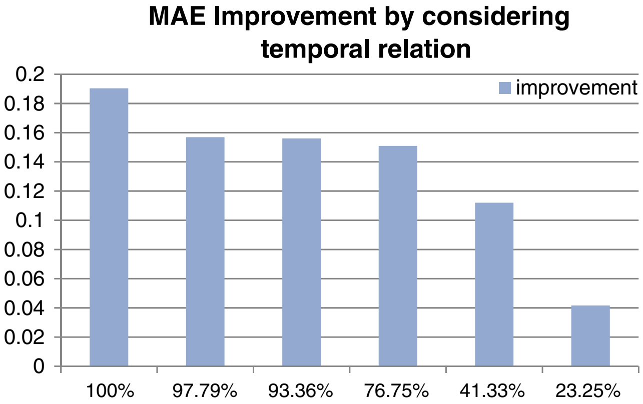


Fig. 8 MAE Improvement by considering temporal relation

Table 9 RMSD performance comparisons

Absolute difference	Percentage of predictions	Without temporal relation	With temporal relation	Performance improvement
3.0 & above	100 %	2.376742696	1.770555148	0.606187548
2.5	97.79 %	1.645643997	1.663967546	0.36128897
2.0	93.36 %	1.645643997	1.31874249	0.326901507
1.5	76.75 %	0.942466659	0.712242835	0.230223824
1.0	41.33 %	0.486284311	0.279861481	0.20642283
0.5	23.25 %	0.132169612	0.047453472	0.08471614

are consistent with our intuitive concepts about trust in social networks. To illustrate, some sample rules extracted from the data are listed below:

About the Profile attributes

- 100 % confidence: users who post college information and current faculty information will post alumni information.
- 94.12 % confidence: users who trust friends on Facebook and who posted alumni information will post other significant information.
- 93 % confidence: users who post college student information will also post other significant information.
- 90.91 % confidence: users who post current faculty information will also post other significant information.

About Intra-activities attributes

- Users who trust friends on Facebook rank the following actions in terms of relevance: read messages on wall, be aware of profiles or pictures.
- 90 % confidence: users who trust friends on Facebook, investigate profiles, and create groups, will also create events.

About Inter-activities attributes

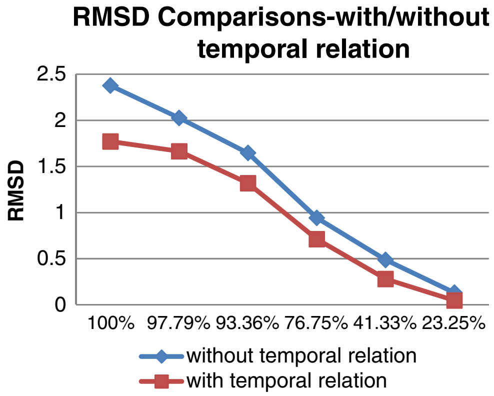


Fig. 9 RMSD Performance comparisons-with/without temporal relation

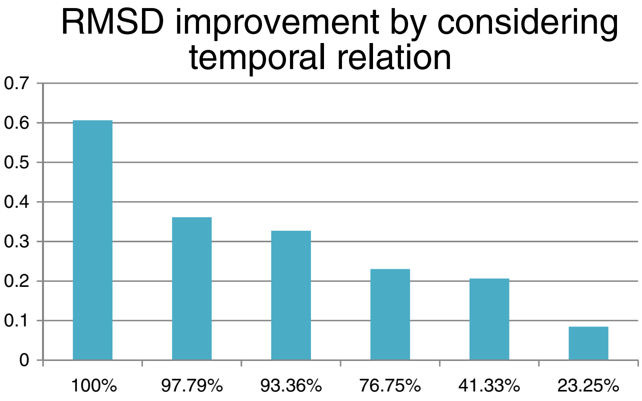


Fig. 10 RMSD Improvement by considering temporal relation

- 80 % confidence: a relation is deemed to be trusted when the following three actions are observed: a message is replied, a wall post is responded to, and poke is reciprocated.

About the Privacy attributes

- Privacy's relation to trust is correlated with a user's tendency to investigate other people's profiles. When this behavior is observed, the user becomes careful to set their own privacy options to control who can see their profile.
- 56.25 % confidence: users feel more comfortable trusting people who have set their privacy settings on their profiles.

4.8 Our complete model

Our proposed model is a general model that represents the inter-activities and relationships between Facebook and other websites. It is illustrated in Fig. 6. Each circle in that figure represents one individual website, which can be viewed as two layers. The upper layer is the user layer; it contains the profiles of the users subscribed to the site and their connections to each other. The information on the users' profiles is used to compute the similarity between users. Moreover, the social connections and interpersonal activities on that layer are used to compute the trust metrics.

Table 10 MAE performance comparisons

Absolute difference	Percentage of predictions	MAE without trust factor	MAE with trust factor
3.0 & above	100 %	1.315911	1.298544
2.5	98.52 %	0.945338	0.908700
2.0	96.62 %	0.894649	0.889889
1.5	76.14 %	0.574617	0.667867
1.0	42.06 %	0.253505	0.533972
0.5	12.55 %	0.099312	0.212729

Table 11 RMSD performance comparisons

Absolute difference	Percentage of predictions	RMSD without trust factor	RMSD with trust factor
3.0 & above	100 %	1.633257	1.568305
2.5	98.52 %	1.364072	1.318761
2.0	96.62 %	1.232829	1.260211
1.5	76.14 %	0.847439	0.712253
1.0	42.06 %	0.412900	0.486299
0.5	12.55 %	0.112386	0.344925

The bottom layer is the item layer, and contains data about the different items reviewed on the site. The information from this item layer is used to compute the similarity between items. Inter-activities between the users in the upper layer and the items in the lower layer do exist. For example, a user may trade an item via an E-marketplace, or write their opinion about an item on a review site. In our example of Fig. 6 Facebook is the central site that the other websites connect onto.

The final predicted score that a user would give to an item in our proposed model, P_{final}, is computed as a combination of the two scores predicted in the previous sections: the score predicted by considering similar ratings (either ratings of the same item by similar users or ratings by the same user of similar items) in Section 4.2, and the score predicted by considering the trust between users in Section 4.6:

$$P_{final} = w_S \widehat{r}^S_{ui}(t) + w_T \widehat{r}^T_{ui}(t)$$

where w_S and w_T are parameters that can be adjusted to balance the influence of each score in the final prediction, with the constraint that $w_S+w_T=1$.

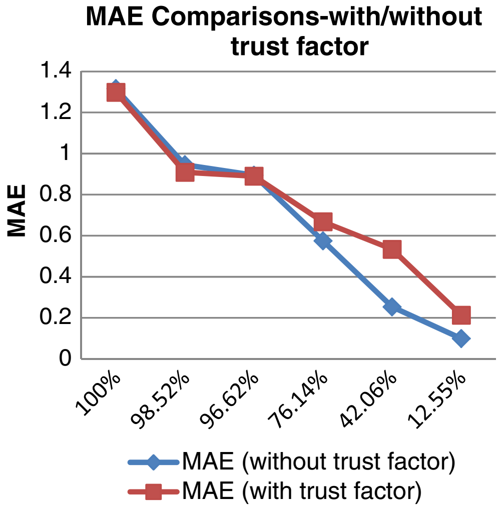


Fig. 11 MAE Performance comparisons-with/without trust

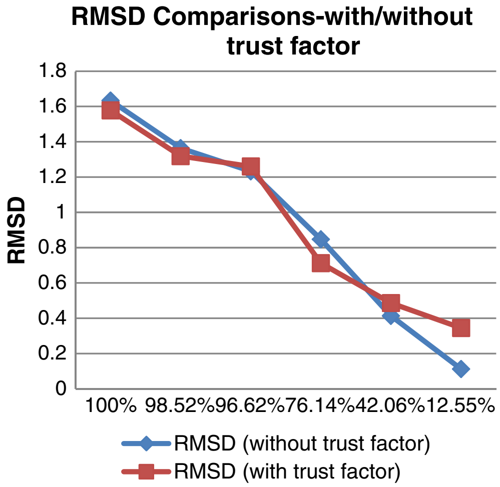


Fig. 12 RMSD Performance comparisons-with/without trust

5 Empirical analysis

5.1 Experimental setup and datasets

In order to evaluate the quality of our MCFTN algorithm, we compared the results it generates to those obtained on the same dataset with the traditional Collaborative Filtering method using UPCC (Pearson Correlation) and TUPCC (Pearson Correlation plus Temporal Effect). For our algorithm, we considered versions of the equation using similarity, trust metrics, and temporal relations.

Three different datasets were used in these experiments, namely movie ratings from the MovieLens site (GroupLens 2011), Facebook data from Networking Group (Networking Group 2011), and suggested bookmarks from the social bookmarking site Delicious. MovieLens is a popular Web-based movie recommender system. It contains 100,000 ratings (on a 1 to 5 scale) from by 943 users on 1,682 movies. Each user on the site has rated at least 20 movies and in one case as many as 737 movies, with the average being around 106 movie ratings per user. The density of the user-item matrix is:

$$\frac{100000}{943 \times 1682} = 6.30\%$$

Facebook is a social networking service launched in February 2004, operated and privately owned by Facebook Inc. As of July 2011, Facebook has more than 800 million active users. Users must register before using the site, after which they may create a personal profile, add other users as friends, and exchange messages, including automatic notifications when they update their profile. The Facebook dataset used in these experiments was obtained from the Networking Group, a research group at University of California, Irvine. The dataset lists users along with profile, privacy, intra-activity and inter-activity attributes. Finally, Delicious is a social networking, bookmarking, and tagging website and the dataset

Table 12 Impact of parameter on MAE

Absolute difference	Percentage of predictions	$\alpha=\beta=10^{-5}$	$\alpha=\beta=10^{-6}$	$\alpha=\beta=10^{-7}$	$\alpha=\beta=10^{-8}$	$\alpha=\beta=10^{-9}$	$\alpha=\beta=10^{-10}$	$\alpha=\beta=10^{-11}$
3.0 & above	100 %	1.14648445	1.083987356	1.026857291	1.025249771	1.025000987	1.024978499	1.024976274
3.0	98.52 %	1.122976477	1.062144565	1.008432466	1.002966998	1.00271635	1.002693694	1.002691452
2.5	94.09 %	1.020964329	0.970951368	0.914711641	0.908988974	0.908726532	0.908702809	0.908700461
2.0	92.99 %	1.003774011	0.955531343	0.897226976	0.890409354	0.889938258	0.889893391	0.889888926
1.5	77.49 %	0.714091276	0.746637612	0.676672371	0.668491225	0.66792591	0.66787207	0.667866712
1.0	64.21 %	0.648415769	0.569290663	0.539586216	0.534215487	0.533994324	0.533974054	0.533972046
0.5	38.01 %	0.100755717	0.272393936	0.225265267	0.213140726	0.21276711	0.212732869	0.212729475

obtained from them lists 2,000 users. It was released at the 2nd International Workshop on Information Heterogeneity and Fusion in Recommender Systems (HetRec 2011) at the 5th ACM Conference on Recommender Systems (RecSys 2011). We join the three datasets by assuming that users with the same user ID in all sets are the same individuals.

The algorithms were tested by withholding a test set from the data, and predicting the rating values users should give to items in that set. We evaluated the results by computing the Mean Absolute Error (MAE) and the Root Mean Square Deviation (RMSD) between the predicted rating values computed by our system and the actual rating values assigned by the users. The MAE and RMSD are defined as:

$$MAE = \frac{\sum_{u,i} \left| r_{u,i} - \widehat{r}_{u,i} \right|}{N}$$

$$RMSD\left(r_{u,i}, \widehat{r}_{u,i}\right) = \sqrt{\frac{\sum_{u,i} \left(r_{u,i} - \widehat{r}_{u,i}\right)^2}{N}}$$

where $r_{u,i}$ denotes the rating that user u gave to item i, $\widehat{r}_{u,i}$ denotes the predicted rating that user u would give item i according to the recommendation algorithms, and N denotes the number of test ratings.

5.2 Evaluation

For the first experiment, we compare the performance of our MCFTN algorithm with temporal relationships but without using the trust metrics to that of the user-based PCC (UPCC) algorithm, which uses neither temporal relationships nor trust. This experiment will demonstrate the benefit of using temporal relationships.

Table 8 and Figs. 7 and 8 give the MAE results of both systems, while Table 9 and Figs. 9 and 10 give the RMSD results of the same experiment. Both sets of results show clearly that the prediction quality is improved when using our algorithm with temporal relations. To further analyze the results, we sorted the predictions by the absolute value of the difference between the predicted rating and actual rating, and grouped them by multiples of 0.5. We then evaluated the proportion of predictions in each set, and the MAE and RMSD of each set separately. For example, in Table 8, we see that 23.25 % of the predictions have an absolute difference of 0.5 or less with their actual values, and our method gives a MAE improvement of 0.04 in that subset. Likewise, 93.36 % of predictions have an absolute difference of 2.0 or less with their actual values, and our method gives an improvement of 0.16 in that subset. The fact the improvement is minimal in the former subset shows that, in the best cases, ignoring the temporal relation can give results that are

Table 13 Impact of parameter on RMSD

Absolute difference	Percentage of predictions	$\alpha=\beta=10^{-5}$	$\alpha=\beta=10^{-6}$	$\alpha=\beta=10^{-7}$	$\alpha=\beta=10^{-8}$	$\alpha=\beta=10^{-9}$	$\alpha=\beta=10^{-10}$	$\alpha=\beta=10^{-11}$
3.0 & above	100 %	2.087581118	1.927752952	1.768934756	1.768903346	1.770384703	1.770555148	1.770572413
3.0	98.52 %	1.954914847	1.822342862	1.66896024	1.662303371	1.663795824	1.663967546	1.663984941
2.5	94.09 %	1.572029784	1.484570761	1.323970134	1.317	1.318562686	1.31874249	1.318760703
2.0	92.99 %	1.514772602	1.440850108	1.271544449	1.259506059	1.260119205	1.260202362	1.260210894
1.5	77.49 %	0.792117827	0.929020129	0.725853339	0.711407271	0.712143046	0.712242835	0.712253073
1.0	64.21 %	0.640547223	0.551078019	0.489853513	0.48485129	0.486135385	0.486284311	0.486299406
0.5	38.01 %	0.080630397	0.241626945	0.141580431	0.129748781	0.13191803	0.132169612	0.144924588

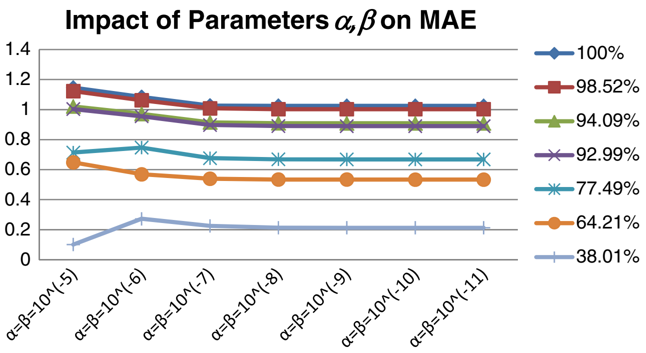


Fig. 13 Impact of parameters α, β on MAE

almost as good as using it. However, over the entire dataset, the predictions made by the algorithm with temporal relations generate a consistently lower error rate.

For our second experiment, we computed the score predictions of our algorithm using temporal relations and trust factors. We compared these results to those using only temporal relations, which we presented in Tables 8 and 9. This test illustrates the benefits of using trust in the system. As before, we partitioned the results by their absolute difference to the actual values. The results are presented in Tables 10 and 11 and in Figs. 11 and 12. We can see from these results that, above the 96.62 % contribution threshold, the algorithm with trust factors has lower MAE and RMDS compared to the algorithm without trust factors. Below that contribution threshold, the MAE and RMDS disagree on which algorithm is better. However, the results confirm that when handling the entire dataset, using trust factors is better.

5.3 Influence of temporal parameters

Section 4.2 introduced two decay rate control parameters. Parameter α controls the decay rate of the temporal relations, while β controls the decay rate of each rating's weight when making score predictions. We experimented with different values for these parameters. Tables 12 and 13 and Figs. 13 and 14 report the results we obtained in these experiments.

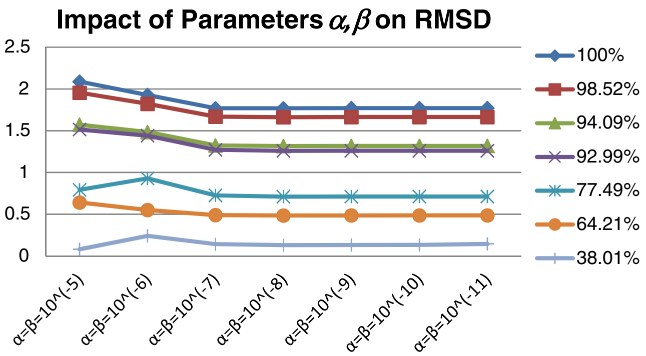


Fig. 14 Impact of parameters α, β on RMSD

Table 14 Impact of parameters on MAE

Absolute difference	Percentage of predictions	w_S=0; w_T=1	w_S=0.1; w_T=0.9	w_S=0.2; w_T=0.8	w_S=0.3; w_T=0.7	w_S=0.4; w_T=0.6	w_S=0.5; w_T=0.5	w_S=0.6; w_T=0.4	w_S=0.7; w_T=0.3	w_S=0.8; w_T=0.2	w_S=0.9; w_T=0.1	w_S=1; w_T=0
3.0 & above	100 %	1.30378937	1.173591423	1.198705859	1.286049621	1.27322296	1.26966209	1.24885992	1.23741086	1.31596184	1.214512743	1.230604112
3.0	99.26 %	1.33027023	1.122511416	1.10687468	1.25907399	1.22044745	1.2676964	1.2275738	1.20696588	1.29013471	1.18231701	1.212008826
2.5	98.90 %	1.235422082	1.108800193	1.09343559	1.24641022	1.20798761	1.2575336	1.2162125	1.1956421	1.27932715	1.17154914	1.209068561
2.0	94.46 %	1.181125492	1.008232246	0.99950791	1.1308462	1.08558092	1.1507225	1.1124788	1.08244852	1.17067934	1.07069472	1.165541283
1.5	83.39 %	1.004646853	0.829187531	0.81698956	0.94503537	0.87410168	0.9320586	0.9206205	0.87863408	0.97866834	0.87573962	0.904182601
1.0	53.14 %	0.592904195	0.487127636	0.47728797	0.55213261	0.48255769	0.5296071	0.5170704	0.46560225	0.54743362	0.48174106	0.717895701
0.5	27.03 %	0.37821407	0.221227551	0.20650889	0.23393381	0.18356666	0.2121562	0.1725244	0.1322174	0.18214929	0.11077023	0.530855652

Table 15 Impact of parameters on RMSD

Absolute difference	Percentage of predictions	w_S=0; w_T=1	w_S=0.1; w_T=0.9	w_S=0.2; w_T=0.8	w_S=0.3; w_T=0.7	w_S=0.4; w_T=0.6	w_S=0.5; w_T=0.5	w_S=0.6; w_T=0.4	w_S=0.7; w_T=0.3	w_S=0.8; w_T=0.2	w_S=0.9; w_T=0.1	w_S=1; w_T=0
3.0 & above	100 %	1.686883042	1.51671616	1.499058644	1.619963517	1.603814289	1.689151917	1.57601789	1.58445072	1.65448384	1.546152838	1.569479603
3.0	99.26 %	1.673105264	1.421571121	1.40424997	1.58228125	1.58521953	1.606357	1.5508562	1.56620177	1.62195549	1.52234407	1.545191978
2.5	98.90 %	1.578516031	1.393749621	1.377155299	1.5595422	1.5635601	1.5907762	1.531672	1.54758049	1.60472909	1.50480138	1.543243953
2.0	94.46 %	1.406954641	1.238065622	1.22602775	1.37659258	1.36781573	1.4357081	1.378929	1.3749278	1.44248417	1.35859677	1.494514861
1.5	83.39 %	1.293698764	0.994087807	0.98324354	1.13577102	1.07420955	1.1323314	1.1274049	1.10279337	1.19559713	1.10282776	1.312109849
1.0	53.14 %	0.977522551	0.575998201	0.57050517	0.66015423	0.59032805	0.6435257	0.6497302	0.5998899	0.68171059	0.63635439	0.951514735
0.5	27.03 %	0.543849519	0.259973046	0.25037518	0.28217485	0.21431934	0.2686548	0.2265096	0.17270579	0.2564229	0.17903991	0.663184851

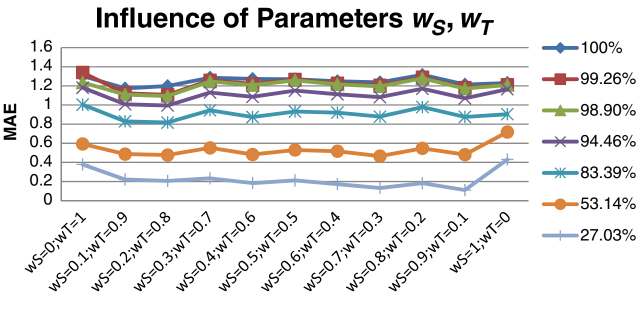


Fig. 15 Impact of parameters w_S, w_T on MAE

From Table 12 and Fig. 13, it can be seen that the system reaches its minimal MAE value when $\alpha=\beta=10^{-11}$. Meanwhile, Table 13 and Fig. 14 show that the minimal RMSD is reached when $\alpha=\beta=10^{-8}$. A good compromise value seems to be around $\alpha=\beta=10^{-9}$. In all cases, these low values indicate that, the slower the temporal decay is, the more precise its influence on the current recommendations is. In other words, more recent ratings are more accurate predictors of current ratings.

5.4 Influence of weight parameters

The final predicted score of our proposed model, P_{final}, presented in Section 4.8, combines the predicted scores computed by our algorithm using trust and using similarity. This final equation uses two weight parameters w_S and w_T to balance the influence of each of these predictions in the final result. For our last experiment, we tried every increment of 0.1 for the values of these parameters to illustrate the impact of this balance. Tables 14 and 15 and Figs. 15 and 16 report the results obtained in these experiments. It can be seen that the minimum error, both in MAE and RMSD, occurs at w_S= 0.2 and w_T=0.8. This means that the opinion of trusted individuals is the determinant factor in the prediction. However, this needs to be tempered somewhat by the ratings given by the user to similar items (showing that there is room for individuality in this model) and by similar users to this item (allowing variance in opinion based on the specific item being rated).

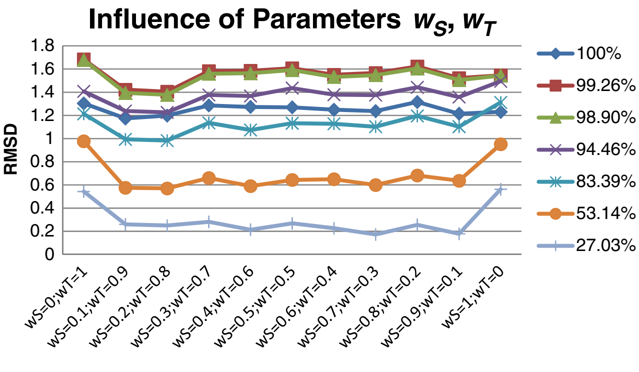


Fig. 16 Impact of parameters w_S, w_T on RMSD

6 Conclusion

Recommendation services need to have a good understanding of individual users in order to offer relevant recommendations based on their preferences. Although a lot of previous work has been done on collaborative filtering algorithms and their variants, this work was based on the Web 1.0 platform. In this paper, we proposed a new algorithm to implement a recommendation service on a Web 2.0 platform, such as a social networking site. Web 2.0 offers a richer source of information than Web 1.0 about users' preferences, interactions, and relationships, but also comes with a host of new challenges related to the availability of this information.

The new algorithm we proposed is called Multi-Collaborative Filtering Trust Network. In this paper, we developed its mathematical framework and showed how it combines a collaborative filtering algorithm with trust network inference, temporal relations, and multiple online data sources. We implemented the algorithm, trained it using datasets downloaded from three different social networks, and tested it by predicting the rating scores that the users in these datasets would give to various items. The results we presented show that the combination of factors we used, with proper weighting to balance them, yields a good improvement in the quality of the recommendations. Our findings and methodology could be applied in the future to develop the next generation of marketing and recommendation systems.

References

Berners-Lee, T., Hendler, J., & Lassila, O. (2001). The semantic web. Scientific American.

Brenner, A., Pradel, B., Usunier, N., & Gallinari, P. (2010). Predicting most rated items in weekly recommendation with temporal regression. Proceedings of the 2010 ACM Conference on Recommender Systems (RecSys), pp. 24–27.

Campos, P. G., Bellogin, A., Diez, F., & Chvarriaga, J. E. (2010). Simple time-biased KNN-based recommendations. Proceedings of the 2010 ACM Conference on Recommender Systems (RecSys), pp. 20–23

Debnath, S., Ganguy, N., & Mitra, P. (2008). Feature weighting in content based recommendation system using social network analysis. 17th International Conference on WWW, 1041–1042.

Desarkar, M. S., Sarkar, S., & Mitra, P. (2010). Aggregating preference graphs for collaborative rating prediction. Proceedings of the 2010 ACM Conference on Recommender Systems (RecSys), 21–28.

Dwyer, C., Hiltz, S. S. R., & Passerini, K. (2007). Trust and privacy concern within social networking sites: a comparison of Facebook and MySpace. Proceedings of the Thirteenth Americas Conference on Information (AMCIS 2007), 339–351.

Facebook Project Research & Resource. http://thefacebookproject.com/resource/datasets.html. Accessed 18 November 2011.

FrontlineSolvers XLMiner. http://www.solver.com/xlminer/. Accessed 18 November 2011.

Gilbert, E., & Karahalios, K. (2009). Predicting tie strength with social media. Proceedings of the 27th International Conference on Human Factors in Computing Systems (CHI 09), 211–220.

Golbeck, J. (2008). Weaving a web of trust. *AAAS Science Magazine, 321*(5896), 1640–1641.

GroupLens Research. http://www.grouplens.org/. Accessed 18 November 2011.

Gursel, A., & Sen, S. (2009). Producing timely recommendations from social networks through targeted search. Proceedings of the 8th International Conference on Autonomous Agents and Multi-agent Systems, 2, 805–812.

Hung, L.-P. (2005). A personalized recommendation system based on product taxonomy for one-to-one marketing online. *International Journal of Expert Systems with Applications, 29*(2), 383–392.

Kazienko, P., & Musial, K. (2006). Recommendation framework for online social networks. *Advances in Web Intelligence and Data Mining, Springer, 23*, 111–120.

Kitisin, S., & Neuman, C. (2006). Reputation-based trust-aware recommender system. Securecomm and Workshops, pp. 1–7.

Korvenmaa, P. (2009). The growth of an Online Social Networking Service Conception of Substantial Elements. Master Thesis, Teknillinen Korkeakoulu, Espoo.

Liu, N. N., Zhao, M., Xiang, E., & Yang, Q. (2010). Online evolutionary collaborative filtering. Proceedings of the 2010 ACM Conference on Recommender Systems (RecSys), pp. 95–102.

Massa, P., & Avesani, P. (2007). Trust-aware recommender systems. Proceedings of the 2007 ACM conference on Recommender systems (RecSys), pp. 17–24.

Massa, P., & Bhattacharjee B. (2004). Using trust in recommender systems: An experimental analysis. Second International Conference in Trust Management (iTrust 2004), Lecture Notes in Computer Science, Springer, 2995, pp. 221–235.

Massari, L. (2010). Analysis of MySpace user profiles. *Information Systems Frontiers Special Issue on Ethics and Information Systems, 12*(4), 361–367.

Networking Group Wiki Page. http://odysseas.calit2.uci.edu/doku.php/public:online_social_networks. Accessed 18 November 2011.

Page, L., Brin, S., Motwani, R., & Winograd, T. (1998). The page rank citation ranking: Bringing order to the web. Technical report, Stanford, USA.

Quinlan, J. R. (1993). *C4.5: Programs for machine learning*. Morgan Kaufmann Publishers.

Rendle, S., & Schmidt-thieme, L. (2010). Factorization models for context-/time-aware movie recommendations encoding time as context. Proceedings of the Workshop on Context Aware Movie Recommendation, pp. 1–6

Sandholm, T., Ung, H., Aperjis, C., & Huberman, B. A. (2010). Global budgets for local recommendations. Proceedings of the 2010 ACM Conference on Recommender Systems (RecSys), pp. 13–20.

Song, J.-G., & Kim, S. (2009). A study on applying context-aware technology on hypothetical shopping advertisement. *Information Systems Frontiers Special Issue on Intelligent Systems and Smart Homes, 11*(5), 561–567.

Sun, J., Yu, X., Li, X., & Wu, Z. (2008). Research on trust-aware recommender model based on profile similarity. International Symposium on Computational Intelligence and Design (ISCID 08), pp. 154–157.

Sztompka, P. (1999). *Trust: A sociological theory*. Cambridge: Cambridge University Press.

Vuk, M., & Curk, T. (2006). ROC curve, accumulative gain chart and calibration plot. *Metodolo shi zvezki, 3*(1), 89–108.

Wei, C., & Fong, S. (2010). Social network collaborative filtering framework and online trust factors: a case study on Facebook. 5th International Conference on Digital Information Management.

Wu, Z., Yu, X., & Sun, J. (2009). An improved trust metric for trust-aware recommender systems. First International Workshop on Education Technology and Computer Science (ETCS 09), pp. 947–951.

Chen Wei was born in China in 1987. She received her B.S. Degree from the School of Electronics Information Engineering of Beijing Jiao Tong University (China P.C.R.) in 2009, and majored in telecommunication. In 2012 she received her M.S. Degree from the Faculty of Science and Technology of the University of Macau (Macau S.A.R.). She specialized in e-commerce technology, Faculty of Science and Technology. Chen did a technical traineeship in Porto Alegre (Brazil) in 2010. In 2011, she was invited to visit the University of Coimbra (Portugal) as a research assistant working in the area of Named-Entity Recognition. Since 2012, Chen works at Lenovo Headquarter in Beijing (China P.R.C.). Her current research interests include Social Media ROI System.

Richard Khoury received his Bachelor's Degree and his Master's Degree in Electrical and Computer Engineering from Laval University (Québec City, QC) in 2002 and 2004 respectively, and his Doctorate in Electrical and Computer Engineering from the University of Waterloo (Waterloo, ON) in 2007. He is an Assistant Professor, tenure track, in the Department of Software Engineering at Lakehead University (Canada). He has published so far a total of 9 articles in international journals and IEEE Transactions, as well as 14 papers in peer-reviewed national and international conference proceedings. Dr. Khoury's main research area is natural language processing, and his other research interests include data mining, knowledge management, machine learning, computer vision, and artificial intelligence. Dr. Khoury was the guest editor for a special issue of the Journal of Emerging Technologies in Web Intelligence dedicated to the growing field of Web Data Mining. He co-chaired special sessions on "Intelligent Data Analysis" and on "Intelligent Knowledge Management" in the International Conference on Autonomous and Intelligent Systems. His research projects have received finalist rankings in competitions organized by the Ontario Centres of Excellence and Canada Health Infoway.

Simon Fong graduated from La Trobe University in Australia, with a First Class Honours BEng. Computer Systems degree and a PhD. Computer Science degree in 1993 and 1998 respectively. Simon is now working as an Assistant Professor in the Computer and Information Science Department of the University of Macau. He is also one of the founding members of the Data Analytics and Collaborative Computing Research Group in the Faculty of Science and Technology. Before his academic career, Simon took up various managerial and engineering posts, such as being a systems engineer, IT consultant, integrated network specialist, and e-commerce director in Melbourne, Hong Kong. and Singapore. Some companies that he worked at before include Hong Kong Telecom, Singapore Network Services, AES Pro-Data, and the United Overseas Bank in Singapore. Dr. Fong has published over 150 peer-reviewed international conference and journal papers, mostly in the area of E-Commerce and Data mining.