On enhancing recommender systems by utilizing general social networks combined with users goals and contextual awareness

Rana Chamsi Abu Quba

ON ENHANCING RECOMMENDER SYSTEMS BY UTILIZING GENERAL SOCIAL NETWORKS COMBINED WITH USERS GOALS AND CONTEXTUAL AWARENESS

Order Number: 61 Year: 2015

University of Lyon

PhD Thesis

ON ENHANCING RECOMMENDER SYSTEMS BY UTILIZING GENERAL SOCIAL NETWORKS COMBINED WITH USERS GOALS AND CONTEXTUAL AWARENESS

A dissertation submitted for the degree of

Doctor of Philosophy (PhD) at University of Lyon

Discipline: Computer Science
Ecole Doctorale: InfoMaths

Presented and publicly defended by

Rana CHAMSI ABU QUBA

on 18 May 2015

in front of a jury composed of:

Directors:

Salima Hassas Professor,	Université Claude Bernard Lyon I, France
Usama Fayyad,	University of Michigan, U.S., Barclays Group, PLC, London

Reviewers:

Boughanem Mohand, Professor,	University Paul Sabatier, France
Marc El Beze, Professor,	Université d'Avignon, France

Examiners:

Sylvie Calabretto, Professor,	INSA de Lyon, France
Imad Saleh, Professor,	Université Paris 8, France
Hocine Cherifi, Professor,	Université de Bourgogne, Dijon, France

My Lord...

You left me with no excuse not to succeed...

Thanks...

I Pray that you accept this work...

Ameen...

CONTENTS

LIST OF NOTATION

GPSN: General purpose social network.

DBSN: Domain based social network.

MAE: Mean Absolute Error.

NMAE: Normalized MAE.

RMSE: Root Mean Squared Error.

NRMSE: Normalized RMSE.

iSoNTRE: Social Network Transformer into recommendation engine.

RS: Recommendation system.

SLiR: Short Life Resources.

CF: Collaborative filtering.

CBF: Content Based Filtering.

SN: Social Network.

M: items or resources.

***U*:** users.

$\boldsymbol{M_u}$**:** The items rated or bought by user u.

$\boldsymbol{U_m}$**:** The users who rated or bought an item m.

$\widehat{\boldsymbol{R}}_{\boldsymbol{UM}}$**:** The rating matrix

$\boldsymbol{r_{um}}$**:** The rating of user u to item m.

$\boldsymbol{r_u}$**:** The vector of all rating provided by user u.

$\boldsymbol{r_m}$**:** The vector of all ratings provided to item m.

$\bar{\boldsymbol{r}}_{\boldsymbol{u}}$**:** The average of user u ratings.

$\bar{\boldsymbol{r}}_{\boldsymbol{m}}$**:** The average of the items m ratings.

$\mathbf{P_{u,i}}$**:** Prediction of how much the item i is interesting to user u

$\boldsymbol{S(u,v)}$ **:** Pearson similarity between u and v

$\boldsymbol{w(x,y)}$**:** Cosine similarity between x and y

VSM: Vector Space Model.

D: The set of document to be recommended in content based.

T: The dictionary or the words of the domain.

$\boldsymbol{w_{kj}}$**:** The weight of the term t_k in the document d_j.

TF-IDF: Term Frequency Inverse Document Frequency.

DCG: Discount Cumulative Gain.

nDCG: Normalized DCG.

CTR: Click through rate.

SM: Social Media.

WoM: Word of Mouth.

eWoM: Electronic WoM.

SN: Social Network.

$\boldsymbol{G(U,F)}$: The graph representation of a social network, U is the set of users and F the set of friendship links.

$\boldsymbol{F_u^+}$, $\boldsymbol{d_u^+}$**:** The users trusted by user u.

$\boldsymbol{F_u^-}$, $\boldsymbol{d_u^-}$ **:** The users who trust the user u.

$\boldsymbol{S}$: The matrix representation of the social graph.

MF: Matrix Factorization.

AP: Average precision.

AR: Average recall.

NN: Nearest Neighbor method.

ODP: The Open Directory Project, or Wikipedia enclopedia.

BK: Blue Kangaroo.

BKT: Blue Kangaroo tree

$\vec{\boldsymbol{G}}_{\boldsymbol{u}}$**:** **G**lobal profile of user u.

$\vec{\boldsymbol{D}}$**:** Concepts of domains.

$\vec{\boldsymbol{D}}_{\boldsymbol{i}}$**:** Domain i contains X concepts.

$\overrightarrow{D_i^X}$:	The user profile over domain i.
$\vec{U}_{1\to R^X}$:	The user vector of concepts.
$\vec{M}_m$:	Global profile of item m
$\overrightarrow{M_i^X}$:	The item profile over domain i.
$\vec{M}_{1\to R^X}$:	The item vector of concepts.
$U_{U\times f}$:	User profile matrix
$M_{U\times f}$:	Item profile matrix
SAW:	Simple adaptive weighting method.
$\widehat{R_{UM}^i}$:	Recommendation matrix over domain i
CFR:	Collaborative filtering recommender.
DR:	Domain recommender.
SR:	Social recommender.
CR:	Circle recommender.
UBCF:	User based collaborative filtering.
IBCF:	Item based collaborative filtering.
SVD:	Singular value decomposition
Z-s, Zs:	Z-score
Nor:	Normalized.

GLOSSARY

GPSN:

Social networks where users can do different action, like, post, comment, example: Facebook, Twitter...

DBSN:

Social networks where users can do a small kind of actions over a specific domain, like Netflix (films).

Collaborative Filtering:

Techniques in recommendation systems based on leveraging the ratings of similar users/items in a user/item population.

Content Based recommendation:

Techniques in recommendation systems based on finding similar items to those liked by users, based on items' keywords and textual similarity.

iSoNTRE:

A social machine of recommendation based on transforming GPSNs into a recommendation engine

iAmélie:

A hybrid recommender based on reflection in recommendation systems.

SLiR:

Items or articles that do not live for a long time, like news in a news site or offers in a commercial site.

Offer:

A special discount available only for short time over a product. Offer is a SLiR kind of items.

Blue Kangaroo Tree:

A world knowledge that contains the categories and relation between them in the commercial domain, similar to e-bay tree.

$\vec{G}_u$ Global Profile of user u:

The global profile is a bag of meaningful words with the frequency of each word.

$\vec{M}_m$ Global profile of item m:

The global profile is a bag of meaningful words with the frequency of each word.

Explicit feedback:

The ratings (1..5), like and dislike provided explicitly by user.

Implicit feedback:

The information extracted from user actions like navigation or time spent on a page.

Accuracy:

To what end the prediction about how a matched item is interesting to a user is correct.

Avatars/Sirens:

Created Twitter intelligent accounts that behave like humans.

Tweets:

Short messages limited to 140 characters and are mostly short status updates of what users are doing, where they are... they are used on Twitter social network.

Z-Score:

Z is a normalization method, its value represents the distance between the row score and the population mean in units of the standard deviation.

MAE:

Evaluation metric which measures the average absolute deviation between a predicted rating and the user's true rating.

RMSE:

Evaluation metric that calculates the square root of the mean/average of the square of all of the error

DR Domain recommender:

Recommending based on users similar to the main user in a specific domain.

Cold Start Problem:

The problem of how to recommend to new user, or new item where there is not yet enough ratings to be recommended.

SR Social recommender:

Recommending based on user's relations in a social graph.

CR Circle recommender:

Recommending based on user's relations who are similar to the main user in a specific domain.

A friend's advice:

A service that provides the user with possibility to send request to his friends to help him finding a particular ite

CHAPTER 1: INTRODUCTION

"We are drowning in information but starved for knowledge."

John Naisbitt, Megatrends

1. Motivation:

Each day we are surrounded by any number of decisions to make. Which book should I read next? Which film do I want to watch tonight or go to over the weekend? Increasingly, we use the web and online resources to help us make a decision. As our decision making is transported and conducted in the online sphere, the use of recommendation systems has become essential in daily life. At the same time, social networks have become an indispensable part of this process; people from all over the globe use them on a daily basis to get input from people and sources they trust.

When people spend time on social networks, they leave valuable information about themselves. This has attracted the attention of researchers and professionals from numerous academic and commercial fields. As recommendations are one domain that have witnessed widespread change due to social networks, there is an obvious interest in the field of social recommenders studies.

However, in this area's literature we found that many social recommenders were evaluated through social networks like Epinions, Flixter and other types of domains based on recommender social networks, which tend to be composed of users, items, ratings and relations. Such solutions cannot be extended directly to General Purpose Social Networks (GPSNs) like Facebook and Twitter, which are open social networks where users can complete a variety of useful actions that aid in recommendation. But since users can't rate items, there is a lack of information to be used in recommender systems. Thus, solutions to the GPSN dilemma are oriented towards enhancing recommendation of resources that are already shared, based on available information, in these social networks. Moreover, evaluations are based on the known metrics like MAE, and RMSE. This does not guarantee user satisfaction or provide quality recommendations.

As one delves deeper into the recommendation world, one finds that recommendation system studies usually focus on one assumption in order to propose recommendations, such as the well-known assumption of the collaborative filtering recommendation family: *If two users are similar in their ratings over some items, then they are more likely to be similar in their ratings towards other items.* While this is a correct assumption about human, it is not the whole reality. One can look to other recommender families, for instance the social-based recommenders that operate on the assumption that: *a user is similar to his social milieu, so that using their information (their rated items) is a good way to recommend to the target user*.

This is another correct assumption, but again it is incapable of describing the whole reality behind recommendation.

In our work we investigate these aspects of the recommendation systems world; more precisely, our work investigates the following problems:

First problem: We investigate GPSNs to determine if one can make predictions based on spontaneous actions on social networks, especially concerning information that was not provided explicitly by users. For example, can one predict shopping information about users from their actions on Facebook and Twitter?

Second problem: Can one transform the implicit data in GPSNs into a Recommendation Engine, or a social machine of recommendation? If that is possible, then we will be able to create social machine that is fed by users' actions on a GPSN. In other words, this might permit us to create a perpetual motion machine based on social recommendation where users' actions feed the engine that provide them ever better personal recommendation in a reinforcing loop.

Third Problem: How can one evaluate such a social machine or recommendation engine aside from using traditional accuracy metrics like MAE and RMSE? Since this question carries greater interest, it will be discussed at length later.

Fourth Problem: Each family of recommendation systems rests on a certain assumption, like the well-known assumption of collaborative filtering where two users who complete similar actions in the past are more likely to carry out similar actions in the future. Our work addresses the fourth problem by taking a step backwards in the recommendation domain to investigate the different assumptions underlying different recommendation systems, allowing us to reflect on the validity of these assumptions and their effects on the community. Then we can provide a recommender that responds to different assumptions at the same time.

2. Proposed Solutions:

This section discusses solutions to previous questions that arose from our research, which are presented as follows:

First Question: -What information can we find in GPSNs?

To address this problem we explored existing solutions in the literature to see if social network users submit real information about themselves or present information that are a self-idealization. If the latter were to be the case, GPSNs would constitute a a waste of time as non-accurate information about users would result in poor recommendations.

Up until 2009general opinions suggested that people put information that is a misrepresentation of their true selves; however, after 2010 many studies have affirmed that users do information about themselves that is representative, descriptive, and truthful. More details about this issue can be found in Chapter 3.

Based on this topic we raised the question: Can we extract non-explicit information solely from users' explicit actions on social networks? To address this question we surveyed 63 Facebook users about why they joined Facebook. None joined Facebook for commercial reasons, which is why we decided to explore the subsequent question: can we deduce commercial information from users' explicit social information?

We extracted users' information from Facebook and Twitter to which we applied typical data cleaning algorithms, followed by the extraction of commercial information from the

profiles with the help of external world knowledge (Blue Kangaroo Tree). The extracted profiles were evaluated by the 63 Facebook users; 49 of them confirmed that the extracted profile described them. Afterward, they had the possibility to update their commercial profile and tended to add 30% more commercial information while deleting about 7% commercial concepts. With Twitter we were able to extract more than 3,000,000 commercial concepts from 12,000 profiles.

Our conclusion was that GPSNs contain a wealth of relevant commercial information. Based on this result, we used the information to build the iSoNTRE recommendation engine. This discussion receives further elaboration in chapter 4.

Second Question: How can implicit information from GPSNs power a the iSoNTRE recommendation engine?

We approached this problem by scanning the available recommendation system models. Among them, matrix factorization technologies have always attracted researchers' interest in the recommendation domain as they have shown a high level of accuracy. Since we did not have a complete recommendation matrix (user, item, rating) it was not possible to use a matrix factorization algorithm directly. However, we were able to use its inverse: based on the extraction of users' social network profiles we were able to create for each user a general profile and a domain-based profile. The domain-based profile is contained world knowledge such as film, shopping, or other interests.

In theory, the same projection can be done for items like films or products based on the same world knowledge. Then the two matrixes can be joined together by multiplying one with the transpose of the other. This works because the two matrixes are both characterized in the same latent factor space or the same world knowledge.

That makes iSoNTRE a hybrid system where the information extraction is done using content-based recommendation techniques, resulting in a matrix that belongs to the collaborative filtering family of recommendation. Thanks to iSoNTRE, items that live for short time, such as news or offers, can be recommended using collaborative filtering techniques.

This contribution will be discussed in detail in chapter 5.

Third Question: How should one evaluate iSoNTRE?

To evaluate iSoNTRE we had to build it first, which meant we had to use GPSNs, choose a world knowledge over a domain (in addition to the recommended items-), and adopt data extracted from Facebook, Twitter, and the Blue Kangaroo Tree as world knowledge. The recommended items are 10.000 offers that we got from Blue kangaroo also. Evaluation goal was to measure to what end the recommended offers were inserting to users?

First we built the matrix over users and items. Specifically, we built one for Facebook (2,000 users and 10,000 items) and one for Twitter (12,000 users and 10,000 items) and then we considered two cases:

Case 1: whereby the matrix is ready for recommendation. In this case once the matrix is built it immediately contains relevant and interesting offers for each user. We constructed a Facebook application with a limited time (two minutes) that presented the recommended items to users and asked them to evaluate them (the choices were "like" and "not like"). As the list was ranked from most important to the least important, we adopted MAP to measure

the results. 73 Facebook users tried our application in an initial trial, followed by 143 users in a second trial. With a MAP of 0.8, the results proved to be highly intriguing.

With Twitter the case was different; we were not able to invite people to use a specific application. Considering this limitation, we decided to choose users from our data set and send them the best offer among their top recommendations.

However, we had to find a way to target these users, as Twitter permits only a limited number of tweets mentioning users who are not already from the account's relations in a day. To overcome this problem we created additional accounts to help us complete the evaluation process: Sirens.

The Sirens acted as Twitter accounts that mimicked human tweeting behavior. To account for the diverse personalities of human users, we created a variety of Sirens with different personalities; these personalities came out as the accounts tweeted. This meant that some Sirens sounded sporty, others showed a knack for fun, while others were optimistic and so on. With the Sirens set up and imbued with personality, we had them send offers to certain users that we chose randomly from our data set.

80 Sirens targeted 2,400 users in the first trail; the same number of Sirens targeted 2,000 users in a second trial. Of note, the Sirens produced a Click Through Rate (CTR) of roughly 12.25% in the first trail and 22.5% in the second trial, compared to an average Twitter CTR of 4% . Feedback was collected over 24 hours. Since the choice of user accounts was randomly generated, we found that many accounts had minimal activity (they had not tweeted for long time or had a low number of tweets compared to the number of years they had been around). However, the Sirens also received a surprising number of tweets that showed a high degree of user interaction and emotion, including some thank yous and other courtesies. Lastly and of note, our Sirens received commercial offers that we had not sent out as part of the study, meaning other Sirens were likely already on Twitter conducting similar research!

Case 2: in this case we considered the system in work, meaning we had the matrix that contained users' extracted information as well as users' recent actions on items. We applied different recommendation algorithms like user-user or SVD over this matrix and we calculated the RMSE and MAE values.

Most of the routine work for the data extraction through Facebook and Twitter was done with the Blue Kangaroo team.

More details about the evaluation of iSoNTRE can be found in Chapter 5.

Fourth Question: How can we consider different assumptions in one system? The answer is iAmélie:

First we began by considering the assumptions of different recommendation systems, taking into particular consideration traditional collaborative filter models, domain-based models, as well as social recommenders and circle-based recommenders. Upon review we argued that while all their assumptions were correct not one system could work for the differing recommendation needs of one user. In some cases a user may prefer the experience of similar people, while in other cases they would prefer the experience of similar people in a specific domain. In another scenario they might prefer the experience of their friends, which could be further qualified through either a general or specific domain. With so many possibilities, the ideal case is to have a recommender that respects the different assumptions together, which in our scenario turned out to be the iAmélie system. iAmélie belongs to the

family of context-aware systems and works through the following context: “my friends might not be exactly like me but they know me better.”

The iAmélie model considers domain-based recommendation as a special case of collaborative filtering. In addition, it considers circle-based recommendation as a special case of social recommendation. The user is the one who decides the appropriate context that correspond to his/her needs.

To evaluate iAmélie we built it over a system that contained users, items, ratings and relation over different categories. For this we used a data set of epinions that contained 127,711 users, 331,274 products, 1,199,632 ratings, and 582,613 trusts over 27 categories.

In order to properly build iAmélie we had to change in the R recommender lab package, which already contained the collaborative filtering recommenders, to which we added the domain-based, social, and circle-based recommenders. After that we explored this data over the normal recommendation methods calculating RMSE and MAE over the new methods that we had added.

To evaluate iAmélie we did not have the possibility to refer to real epinions users as the site had been closed. Thus we referred our users (25 students) to judge the results that were produced for a group of epinions users. Furhter, we asked our users to evaluate each recommendation that resulted from iAmélie and another recommender (collaborative filtering, domain based, social recommendation, and circle based) based on a defined list of questions. The results showed a high tendency towards iAmélie system besides users motivation to the system. iAmélie is discussed in detail in Chapter 6.

iSoNTRE and iAmélie complement one another. By putting these two models together, we will be able to build a recommendation system from GPSNs that can respond to different contexts and user assumptions.

3. Dissertation Plan

Historically, researchers have reviewed recommendation domains in a general overview; however, in the dissertation we decided to introduce the recommendation systems in Chapter 2, followed by a discussion of the social recommendation and contextual recommendation systems in Chapter 3. We begin the latter by introducing each of the social networks and context awareness concepts in general then investigate how each of them has affected the accuracy and quality of recommendation.

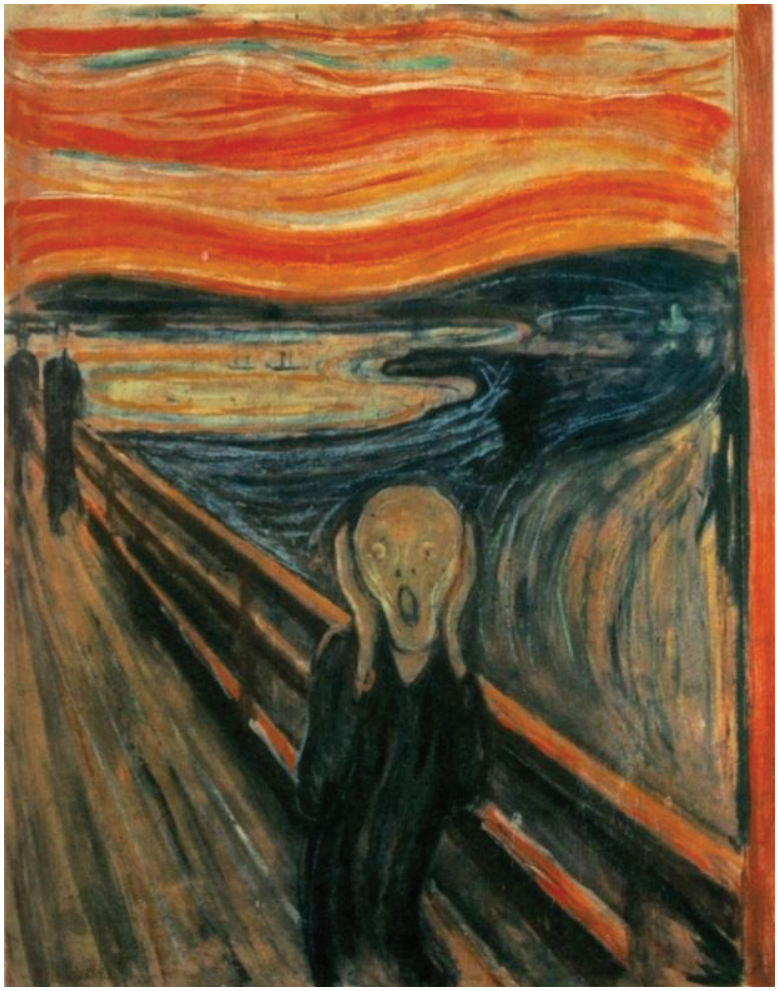
In Chapter 4 we introduce the two proposed models, iSoNTRE and iAmélie. We elaborate the argument for and motivation behind each of them, and how both can be linked to collaborate together. Then in Chapter 5 the iSoNTRE model will be introduced and fully evaluated, followed by Chapter 6 where the iAmélie model will be introduced and evaluated equally. Finally, in Chapter 7 we conclude and discuss the limitations and extensions of our work as well as discuss possible areas for future work.

CHAPTER 2: RELATED WORK

-Recommendation systems-

Technology made large populations possible; large populations now make technology indispensable.

-Joseph Wood Krutch



The Scream", by Edvard Munch (early 20th century). What to read next, what to watch next, what information to consume next? Decisions to take all around

1. Introduction

We are surrounded by decisions to make: which movie to watch? Which book to read? Which blog to follow? Or which item to buy? Finding the appropriate choice is like finding a needle in a haystack. For example, Netflix has more than 17,000 movies, while Amazon has 410,000 titles in the Kindle store alone. That is why recommendation systems are everywhere, and their use becomes essential in daily life.

Recommendation systems stem from the needs of users. They hold interesting scientific questions, as they combine aspects from human-computer interaction, information retrieval and machine learning domains. From a financial standpoint, good recommendations promise to increase income for the reason that when users find interesting items, they will consume them and then come back for more good recommendations. The movement towards recommender systems covers many domains, likes movies, music, news, books, research articles, search queries, social tags, products, restaurants, and jokes. Even experts, financial services, life insurance, persons (online dating), and twitter followers.

The main idea in recommender systems is to offer interesting resources or items to users. Recommender systems follow the assumption that recommended items might be interesting for a user, if he knows about them he will consume them, but he is not able to find them himself. That is what differentiates them from information retrieval domains where the user knows what he is looking for. Other difference between recommendation systems and information retrieval domains is personalization—recommender systems specialize in this regard. Two users can receive totally different lists of recommended items based on their past actions, which is not the case in information retrieval systems. However, we can see the recommendation process as an information retrieval problem, where the domain of items is queried by the user's preference profile.

Recommendation systems have been studied for more than two and a half decades. Within this period, a variety of algorithms has been developed; mainly, collaborative filtering techniques based on leveraging the ratings of similar users in a user population. In the last few years, social recommender systems have appeared based on the increase in online social networks use which has created a tremendous amount of information from users. In this regard, content filtering techniques have also evolved based on finding similar items to those liked by users, based on items' keywords and textual similarity, and hybrid solutions which combine both. In addition, a variety of tools for evaluating recommenders' performance have been studied, proposed and used.

Users are the core of any recommender system and recommender systems need to learn about users in order to perform better, particularly questions relating to how to collect information about the user, how to have reliable information, how to be able to respond to their needs, as well as how to consider privacy, diversity and transparency aspects in recommendation.

We agree with Martin [1] in his 2009 keynote at the recsys conference, where he argued that in the recommendation world algorithms themselves are only a small part of how to provide recommendation to users. There are already plenty of algorithms that work well. Currently, work needs to be done concerning the user experience, how to get data, and other problems related to the recommendation experience.

We cannot cover the whole recommendation domain in this chapter, as there is a lot of work in this domain, but we will present a brief overview of the fundamental works and then focus on works related to the model discussed in later chapters.

The focus of this chapter will be on collaborative and content filtering recommender, especially semantic enhanced content based recommenders, as we will propose in next chapters hybrid solutions in order to generalize the social recommenders into a generic social recommender machine. In this chapter also, we will detail the evaluation methods because in our solutions we will use live evaluation methods besides traditional ones. However social recommenders will be discussed in the next chapter. At the end of this chapter, we will cover the cold start problem of recommender systems, where new users and items in collaborative filtering recommenders suffer from not having enough ratings to be recommended. As well as we will introduce the SLiR resources which are the short life resources. Which live too shortly so they can't collect enough ratings to be recommended. The contribution introduced later will address both of these two problems.

2. Historical Overview:

Group lens [2] was one of the first user-user automated recommender systems in the movie domain, followed by Ringo [3] for music, BellCore for video recommendation [4], and Jester [5] for jokes. After the debut of these systems, collaborative filtering begun to be more common throughout the 1990s, especially with the rise of Amazon in 1994, which recommend items to users based on their purchase history, viewed and browsed items. Later, other recommendation algorithms were developed and used, like the case-based recommenders as well as content-based ones based on similarity between items' content and the items the user has rated [6], [7].

Consequently, as recommender systems refined their algorithms for specific operations, hybrid recommenders [8] emerged in order to merge the benefits and advantages of different recommendation algorithms into one recommender. A high variety of hybrid recommenders exists in the domain. Below, we discuss a number of recent recommender systems and the topics they cover.

In 2006, Netflix offered a prize of 1 million USD to the group that would enhance the state of the art in movie recommendation, with the goal to beat the CineMatch algorithm in offline tests by 10%. In 2009, the prize was given to BellKor's Pragmatic Chaos team in 2009, which enhanced Netflix's algorithm in the predicting ratings by 10.06%, with RMSE metric to evaluate of 0.8567[1].

Pandora[2] and last.fm[3] [9] are examples of content-based recommenders; in this case through song recommendation based on song attributes and using the feedback of users to provide new recommendations that are similar to songs they have already liked. In Pandora's case, the purpose is to seed a "station" that plays a selection of music with similar properties. In order to accomplish this, song attributes were studied by a team of musician-analysts who listened to numerous songs and determined nearly 400 possible attributes. Nonetheless, as the music annotation process is manual, the system suffers from scalability problem [10]. Last.fm is similar in that it also offer songs recommendation for users, but it is based on the actions of similar users in the system. This is an example of song recommendation by collaborative filtering techniques.

[1] http://www.netflixprize.com//community/viewtopic.php?id=1537
[2] http://www.pandora.com
[3] http://www.last.fm

Another example of content based recommenders is the Internet Movie Database which contains a wide range of attributes to describe movies including genre, director, writer, cast and storyline [4].

Epinions is an example of a recommendation social network for consumer reviews. Established in 1999, Epinions allows users to provide ratings and reviews, as well as the ability to "trust" a user; in turn, that user may be trusted by other users. As well, Epinions permits users to construct a social network of consummation and ratings. Unfortunately all community features on epinions have been disabled since 25 March 2014 according to the site's FAQs[5].

Similar to epinions, Flickr is another recommendation social network for videos and images, where users can have an online community among themselves. As both Epinions and Flickr contain recommendation systems combined with social networks, they have been widely used in most of the social recommenders (discussed in the next chapter).

Many other common internet sites use recommendation systems in order to work, such as YouTube, Yahoo, Trip Advisor, and IMDb.

So the recommendation domain is well established, but there are still plenty of disciplines to investigate.

3. Recommender Systems Classifications

Recommender systems are classified into different categories based on the general way they work or whatever technique they use [10]. The following is a list of classifications:

1. Demographic recommender system: In these recommender systems the demographic aspects of users play a role in recommendation, like the age, the language, or country. The assumption is that recommendation should vary following the change in demographic. These solutions are common in the marketing domain. However, there is not a lot of work about them in recommender systems [11].

2. Knowledge-based recommenders: In these systems recommended items are based on the domain knowledge, answering questions, how some of the item features respond to user needs and preferences, as well as how useful the item is to the user. These knowledge recommenders are case based [12], [13]. A similarity function finds how much the user needs match the recommendations.

3. Constraint-based systems: this is another kind of knowledge-based recommendation system; the main difference between the two is how the solution is calculated. In the case-based recommendation, the recommended items are based on the similarity metrics, while constraint-based recommendations predominantly exploit predefined knowledge bases that contain explicit rules about how to relate customer requirements with item features. Knowledge-based systems usually give better results in the beginning of their work. In order to maintain good results they need to have learning algorithms, otherwise they can be surpassed by algorithms like CF ones.

4. Community-based recommenders: The main idea in the community-based recommendation is to recommend to the user based on the preference of his friends, following the epigram "Tell me who your friends are, and I will tell you who you are" [14], [15]. Works

[4] http://www.imdb.com/
[5] http://www.epinions.com/help/faq/?show=faq_earnings&sb=1

in this domain were made possible thanks to the widespread use of online social networks. This kind of recommender system will be detailed in the following chapter.

5. Contextual recommendation system: In this kind of recommender system the results of recommendation vary according to the context of the user. For example, in the temporal context, the clothes recommended in summer vary totally than those recommended in winter. For a social context, an example is that a film recommended to the user alone can vary from a film recommended to be seen with his girlfriend or with the family, recommending a restaurant for a Saturday night with friends, which would vary from a restaurant for a lunch during the week with co-workers [16].

The two models that we propose in the following chapters are related to the Community- and contextual based recommenders

4. Recommendation system's architecture overview:

Any recommendation system contains a recommendation algorithm; this algorithm is a collaborative filtering, a content filtering, a case based, any other type of methods, or an ensemble of them in a hybrid model. Collaborative and content based solutions use a similarity method in order to find similar items or users to recommend based on their information. A recommender needs also a source of users' information, usually in collaborative filtering recommenders; this source is the users' ratings. While in content based recommenders, this source can be any user action like the browsing or the user log file. Social networks also rose to be a source of information about user leading to social recommenders (next chapter). To evaluate a recommender systems, accuracy has a main role, usually performed on available data sets (offline evaluation). User satisfaction although highly important is less investigated in the literature. The recommendation systems components are presented in Figure 1.

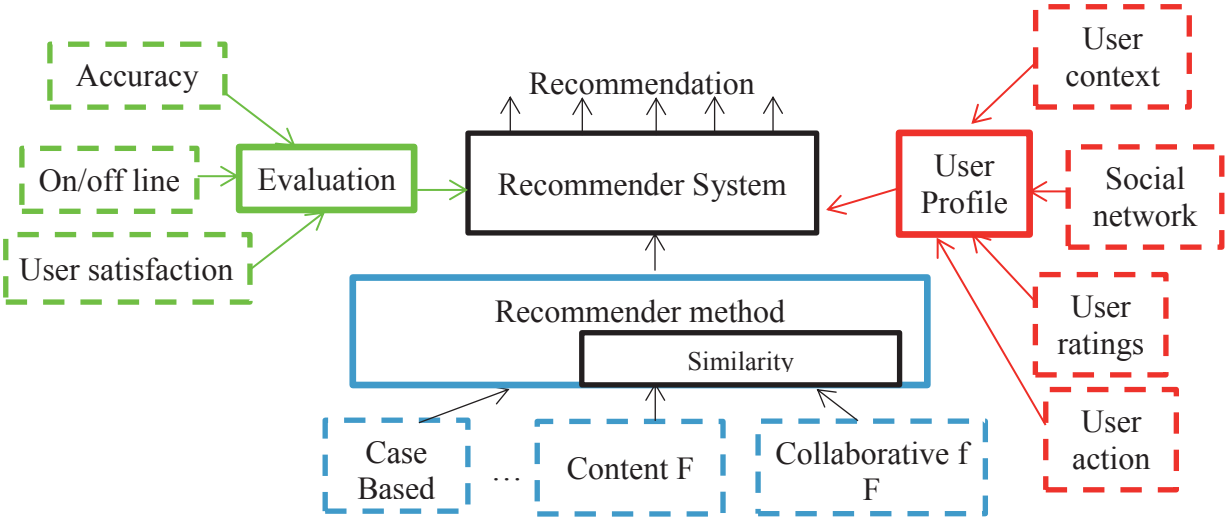


Figure 1 A recommendation system overview, mainly based on a recommendation method, a source to user information and evaluation strategy.

In the rest of this chapter we will cover the main components Figure 1, however we will not cover the case based recommenders.

4.1. Recommendation Systems algorithms

We will begin be introducing collaborative filtering, content filtering and hybrid solutions:

4.1.1. Collaborative Filtering Techniques

Collaborative filtering (CF) is based on the assumption that if two users behave towards some items in the same way they will likely behave similarly towards other items.

Most collaborative filtering techniques are based on having the user preferences, and then finding predictions of new items based on these preferences; finally, they produce their recommendations based on ranking the candidate items by the users' preferences.

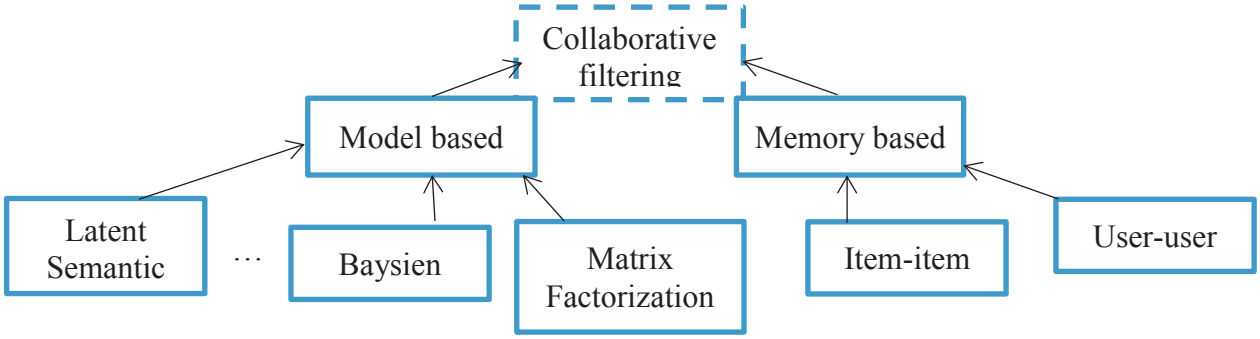


Figure 2 Collaborative filtering, a very common way for recommendation has a variety of implementation

In general, collaborative filtering techniques are divided into memory-based and model-based ones [17] Figure 2. Memory-based leverages the ratings of similar users (user-user) or similar items (item-item) in a user-item population. These methods are easy to implement and give high prediction value, which is why they are preferable by commercial enterprises like Google, Amazon, and Netflix [18].

Another family is the model-based solutions, such as using the clustering, matrix factorization [19], Bayesian algorithms [20], Latent semantic [21], and so on. Model-based algorithms are usually hard to implement and take more offline time to build the recommender. They give relatively better predictions and recommendations but at the risk of losing important information, especially in case of non-active users. Most of them do some sort of information summarizing or compacting.

Some definitions are common in the recommendation domain, the *User* who has done some actions and is to be recommended to, then the *Item* that needs to be recommended, and finally the *Rating*, which represents how much a user is interested in an item. So the tuple *(User, Item, Rating)* is the core of recommender systems.

Ratings can have many forms. It can be a value from one to five (1-5 score) or it can be any other form that indicates an interest level (like, dislike). Unary data are one kind of rating that is not direct. For example, when a user buys an item he might not rate it but buying it is usually an indication that the user is interested in the item [10] (Amazon assumes that the user rated 5 if he bought the item). Unary can be extracted from different sources (users' logs, browsing history, or even information in social networks). Dealing with this data needs special treatment because it is not as direct as a personal rating. If a bought item is a gift, for example, it doesn't describe the users' interests; rather, it describes the friend's (according to the buyer) interest. Usually these problems are ignored when dealing with unary data.

The set of all values *(User, Item, Rating)* result in a sparse matrix called rating matrix. The pairs of *(User, Item)* are used where a user did not provide a rating.

Any collaborative filtering algorithm needs this matrix as a starting point, and then comes two main recommendation tasks:

- The prediction task: for these unknown values of *(User, Item)* how much would the user rate this item if has been asked to do so?
- The recommend task: for a user, find the best ranked list of *n* items for the user needs.

Many elements are taken in consideration in the recommend task, that is why it is not necessarily the items with high prediction values that are recommended.

All in this chapter, as well as in the next ones, we will refer to items by *M* and to users by *U*. The items rated or bought by user *u* are M_u. While the users who rated or bought an item *m* are referred to by U_m. The rating matrix is $\hat{R}_{UM}$ and the rating of user u to item m is r_{um} . An example of the rating matrix is in table 1.

	Product 1	Product 2	Product 3
User 1	5	4	5
User 2	5	4	?
User 3	?	4	5
User 4	3	1	?

Table 1 An example of a rating matrix on a 1-5 scale.

r_u is a vector of all the rating provided by user u , as well r_m is the vector of all ratings provided to item m. The matrix $\hat{R}_{UM}$ is usually sparse.

$\bar{r}_u$ is the average of user u ratings, and $\bar{r}_m$ is the average of the item m ratings.

1.1.1.a. User-User Collaborative Filtering:

User-user collaborative filtering, known also as *k-NN collaborative filtering,* is the first automated CF method [2]. In the direct interpretation of the concept of collaborative filtering the past ratings of the user are used to find similar users to him, then the items that these similar users rated and the user himself did not might be interesting to him so that these items are weighted by how much each of these users is similar to the main one.

This is made clear in the previous table, where user 1 and user 2 are too similar in product 1 and product 2, so product 3 might be interesting to user 2 as it was for user 1.

In order to work, user-user collaborative filtering needs a similarity function between users as well as a way to use the information and similarity to find prediction. The prediction is performed using the neighborhood predictor, based on finding the N neighbor of the user u, then the system combines the ratings of these N users in order to provide a prediction for the item i, using the similarity in order to weight:

$$\mathrm{P}_{\mathrm{u,i}} = \bar{r}_u + \frac{\sum_{\dot{u} \in N} s(u,\dot{u})(r_{\dot{u},i} - \bar{r}_{\dot{u}})}{\sum_{\dot{u} \in N} |s(u,\dot{u})|} \qquad (1)$$

Some solutions do not provide any weighting to calculate prediction like in [4] but using weight so far is the most common. It is simple and also works well in practice.

Another important factor is the choice of the *N* or the neighbors. Some systems may choose to have the users as neighbors; some might choose to consider only the most N_i users

who have rated the item i. Limiting the size of neighbor enhances the results because it reduces the noise resulting from having all users in order to find the prediction.

The choice of i in the N_i is domain dependent; it needs some investigation in order to find the neighbor size in each domain. [22] show that in the film domain, a size of 20 is a good value in the offline analysis of available film ratings. In many domains a starting value between 20 and 50 is a good starting point.

However, user-user collaborative filtering cannot provide recommendation for users with strange tests, because it is not easy to find for them similar users.

1.1.1.b. Item-item Collaborative Filtering

Although effective, user-user CF suffers from a scalability problem when the number of users grow. Finding neighbors has $O(|U|)$ complexity, or even might be worse, based on how many similarities are calculated. That is why the item-item collaborative filtering algorithm was proposed. Item-item collaborative filtering is one of the most widely used recommendation techniques today.

Item-item techniques were introduced in [23], [24], at the same time it has been used by Amazon [25]. User-user CF calculates the similarity between users' vectors in order to find prediction; item-item finds the similarity in the item's vector. If two items gathers similar values of likes/dislikes then they are similar, and users will consume items similar to their previous items. The concept is somewhat similar to the concept of content based recommendation, but this time, the item similarity is calculated from the users' actions over items rather than the item's data.

That is why an item-item algorithm can provide a similarity between, for example, a summer clothing item and an ice maker, as users might have similar behavior over both—this is something that content-based methods cannot find.

Item-item still needs to find the most similar items in order to work; this similarity is to be calculated over smaller dimensions, because usually the items' number is less than the users' number. However, the gain in calculating the similarity is not that big. The main gain in item-item collaborative filtering is that it allows one to calculate the similarity matrix offline.

When users complete more actions in the system, like the rate or the purchase, his profile vector (ratings) will change. This means his similar users will also change. As well, the ratings of other users can change, also resulting in changing the similarity. This means that the similarity in the system can change by the actions of any user in the system. That is the reason why similarity in user-user collaborative filtering is computed in the time of prediction or recommendation; it cannot be calculated offline as any action of any user can change the similarity.

When the number of users is greater than the number of items, the similarity between two items will not change when an action is done in the system, like if a rating is added or updated, especially if the two items have already enough previous rating. This permits us to pre-calculate the similarity between items in an item similarity matrix to be ready to find prediction and recommendation. The matrix can contain only the most similar items. The matrix can be updated from time to time in order to include the new ratings and actions, but in the meantime the recommendations that are produced will be good. So the update can be done in the low-load time of the system—this is the main benefit of item-item collaborative filtering.

Item-item collaborative filtering uses the users' own ratings combined with the items' vectors of received ratings to find recommended items; not like user-user, which depends on the similarity between two users in order to find new recommendation.

However, item-based recommender systems fail to find surprising items for users, as their recommendations are based on similar items to the ones the user has rated. Additionally, they are not able to find best-seller or hot items.

Item-item collaborative filtering needs a prediction function as well as a similarity function to work, like the user-user collaborative filtering. The prediction is similar to the user-user algorithms where the similarity score is used to generate a prediction using a weighted average.

The principle is to find the N similar items to i, then the prediction $P_{u,i}$ can be calculated using the following expression:

$$P_{u,i} = \frac{\sum_{j \in S} s(i,j) r_{u,j}}{\sum_{j \in S} |s(i,j)|} \qquad (2)$$

Where S is selected to be the N elements the most similar to the item j while the user has rated N items of the neighbors. In the movies domain, this N has been found to be = 30 according to [24]

This equation's problem is that some weights might be negative; this is solved by thresholding similarities so only non-negative values are considered or by averaging the distance from baseline prediction.

1.1.1.c. User Similarity

Different similarity algorithms exist that can be used to find similar users/items. The choice of a similarity method affects the recommender performance. The most common ones are the Pearson correlation and the cosine similarity.

a) **Pearson similarity:** It calculates the statistical correlation between two users' common ratings to determine their similarity. It is used in [62, 119]

$$S(u,v) = \frac{\sum_{i \in I_u \cap I_v} (r_{u,i} - r_z)(r_{v,i} - r_z)}{\sqrt{\sum_{i \in I_u \cap I_v} (r_{u,i} - r_z)^2} \sqrt{\sum_{i \in I_u \cap I_v} (r_{v,i} - r_z)^2}} \qquad (3)$$

Its problem is that it can give high value of similarity between two users who have few ratings in common. Therefore, adding a limit to the number of co-rated items between the two users, as well as scaling the similarity of the number of co-rated items is less than the limit [26]. A threshold of 50 showed to be a good value in practice.

b) **Cosine similarity:** Cosine similarity calculates the differences between the two vectors of users. It is an algebra model different from the statistical Pearson similarity. Similarity is measured by the cosine distance between two ratings vectors.

$$w(x,y) = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} (x_i)^2 \sum_{i=1}^{n} (y_i)^2}} \qquad (4)$$

The value provided to an unknown rating is 0 so that these values don't harm the similarity calculation.

[27], [26] showed that the Pearson correlation has been found to give the best results for user user collaborative filtering methods. However, cosine similarity [24] is the most popular metric in finding item-item similarity.

4.1.2. ***Content-Based Recommenders***

Content-based recommendation systems are another family of recommendation algorithms. The main concept is to recommend items to a user that are similar to the items that he has already consumed. This is done first by analyzing the descriptions of items the user has rated then building a user profile [28]. The profile should represent the user's interests of items. Then the recommendation is performed by matching the user's attributes with items' attributes, resulting in a score of how interesting this particular item to a user. An accurate representation of the user can result in a tremendous gain.

Recommendation involves three steps [10]:

1. *The content analyzer:* sometimes information is not well structured to build the user profile directly; sometimes it is in a text format like web pages, description of an item and so on. In this case, a feature extraction technique is used in order to transform the information into the target representation, like transforming web pages into keyword vectors. The techniques used are borrowed from the information retrieval domain [29][30]. In these techniques the item description coming from the information source is processed in order to extract features—keywords, n-grams, concepts, etc—so that the item moves from unstructured text to a structured representation useful for recommendation.
2. *The profile learner:* based on the gathered data about the user, this step tries to generalize the data into a user profile. Usually a machine learning technology is used; for example, it is in the web pages using a relevance feedback method [31]. In this method the user's negative and positive feedback on web pages (likes, dislikes) is collected in order to find a prototype vector representing the user profile.
3. *The filtering component:* this step uses the user profile in order to find relevant items to the user that match his profile. The result is either a binary value or a continuous judgment that is computed using similarity metrics [32]. Then a ranking is used to choose the best items to recommend for the user. The similarity can be found using any similarity metric like the cosine similarity.

Content based recommenders do not suffer from a cold start with a new item in case that we have a good description of the item, as the recommender is based on finding the similarity between items. However, they suffer at the user level, as a new user has no actions yet and has no profile, so he cannot be recommended to; in the time that he needs in order to create a richer profile recommendations will be poor. Yet content-based recommenders have the advantage of user independence, which means the user does not need to have active users that are similar to him in order to receive recommendations. In addition, the recommendation obtained by this family of recommenders has the advantage that it is easy to be explained to users, not like the collaborative filtering ones, which are based on similar but unknown users and are not easily explained to the user.

Content-based recommenders suffer however from over personalization, which means that they cannot find unexpected items. That is why some ways to add serendipity to the resulting recommendation have been explored. One way to do that is by adding

randomness, like in [13] that was achieved by using a genetic algorithm. Another aspect in those recommenders is not to recommend an item that is too similar to the item already consumed by the user. In Daily-Learner [33] the very similar news were excluded. Zhang in [34] has proposed redundancy measures in order to ensure that the recommended items contain new information.

In addition, as the features are usually limited, sometimes this recommender type cannot correctly represent the user's profile. One solution is to integrate "semantic analysis" through ontology or world knowledge so that they can go beyond traditional keyword matching. However, the domain might play a major role in such recommenders, as in some domains like jokes or poems it is immediately evident as to what the user interests are.

Unfortunately, the shortcomings of over personalization make content-based systems not highly effective for the real-world needs.

1.1.1.d. Content-based Techniques

At the level of item representation most content-based recommender systems use relatively simple retrieval models, such as keyword matching or the Vector Space Model (VSM) with basic TF-IDF weighting.

In the VSM, each document is modeled as a vector of term weights; the weight represents the association between the document and the term. If $D = \{d_1, d_2, ..., d_N\}$ is the set of documents to be recommended, and $T= \{t_1, t_2, ..., t_N\}$ is the dictionary or the words of the domain, then each document d_i can be represented as an n-dimensional vector $d_i= \{w_{1j}, w_{2j}, ..., w_{nj}\}$ where w_{kj} is the weight of the term t_k in the document d_j. T is obtained by using some standard natural language processing steps like the stopwords removal and stemming [35]. There are many ways to calculate the weights, the most common is the TF-IDF (Term Frequency-Inverse Document Frequency) weighting method, equation 5:

$$w(t,d) = \frac{tf_{t,d}\ log(\frac{N}{df_t})}{\sqrt{\sum(tf_{t_i,d})^2\ log(\frac{N}{df_t})^2}} \qquad (5)$$

The similarity between two documents is calculated using any similarity method. Cosine similarity, however, is the most widely used.

In the VSM model the user profile also is modeled as weighted term vectors. Prediction is done using the similarity as in the cosine one.

An example of a system that uses a keyword model can be found in the area of web recommenders, such as famous systems like Letizia [36], Personal WebWatcher [28], [37] Syskill & Webert [38], [39], ifWeb [40], Amalthea [41], and WebMate [42]. They are surveyed in detail in [10]. This class of systems can also be classified further. For example, Letizia relies on the implicit user feedback in order to recommend web pages [40], whereas ifWeb, which represents profiles in the form of a weighted semantic network, uses explicit user feedback.

A new field of filtering has also come into use, some examples of it are NewT [13], INFOrmer [43], NewsDude [44], Daily Learner [33], and YourNews [45]. In NewT many filtering agents are trained to use the user explicit feedback (positive and negative feedback) for articles, authors or sources in different types of information, like sport or political domains. YourNews creates for a user a profile of 8 different topics represented as a weighted prototype term vector extracted from the user's news view history. Then, a short-life profile is

produced by considering only the 20 most recently viewed news item, whereas long-term profiles consider all past views.

Different systems of the new filtering domain compute a short-life profile and a long-life one, such as NewsDude, which finds the short-term one by the cosine similarity and long-term one by a naive Bayesian classifier.

Other systems include more complex representations of profiles, like in PSUN and INFOrmer. In the former, profiles are initially provided by presenting the system with some articles the user finds interesting. Recurring words are recorded by means of n-grams stored in a network of attracting words, and then a genetic algorithm method is used. Importantly, the system needs an explicit feedback of the user.

To conclude, this family of content recommenders gives accurate recommendations when training sets with big numbers of examples can be used, leading to a meaningful profile built for the user. However, such models cannot find relevant semantic items. If a user likes "French impressionism" the system cannot recommend to him documents related to "Claude Monet" or "Frédéric Bazille". This is why integrating semantic analysis with content-based recommenders has been studied [46].

1.1.1.e. Integrating Semantic Analysis into Content Recommenders

Integrating semantic information in content-based recommenders can result in richer and more accurate users' profiles, which refer to concepts that are used from an external knowledge base. The challenge is to find a knowledge base that contains the information needed in the domain, which can help for both building the users' profiles and filtering the recommended resources.

Solutions in this domain should answer the following questions [10]:

- What type of knowledge base is used? (e.g. lexicon, ontology, etc.);
- How is the item represented?
- What content is included in the user profile?
- What is the item-profile matching strategy?

Some main works that include the semantic analysis in content recommenders are listed in Table 2. We notice that different domains are covered by SiteIF [47], ITR (ITem Recommender) [48][49], SEWeP (Semantic Enhancement for Web Personalization)[50], Quickstep [51], News@hand [52], and Interactive Digital Television is proposed in [14]. In addition, there is the JUMP System [53], which is capable of intelligent delivery of contextualized and personalized information to knowledge workers acting in their day-to-day working environment.

We conclude that WordNet plays a role in the disambiguation in many of the previous works; this highlights the importance of linguistics knowledge in such recommenders. However, most of the solutions also incorporate a domain knowledge, which was ontology in most of the cases.

To summarize, all the studies used either linguistic or domain-specific knowledge or both in content-based filtering methods, which performed better than traditional methods of content recommendation. This encourages researchers to design novel filtering methods that formalize and contextualize user interests by exploiting external knowledge sources such as thesauri or ontologies.

- **Encyclopedic Knowledge Sources for Semantic Analysis**

Many sources of world knowledge have become available in recent years. Users of the web have played a role in building these general purpose knowledge bases, including the Open Directory Project (ODP), Yahoo! Web Directory, and Wikipedia.

Now there is an effort in the recommendation domain to try to link this general purpose knowledge with recommenders like Explicit Semantic Analysis (ESA) [54][55], and the Wikify! system [11], which has the ability to identify important concepts in a text, a process known as keyword extraction, and then link these concepts to the corresponding Wikipedia pages.

In [56], the Wikipedia pages were exploited about films in order to enhance the predictions for the Netflix Prize competition in the film recommender. Other different solutions exist using Wikipedia [57][58].

We highlight once again that the main contribution of building semantical recommenders is finding users' profiles of concepts, then trying to recommend items or links that are linked to these concepts to users. The advantage of these solutions is that it permits the finding of likeminded users, even if they do not consume or click on the same items. For example, two users can both listen to, like and buy articles related to jazz music and still not be different according to traditionally collaborative filtering similarity calculation if they do not consume exactly the same items.

	Goal of the System	Knowledge Source	Item Representation & Disambiguation	User Profile	Matching Strategy
SiteIF	Personal agent for a multilingual news web site	MultiWordNet: multilingual lexical database where English and Italian senses are aligned	List of MultiWordNet synsets using Word Domain Disambiguation	Semantic network nodes represent synsets found in the documents read by the user	Semantic Network Value Technique [92].
ITR	recommending items (has texts) in different domain	WordNet lexical ontology	bag-of-synsets BOS (synset vector), Word Sense Disambiguation	Na¨ıve Bayes binary text classifier	Probability for the item of being in the class "interesting"
SEWeP	Web personalization system	Manual domain-specific taxonomy of categories: WordNet	Keywords extracted from their content then keywords are mapped to the concepts WordNet-based word similarity measure	No profile but navigational patterns	categories which are "semantically associated" to a pattern to expand the recommendation

Quickstep	On-line academic research papers	Research paper topic ontology by the DMOZ open directory project6 (27 classes used)	Semantic annotation of papers consists in associating them with class names within the research paper topic ontology, By k-Nearest Neighbor classifier	Set of topics and interest values in these topics	Correlation between the top three interesting topics in the user profile and papers classified as belonging to those topics
Informed Recommender	Consumer product reviews to make recommendations	Ontology based on knowledge representation and sharing (opinion quality, product quality,)	No profile	No profile: computes a set of recommendations on the basis of a user's request	Text-mining process automatically maps sentences in the reviews into the ontology information structure
News@Hand	News recommenders	Ontology based on user and items (IPTC ontology7)	Item descriptions are vectors of TF-IDF scores in the space of ontology concepts	User profiles are represented in the same space, except that a score measures the intensity of the user interest for a specific concept	Cosine similarity
Interactive Digital Television	TV recommender	OWL ontology	Item ontology	Profile ontology, a formal representation of the users' preferences	Exploits the knowledge stored in the user profile to discover hidden semantic associations between the user's preferences and the available products

Table 2 Works that include semantic analysis in the content based recommenders.

4.1.3. ***Hybrid Recommenders***

Hybrid recommender systems are based on combining different recommendation algorithms into one recommender in order to enhance recommendation. In [59], hybrid recommender systems are classed into seven groups:

1. *Weighted recommenders:* they combine the score generated by different recommender systems (or predictors) into one recommendation list.
2. *Switching recommenders:* depending on the context, the recommender can switch from one recommendation algorithm to another in order to have the best recommendation.

3. *Mixed recommenders*: somewhat like the weighting recommenders, where many recommender systems are used, but the results can be presented to users in different lists instead on only one list collecting the results.
4. *Feature-combining recommenders:* these use information gathered from many recommendation sources as input to a meta-recommender algorithm.
5. *Cascading recommenders:* the output of one recommendation algorithm is used by another one.
6. *Feature-augmenting recommenders:* the result of one recommendation algorithm is used as a feature input to another recommender.
7. *Meta-level recommenders*: it trains a model using one algorithm, then uses the whole model as an input for another recommendation algorithm.

The use of hybrid solutions is very common in order to overcome the disadvantages of some recommendation algorithms by utilizing the advantages of other ones. A good example is overcoming the cold start problem in collaborative filtering by combining it with a content-based recommender, where content based recommenders are based on items features which are usually available.

Linking content-based recommenders with collaborative-filtering recommenders had always been an attractive idea. Some of the works that include both are Fab [60], which do it to recommend web pages using agents that learn users' profiles from the pages they visited, and WebWatcher [61], which offers an agent that provides users with relevant pages when they are browsing the web based on their log files and past behaviors as well as the other users. users are modeled as vectors and cosine similarity was used in the model, P-Tango [62] is a news recommender that keeps the content-based recommender separate from the collaborative filtering one, but at the end it uses the two in order to offer recommendation. Learning about the user is performed directly and indirectly. There is also ProfBuilder [63], which is intended to enhance recommendation on the web for users, whereby the basic idea is to use the entropy of the page combined with the sequence of pages viewed by the user in order to find out to what extent the web page is interesting to this user. It combines a probabilistic method with an agent to work. PTV [64] Personalised Television Listings[6] compiles automatically personalized guides to match the likes and dislikes of individual users. In the Content-boosted Collaborative Filtering [65] a content-based predictor was combined with collaborative filtering in order to enhance existing user data and then provide personalized suggestions. Cinema Screen [66], a film recommender agent, expands and fine-tunes collaborative-filtering results according to filtered content elements, namely actors, directors, and genres. This approach supports recommendations for newly released, previously unrated titles. And lastly, in [67] a feature profile of a user is used to reveal the duality between users and features. Then they apply Latent Semantic Indexing Model (LSI) to reveal the dominant features of a user, which then provides recommendations according to this dimensionally-reduced feature profile.

4.2. Recommendation systems Cold Start Problem

A recommendation system's main drawback is the cold start problem, which means the difficulty of recommending new items when there are not enough completed ratings, and the difficulty in offering recommendations to new users who have not yet rated items. The worst case is the need to recommend new items to new users.

[6] http://www.ptv.ie

Collaborative filtering methods suffer heavily from both issues, while content-based recommenders suffer from the user cold start problem; as for new users, on whom we know nothing, we cannot recommend items. However, content-based recommenders do not suffer from the problem at the item level.

The major factor that makes the cold start problem a real challenge is that it is a continuous problem—new items appear each day, and new users also join systems daily.

To say that an item is part of the cold start problem, we need a threshold; say, an item that has been in the system for less than a certain amount of days (1 day), or an item that has less than 10 ratings [68]. We prefer to consider the amount of ratings and actions the item has gathered as an indication of a cold item, as some new items can gather a great amount of ratings in only a short time (like a long awaited film).

Many solutions have been proposed for the new item problem, but the field has found the research lacking, [69] tried to overcome the cold start problem by using the users, items, and item description. Their model combines the collaborative data with content in a probabilistic model. In [70] Six methods to learn about the user were proposed to be used in collaborative filtering recommenders. The idea was to choose a sequence of items to present to every user. [71] propose the trust-aware system in order to solve the cold problem. This model uses each user's "web of trust". Then the trust propagation between users and their cold users is used to infer the weights of unknown users. Another group of authors propose a hybrid approach in [72]. Their model utilizes a combination of the CF approach with the CB combined with probabilistic aspect models. Following this work comes [73] who used a predictive feature-based regression model that combines all available information about users and items to overcome the problem. In [77], an algorithm to provide personalized recommendation was proposed on social tags, especially when tags are assigned to diverse topics. In [74], authors adopt a solution-based system on association rules. They use these rules to expand the user profile, which overcomes the cold start problem. In order to give good results there should not be redundant rule sets. This condition is not possible with large data sets. In [75] authors use functional matrix factorization (fMF). This fMF builds a decision tree for each interview, where each node is considered as an interview question, so that the recommender system can query the user adaptively. Lastly, in [76], authors propose a new user profiling model. It is a kind of interview that tries to elicit the opinion of users about items, then an adaptation schema on the users' answers is proposed in order to have better results. In [77] authors use social tags to solve the cold start problem.

Most recently in [78], [79] authors propose models to overcome the user cold start problem by using the users' demographic characteristics, following the assumption that people with a common background and similar characteristics are likely to have similar preferences. Their model involves three phases: first, provide means for the classification of new user in a specific domain; second, they propose an intelligent algorithm to find "neighbors" of a new user in the most corresponding group; the third step is to use the prediction techniques to find ratings for a new user.

When evaluating a recommender (discussed later) one important factor is to know how many cold start items/users the system contains, knowing that recommending from these cold items affects the accuracy of the system, which is another aspect to consider when evaluating. However, it might be better for the whole system to recommend from these cold start items in order to achieve the two goals of novelty and serendipity, which perhaps results in lower accuracy, but more user satisfaction [80].

In the recommender industry the cold start problem can be solved easily by beginning with the popular items, as they are popular they are most likely to be liked by these users. The next step is based on the users' actions towards these items so that the personalization begins.

4.3. Short-Life Resources Recommendation (SLiR)

Short-life resources are the resources, items, or articles that do not live for a long time, like news on a news site, or offers on a commercial site. The news article is highly important directly within the minute in which it is published, and after a while, they are old news and not interesting anymore. In the commercial example, the offer is available only for short time, an example: "A discount of 50% on a camera within seven days". The product itself (a camera, for example) will stay for long time, while the offer (discount of 50%) will disappear within a short period (7 days). These types of resources appear and disappear before collecting enough ratings to be recommended correctly, which makes impossible to use collaborative filtering techniques. Instead, popularity recommenders, as well as content-based recommenders, are used in order to recommend them.

4.4. Data Sources for Recommender Systems

Recommender systems are based completely on users' feedback. User feedback towards items can be obtained two different ways. First there is explicit feedback, where the recommender system asks users explicitly to provide ratings. This feedback usually is on a scale of 5 points, 7 points, or 100 points. Most recommender systems ask the user to provide a unique rating for an item; however, some systems ask for more than one rating, such as Zagat's restaurant guides[7], which require 3 ratings (food, service, and décor) for each restaurant. In addition there is implicit feedback which is the data extracted from the user's actions (browsing, purchased items, etc.). Implicit feedback is very common in the commercial sites and is referred to as unary data, where unary liked items are marked as 1, while the others as 0. We cannot call this data binary data, as binary indicates liked or disliked, and in the case of implicit data a user who has not clicked on or purchased an item does not show an indication that the he did not like the item; perhaps he did not know about the item, and if it became known then he would consume it. The time spent on an item is also an indication of how interesting the item is to the user.

Most of the work in the literature deals with explicit feedback, while some works deal with implicit data, such as [81][23].

The time stamp is important information that can also be used for both kinds of feedback. Usually explicit data is hard to obtain; especially in cases where the user did not rate an item (a film or a song), where his behavior is to leave the rating for something interesting, and not necessarily that he will provide a low rating. Another problem with explicit rating is that it may change with time. In [4] they have shown that users provide inconsistent ratings when asked to rate the same movie at different times. They suggest that an algorithm cannot be more accurate than the variance in a user's ratings for the same item. On the contrary, implicit feedback is widely available and it reflects the user's real interests.

5. Evaluation Metrics

How to evaluate a recommender system is a difficult task. Many elements play a role in how to evaluate a proposed approach, like the chosen data set size, its ratings density, ratings scale, and other properties of the data set [80]. Many aspects play a major role in

[7] http://www.zagat.com/

evaluating recommender systems, like accuracy, coverage of recommended items, confidence, trust, novelty, serendipity, diversity, utility, robustness and privacy [82][32][80].

5.1. Accuracy

Accuracy is one of the main aspects of evaluating recommender systems that has been well studied so far. In this case, accuracy is defined by: to what end the prediction about how a matched item is interesting to a user is correct. In other words, can the system predict for a user, and an item, the accurate rating value that the user might give (or has given) to this item?

The most two common metrics to evaluate accuracy are the Mean Absolute Error (MAE) and the Root Mean Squared Error (RMSE). A lot of other metrics have been proposed, like Normalized RMSE (NMRSE), Normalized MAE (NMAE), Average RMSE and Average MAE. MAE measures the average absolute deviation between a predicted rating and the user's true rating. If we consider as the data set where the evaluation is done, the real rating value of a user towards an item, and the predicted value using the model to be evaluated. Usually the test is done by cross validation runs where a part of the data is hidden so that they are not known in the evaluation process.

$$MAE = \frac{1}{N} \cdot \sum_{u,i \in N} \left| \hat{r}_{u,i} - r_{u,i} \right| \qquad (6)$$

RMSE calculates the square root of the mean/average of the square of all of the errors:

$$RMSE = \sqrt{\frac{1}{N} \cdot \sum_{u,i \in N} \left(\hat{r}_{u,i} - r_{u,i} \right)^2} \qquad (7)$$

From RMSE definition, it is clear that it is always greater than the MAE, and gives more weights to errors with larger absolute values, while MAE gives the same weight to all the errors.

Thus RMSE is suitable for situations where small prediction errors are acceptable, predicting 3 as rating instead of 4 is acceptable, but 4 instead of 0 is not.

As shown in the table 3, as few individual error values increase, RMSE increases. One large error increases dramatically the RMSE.

Error on Predicted Rating->	Error 1	Error 2	Error 3	Error 4	RMSE	MAE
Scenario 1	1	1	1	1	1	1
Scenario 2	2	2	0	0	$\sqrt{2}$	1
Scenario 3	4	0	0	0	2	1

Table 3 RMSE and MAE comparison on various scenarios

In order to evaluate the whole recommended list and how it is useful for the user, precision (measure of exactness or *quality)* and recall (measure of completeness or *quantity*) are borrowed from the information retrieval domain. In simple terms, high precision means that an algorithm returned substantially more relevant results than irrelevant, while high recall means that an algorithm returned most of the relevant results.

When performing offline evaluation a part of the user-item ratings is hidden, then we ask the recommendation system to predict the ratings that the user might give to the item. Four cases might result from this assumption and are shown in Table 4.

	Recommended	Not Recommended
Used	True-Positive (tp)	False-Negative (fn)
Not used	False-Positive (fp)	True-Negative (tn)

Table 4 Classification of the possible result of a recommendation of an item to a user.

Based on table 4 the precision and recall can be calculated by the expression:

$$precision = \frac{\#\text{tp}}{\#\text{tp} + \#\text{fp}} \quad (8)$$

$$Recall = \frac{\#\text{tp}}{\#\text{tp} + \#\text{fn}} \quad (9)$$

One shortcoming of this evaluation approach is that we are forced to assume that unused items are not interesting to users and they will not be used even if the user has seen them, but this is not completely true. In some cases the user does not consume because he does not know that such an item exists but when he is made aware of it he will consume it.

Measures that summarize the precision recall are ROC curve such as F-measure [83]and the Area Under the ROC Curve (AUC) [84], which are useful for comparing algorithms independently of application. Half-life Utility Metric [27], can also be used in this kind of evaluator in attempting to evaluate the utility of a ranked list to the user.

The third family of evaluators used in the literature is the Ranking Measures, like the Discounted Cumulative Gain (DCG), and Normalized Discounted Cumulative Gain (NDCG) –once again, two metrics form the information retrieval domain. In DCG, positions of recommended items are discounted logarithmically. Assuming each user u has a "gain" *guy* from being recommended an item i, the average DCG for a list of J items is defined as [32],

$$DCG = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{J}\frac{g_{ui_j}}{\max(1, log_b j)} \quad (10)$$

$$NDCG = \frac{DCG}{DCG^*} \quad (11)$$

Where $DCG*$ is the ideal DCG. Other metrics to evaluate the ranking like The R-Score metric [27].

Accuracy has been the main factor used to compare recommender systems, but a high accuracy level cannot guarantee user satisfaction; for example, in a supermarket it would be too accurate to recommend to people to buy bananas, most of them will end up buying bananas because people do this frequently anyway. This example can summarize some problems in depending only on accuracy, relying on recommending already well known items

and resulting in poor user satisfaction and no increase in the supermarket income. That is why other aspects are also important in the recommendation evaluation.

5.2. User Satisfaction Evaluation

The user satisfaction evaluation includes evaluating different elements about the user experience with the recommender system:

- *Confidence:* It refers to how much the recommender system has confidence in an item; usually it is referred to by a value (star or number). This is a delicate issue as even if the recommender ranked the list of recommended items well, the system can fail if the prediction ratings it displayed to the user were incorrect [32].
- *Trust:* How much the user trusts the recommender system. This aspect varies between users. Some users trust recommenders that offer them new items, while others prefer having some of their preferred items in the recommendation list in order to trust the system [85], [104] . One main way to build the trust is to explain the recommendation to the user.
- *Diversity:* Usually users prefer systems with high diversity as they can explore new horizons of items [29].
- *Robustness:* Users need to be protected from fraud in systems with user participation. [86].
- *Novelty:* Recommending new items to users is essential in any recommendation system [10].
- *Serendipity:* A serendipitous recommendation helps the user find a surprisingly interesting item he might not have otherwise discovered. If a user likes an actor, a novel recommendation is this actor's new film while a serendipity recommendation is done by choosing another film that contains an actor who might be interesting to the user. The difference between novelty and serendipity is explained further in [32].

5.3. Communicating Recommendations to Users

One aspect of a good recommendation is how to convey the recommendation results to users, i.e. explain why the user has received a recommended item. Recommendation systems usually work like a "Black Box", they offer recommendations as a list without explaining why they recommend these items. However, explaining can help user make a better decision [87], [88], [104]; further, this aspect may affect the uses trust in the system [89].

Explaining recommendations for users might be for different goals. In [87] the authors identify seven potential goals for explaining recommendations: show the transparency of the recommendation system, enable scrutablity of the results, improve the trustworthiness and the persuasiveness of the recommendations, increase effectiveness for supporting decisions, and enhance efficiency for decision making and create satisfaction in the recommender systems.

One of the first systems to evaluate recommendations' explanations is in [89]. The study focuses on the fact that although users' enjoyment of recommended items did not change, users showed a high interest in the explanation of their recommendation. Another study pointed to the fact that recommendation explanation increases the user's satisfaction in the systems by helping him to know what is he expecting from the system. [72]. A good example is in [90]: giving labels to recommendations affected the recommendation where the click through rate (CTR) varied from 5.93%, to 8.86% and 9.87% according the change in labels, from nothing, to sponsored to organic.

Commercial interest is in increasing the sale amount, (the number of items sold and not returned) that is why this increase in the satisfaction without a change in the users' communication behavior might not interest the commercials.

5.4. Online Towards Offline Evaluation:

Evaluation varies between offline and online evaluation. Offline evaluation includes applying the evaluation metrics on data sets, either collected or available in the internet. Online evaluation is about how to get in contact with users and ask them to try the system, or give their live feedback, or asking them to answer a questionnaire.

In 2004 [32], most of the recommender systems were evaluated offline, one decade later, in 2013 [82], recommendation approaches are still tested mainly offline. From the same study in 2013: 69% of the evaluated works were using an offline evaluation, 7% were evaluated in real-world systems with an online evaluation [91], [92], [93], [51], [94] and two approaches 3% were evaluated using a qualitative user study [95], [96].

The authors observed that researchers cannot have access to real-world systems in order to test their works, and those who can evaluate in live scenarios usually do not. Like C. Lee Giles and his co-authors, who have important contributions in the field [97], [98], [99], [100], [101], [102], [103], they can perform real tests on their academic search engine CiteSeer, but they choose to perform offline evaluations instead.

One reason might be the simplicity of performing an offline evaluation in a couple of minutes or hours instead a couple of days or weeks with an online test. Another factor might be that in many cases tests in offline scenarios give better results and are more convenient than online tests and user studies [82].

From a commercial point of view, all the previous metrics would not be that important towards some commercial aspects like selling diverse items and increasing the number of sold items. The important commercial aspect is enabling users to find interesting items, which are hard to find without a recommender.

Increasing user fidelity by recognizing old customers and treating them as valuable users, for example by offering to them recommendation related to their previous visit. [Rogers 2001].

We conclude that evaluating a recommender system is a difficult issue. How should one integrate all the previous aspects? Knowing why the system has been designed and what its goals are help one to choose the best evaluation methodology.

6. Recommendation System Tools

Many implementations of recommender algorithms are available, especially for collaborative filtering algorithms. A lot of tools are free, open source projects that researchers can use. Tools vary depending on what they offer; for example, Crab, easyrec, MyMediaLite and Vogoo PHP LIB offer simple recommendation systems that can be integrated into web sites without a lot of effort. LensKit, provides researchers with reference implementations for common collaborative filtering algorithms using Java. Cofi as well provides a Java package that implements many collaborative filtering algorithms. Apache Mahout is a machine learning library and its main goal is to offer scalable algorithms implementation for large data sets, between other algorithms it includes an implementation of collaborative filtering algorithms -formerly developed under the name Taste-. Some tools focuse on one kind of recommender, like the SVDFeature, which focuses only on matrix factorization.

RecommenderLab is an R extension package that provides a general research infrastructure for recommender systems. It has a completely different goal from existing software packages, as it is not a library dedicated to the creation of recommender applications; instead its focus is on consistent and efficient data handling, easy incorporation of algorithms. In our conducted experiments we adopted R as it is effective, and easy to use.

7. Conclusion

Recommendation is a difficult domain where different aspects are gathered in order to offer to users the highest satisfaction. Covering all the existing work in the field of recommendation is beyond the scope of this chapter, instead we have explained the whole domain concisely from collaborative filtering, to content-based recommendation including semantic analysis, a discussion of evaluation metrics, defining and investigating the cold start problem, investigating the SLiR documents, and finally discussing the datasets used for evaluation followed by the introduction of some recommendation tools. We attempted to avoid details; at the same time, we had to explain some aspects that we will need in next chapters.

Returning to the Martin keynote, we see that algorithms and techniques have evolved significantly over the last two and a half decades. A variety of metrics to measure also have evolved. Now the focus should be about how to get benefits from the best of this domain in order to increase users' satisfaction, which takes place in an online, tangible world.

As we end this chapter, we will begin the next one with social recommenders, and introduce the social networks and social media aspect.

8. References

[1] F. Martin, « Recsys'09 industrial keynote: top 10 lessons learned developing deploying and operating real-world recommender systems », présenté à RecSys '09: Proceedings of the third ACM conference on Recommender systems, 2009, p. 1‑2.

[2] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, et J. Riedl, « GroupLens: an open architecture for collaborative filtering of netnews », in *Proceedings of the 1994 ACM conference on Computer supported cooperative work*, 1994, p. 175–186.

[3] U. Shardanand, « Social information filtering for music recommendation », Massachusetts Institute of Technology, 1994.

[4] W. Hill, L. Stead, M. Rosenstein, et G. Furnas, « Recommending and evaluating choices in a virtual community of use », in *Proceedings of the SIGCHI conference on Human factors in computing systems*, 1995, p. 194–201.

[5] K. Goldberg, T. Roeder, D. Gupta, et C. Perkins, « Eigentaste: A constant time collaborative filtering algorithm », *Inf. Retr.*, vol. 4, n° 2, p. 133–151, 2001.

[6] J. B. Schafer, J. A. Konstan, et J. Riedl, « E-commerce recommendation applications », in *Applications of Data Mining to Electronic Commerce*, Springer, 2001, p. 115–153.

[7] B. Smyth, « Case-based recommendation », in *The adaptive web*, Springer, 2007, p. 342–376.

[8] R. Burke, « Hybrid recommender systems: Survey and experiments », *User Model. User-Adapt. Interact.*, vol. 12, n° 4, p. 331–370, 2002.

[9] J. John, « Pandora and the music genome project », *Sci. Comput.*, vol. 23, n° 10, p. 40–41, 2006.

[10] F. Ricci, L. Rokach, et B. Shapira, *Introduction to recommender systems handbook*. Springer, 2011.

[11] R. Mihalcea et A. Csomai, « Wikify!: linking documents to encyclopedic knowledge », in *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, 2007, p. 233–242.

[12] Ò. Celma, « Foafing the music: Bridging the semantic gap in music recommendation », in *The Semantic Web-ISWC 2006*, Springer, 2006, p. 927–934.

[13] B. Sheth et P. Maes, « Evolving agents for personalized information filtering », in *Artificial Intelligence for Applications, 1993. Proceedings., Ninth Conference on*, 1993, p. 345–352.

[14] Y. Blanco-Fernandez, J. Pazos-Arias, A. Gil-Solla, M. Ramos-Cabrer, et M. Lopez-Nores, « Providing entertainment by content-based filtering and semantic reasoning in intelligent recommender systems », *Consum. Electron. IEEE Trans. On*, vol. 54, n° 2, p. 727–735, 2008.

[15] P. Basile, M. de Gemmis, A. L. Gentile, P. Lops, et G. Semeraro, « UNIBA: JIGSAW algorithm for word sense disambiguation », in *Proceedings of the 4th International Workshop on Semantic Evaluations*, 2007, p. 398–401.

[16] M. Baldauf, S. Dustdar, et F. Rosenberg, « A survey on context-aware systems », *Int. J. Ad Hoc Ubiquitous Comput.*, vol. 2, n° 4, p. 263–277, 2007.

[17] M. Khabbaz et L. V. Lakshmanan, « TopRecs: Top-k algorithms for item-based collaborative filtering », in *Proceedings of the 14th International Conference on Extending Database Technology*, 2011, p. 213–224.

[18] X. Su et T. M. Khoshgoftaar, « A survey of collaborative filtering techniques », *Adv. Artif. Intell.*, vol. 2009, p. 4, 2009.

[19] H. Ma, H. Yang, M. R. Lyu, et I. King, « Sorec: social recommendation using probabilistic matrix factorization », in *Proceedings of the 17th ACM conference on Information and knowledge management*, 2008, p. 931–940.

[20] L. M. de Campos, J. M. Fernández-Luna, J. F. Huete, et M. A. Rueda-Morales, « Combining content-based and collaborative recommendations: A hybrid approach based on Bayesian networks », *Int. J. Approx. Reason.*, vol. 51, n° 7, p. 785–799, 2010.

[21] T. Hofmann, « Latent semantic models for collaborative filtering », *ACM Trans. Inf. Syst. TOIS*, vol. 22, n° 1, p. 89–115, 2004.

[22] M. D. Ekstrand, J. T. Riedl, et J. A. Konstan, « Collaborative filtering recommender systems », *Found. Trends Hum.-Comput. Interact.*, vol. 4, n° 2, p. 81–173, 2011.

[23] G. Karypis, « Evaluation of item-based top-n recommendation algorithms », in *Proceedings of the tenth international conference on Information and knowledge management*, 2001, p. 247–254.

[24] B. Sarwar, G. Karypis, J. Konstan, et J. Riedl, « Item-based collaborative filtering recommendation algorithms », in *Proceedings of the 10th international conference on World Wide Web*, 2001, p. 285–295.

[25] G. Linden, B. Smith, et J. York, « Amazon. com recommendations: Item-to-item collaborative filtering », *Internet Comput. IEEE*, vol. 7, n° 1, p. 76–80, 2003.

[26] J. Herlocker, J. A. Konstan, et J. Riedl, « An empirical analysis of design choices in neighborhood-based collaborative filtering algorithms », *Inf. Retr.*, vol. 5, n° 4, p. 287–310, 2002.

[27] J. S. Breese, D. Heckerman, et C. Kadie, « Empirical analysis of predictive algorithms for collaborative filtering », in *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence*, 1998, p. 43–52.

[28] D. Mladenic, « Text-learning and related intelligent agents: a survey », *IEEE Intell. Syst.*, vol. 14, n° 4, p. 44–54, 1999.

[29] C.-N. Ziegler, S. M. McNee, J. A. Konstan, et G. Lausen, « Improving recommendation lists through topic diversification », in *Proceedings of the 14th international conference on World Wide Web*, 2005, p. 22–32.

[30] M. Anderson, M. Ball, H. Boley, S. Greene, N. Howse, S. McGrath, et D. Lemire, « Racofi: A rule-applying collaborative filtering system », in *Proceedings of COLA*, 2003.

[31] Z. Xia, Y. Dong, et G. Xing, « Support vector machines for collaborative filtering », in *Proceedings of the 44th annual Southeast regional conference*, 2006, p. 169–174.

[32] J. L. Herlocker, J. A. Konstan, L. G. Terveen, et J. T. Riedl, « Evaluating collaborative filtering recommender systems », *ACM Trans. Inf. Syst. TOIS*, vol. 22, n° 1, p. 5–53, 2004.
[33] D. Billsus et M. J. Pazzani, « User modeling for adaptive news access », *User Model. User-Adapt. Interact.*, vol. 10, n° 2‑3, p. 147–180, 2000.
[34] Y. Zhang, J. Callan, et T. Minka, « Novelty and redundancy detection in adaptive filtering », in *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, 2002, p. 81–88.
[35] R. Baeza-Yates, B. Ribeiro-Neto, et others, *Modern information retrieval*, vol. 463. ACM press New York, 1999.
[36] H. Lieberman et others, « Letizia: An agent that assists web browsing », *IJCAI 1*, vol. 1995, p. 924–929, 1995.
[37] D. Mladenic et B. A. P. Webwatcher, « Machine learning used by Personal WebWatcher », 1999.
[38] M. Pazzani et D. Billsus, « Learning and revising user profiles: The identification of interesting web sites », *Mach. Learn.*, vol. 27, n° 3, p. 313–331, 1997.
[39] M. J. Pazzani, J. Muramatsu, D. Billsus, et others, « Syskill & Webert: Identifying interesting web sites », in *AAAI/IAAI, Vol. 1*, 1996, p. 54–61.
[40] F. A. Asnicar et C. Tasso, « ifWeb: a prototype of user model-based intelligent agent for document filtering and navigation in the world wide web », in *Proceedings of WorkshopAdaptive Systems and User Modeling on the World Wide Web'at 6th International Conference on User Modeling, UM97, Chia Laguna, Sardinia, Italy*, 1997, p. 3–11.
[41] A. Moukas, « Amalthaea information discovery and filtering using a multiagent evolving ecosystem », *Appl. Artif. Intell.*, vol. 11, n° 5, p. 437–457, 1997.
[42] L. Chen et K. Sycara, « WebMate: a personal agent for browsing and searching », in *Proceedings of the second international conference on Autonomous agents*, 1998, p. 132–139.
[43] H. Sorensen, A. O'Riordan, et C. O'Riordan, « Profiling with the informer text filtering agent », *J. Univers. Comput. Sci.*, vol. 3, n° 8, p. 988–1006, 1997.
[44] D. Billsus et M. J. Pazzani, "A hybrid user model for news story classification". *Proceeding UM '99 Proceedings of the seventh international conference on User modeling Pages 99-108 Springer,* 1999.
[45] J. Ahn, P. Brusilovsky, J. Grady, D. He, et S. Y. Syn, « Open user profiles for adaptive news systems: help or harm? », in *Proceedings of the 16th international conference on World Wide Web*, 2007, p. 11–20.
[46] P. Lops, M. De Gemmis, et G. Semeraro, « Content-based recommender systems: State of the art and trends », in *Recommender systems handbook*, Springer, 2011, p. 73–105.
[47] B. Magnini et C. Strapparava, « Improving user modelling with content-based techniques », in *User Modeling 2001*, Springer, 2001, p. 74–83.
[48] M. Degemmis, P. Lops, et G. Semeraro, « A content-collaborative recommender that exploits WordNet-based user profiles for neighborhood formation », *User Model. User-Adapt. Interact.*, vol. 17, n° 3, p. 217–255, 2007.
[49] G. Semeraro, P. Basile, M. de Gemmis, et P. Lops, *User Profiles for Personalizing Digital Libraries.* 2009.
[50] M. Eirinaki, M. Vazirgiannis, et I. Varlamis, « SEWeP: using site semantics and a taxonomy to enhance the Web personalization process », in *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2003, p. 99–108.
[51] S. E. Middleton, N. R. Shadbolt, et D. C. De Roure, « Ontological user profiling in recommender systems », *ACM Trans. Inf. Syst. TOIS*, vol. 22, n° 1, p. 54–88, 2004.
[52] I. Cantador, A. Bellogín, et P. Castells, « News@ hand: A semantic web approach to recommending news », in *Adaptive Hypermedia and Adaptive Web-Based Systems*, 2008, p. 279–283.
[53] P. Basile, M. Degemmis, A. L. Gentile, L. Iaquinta, et P. Lops, « The JUMP project: Domain Ontologies and Linguistic Knowledge@ Work. », in *SWAP*, 2007.
[54] E. Gabrilovich et S. Markovitch, « Overcoming the brittleness bottleneck using Wikipedia: Enhancing text categorization with encyclopedic knowledge », in *AAAI*, 2006, vol. 6, p. 1301–1306.
[55] E. Gabrilovich et S. Markovitch, « Computing Semantic Relatedness Using Wikipedia-based Explicit Semantic Analysis. », in *IJCAI*, 2007, vol. 7, p. 1606–1611.
[56] J. Lees-Miller, F. Anderson, B. Hoehn, et R. Greiner, « Does wikipedia information help netflix predictions? », in *Machine Learning and Applications, 2008. ICMLA'08. Seventh International Conference on*, 2008, p. 337–343.
[57] G. Semeraro, P. Lops, P. Basile, et M. de Gemmis, « Knowledge infusion i
\nto content-based recommender systems », in *Proceedings of the third ACM conference on Recommender systems*, 2009, p. 301–304.
[58] A. V. Smirnov et A. A. Krizhanovsky, « Information filtering based on wiki index database », *ArXiv Prepr. ArXiv08042354*, 2008.
[59] R. Burke, « Hybrid web recommender systems », in *The adaptive web*, Springer, 2007, p. 377–408.
[60] M. Balabanović et Y. Shoham, « Fab: content-based, collaborative recommendation », *Commun. ACM*, vol. 40, n° 3, p. 66–72, 1997.
[61] T. Joachims, D. Freitag, T. Mitchell, et others, « Webwatcher: A tour guide for the world wide web », in *IJCAI (1)*, 1997, p. 770–777.
[62] M. Claypool, A. Gokhale, T. Miranda, P. Murnikov, D. Netes, et M. Sartin, « Combining content-based and collaborative filters in an online newspaper », in *Proceedings of ACM SIGIR workshop on recommender systems*, 1999, vol. 60.
[63] A. M. Ahmad Wasfi, « Collecting user access patterns for building user profiles and collaborative filtering », in *Proceedings of the 4th international conference on Intelligent user interfaces*, 1998, p. 57–64.
[64] B. Smyth et P. Cotter, « A personalised TV listings service for the digital TV age », *Knowl.-Based Syst.*, vol. 13, n° 2‑3, p. 53‑59, avr. 2000.

[65] P. Melville, R. J. Mooney, et R. Nagarajan, « Content-boosted collaborative filtering for improved recommendations », in *AAAI/IAAI*, 2002, p. 187–192.
[66] J. Salter et N. Antonopoulos, « CinemaScreen recommender agent: combining collaborative and content-based filtering », *Intell. Syst. IEEE*, vol. 21, n° 1, p. 35–41, 2006.
[67] P. Symeonidis, « Content-based Dimensionality Reduction for Recommender Systems », in *Data Analysis, Machine Learning and Applications*, Springer, 2008, p. 619–626.
[68] A. I. Schein, A. Popescul, L. H. Ungar, et D. M. Pennock, « Methods and metrics for cold-start recommendations », in *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, 2002, p. 253–260.
[69] T. Qiu, G. Chen, Z.-K. Zhang, et T. Zhou, « An item-oriented recommendation algorithm on cold-start problem », *EPL Europhys. Lett.*, vol. 95, n° 5, p. 58003, 2011.
[70] A. M. Rashid, I. Albert, D. Cosley, S. K. Lam, S. M. McNee, J. A. Konstan, et J. Riedl, « Getting to know you: learning new user preferences in recommender systems », in *Proceedings of the 7th international conference on Intelligent user interfaces*, 2002, p. 127–134.
[71] P. Massa et B. Bhattacharjee, « Using trust in recommender systems: an experimental analysis », in *Trust Management*, Springer, 2004, p. 221–235.
[72] X. N. Lam, T. Vu, T. D. Le, et A. D. Duong, « Addressing cold-start problem in recommendation systems », in *Proceedings of the 2nd international conference on Ubiquitous information management and communication*, 2008, p. 208–211.
[73] S.-T. Park et W. Chu, « Pairwise preference regression for cold-start recommendation », in *Proceedings of the third ACM conference on Recommender systems*, 2009, p. 21–28.
[74] G. Shaw, Y. Xu, et S. Geva, « Using association rules to solve the cold-start problem in recommender systems », in *Advances in Knowledge Discovery and Data Mining*, Springer, 2010, p. 340–347.
[75] K. Zhou, S.-H. Yang, et H. Zha, « Functional matrix factorizations for cold-start recommendation », in *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, 2011, p. 315–324.
[76] N. Golbandi, Y. Koren, et R. Lempel, « Adaptive bootstrapping of recommender systems using decision trees », in *Proceedings of the fourth ACM international conference on Web search and data mining*, 2011, p. 595–604.
[77] Z.-K. Zhang, C. Liu, Y.-C. Zhang, et T. Zhou, « Solving the cold-start problem in recommender systems with social tags », *EPL Europhys. Lett.*, vol. 92, n° 2, p. 28002, 2010.
[78] H. Gao, J. Tang, et H. Liu, « Addressing the cold-start problem in location recommendation using geo-social correlations », *Data Min. Knowl. Discov.*, p. 1–25, 2014.
[79] B. Lika, K. Kolomvatsos, et S. Hadjiefthymiades, « Facing the cold start problem in recommender systems », *Expert Syst. Appl.*, vol. 41, n° 4, p. 2065–2073, 2014.
[80] G. Shani et A. Gunawardana, « Evaluating recommendation systems », in *Recommender systems handbook*, Springer, 2011, p. 257–297.
[81] P. Zigoris et Y. Zhang, « Bayesian adaptive user profiling with explicit & implicit feedback », in *Proceedings of the 15th ACM international conference on Information and knowledge management*, 2006, p. 397–404.
[82] J. Beel, S. Langer, M. Genzmehr, B. Gipp, C. Breitinger, et A. Nürnberger, « Research paper recommender system evaluation: a quantitative literature survey », in *Proceedings of the International Workshop on Reproducibility and Replication in Recommender Systems Evaluation*, 2013, p. 15–22.
[83] C. J. Van Rijsbergen, « Information Retrieval. Dept. of Computer Science, University of Glasgow », *URL Citeseer Ist Psu Eduvanrijsbergen79information Html*, 1979.
[84] D. Bamber, « The area above the ordinal dominance graph and the area below the receiver operating characteristic graph », *J. Math. Psychol.*, vol. 12, n° 4, p. 387–415, 1975.
[85] M. Montaner, B. López, et J. L. de la Rosa, « Developing trust in recommender agents », in *Proceedings of the first international joint conference on Autonomous agents and multiagent systems: part 1*, 2002, p. 304–305.
[86] J. A. Konstan et J. Riedl, « Recommender systems: from algorithms to user experience », *User Model. User-Adapt. Interact.*, vol. 22, n° 1‑2, p. 101–123, 2012.
[87] N. Tintarev et J. Masthoff, « A survey of explanations in recommender systems », in *Data Engineering Workshop, 2007 IEEE 23rd International Conference on*, 2007, p. 801–810.
[88] N. Tintarev et J. Masthoff, « Effective explanations of recommendations: user-centered design », in *Proceedings of the 2007 ACM conference on Recommender systems*, 2007, p. 153–156.
[89] J. L. Herlocker, J. A. Konstan, et J. Riedl, « Explaining collaborative filtering recommendations », in *Proceedings of the 2000 ACM conference on Computer supported cooperative work*, 2000, p. 241–250.
[90] M. Dobreva, C. Papatheodorou, G. Tsakonas, et C. Farrugia, « Sponsored vs. Organic (Research Paper) Recommendations and the Impact of Labeling », in *International Conference on Theory and Practice of Digital Libraries, pp391-395, Valletta, Malta,* 2013.
[91] J. Beel, S. Langer, M. Genzmehr, et A. Nürnberger, « Introducing Docear's research paper recommender system », in *Proceedings of the 13th ACM/IEEE-CS joint conference on Digital libraries*, 2013, p. 459–460.
[92] T. Bogers et A. Van den Bosch, « Recommending scientific articles using citeulike », in *Proceedings of the 2008 ACM conference on Recommender systems*, 2008, p. 287–290.
[93] M. Mönnich et M. Spiering, « Adding value to the library catalog by implementing a recommendation system », *-Lib Mag.*, vol. 14, n° 5, p. 4, 2008.
[94] J. Bollen et H. Van de Sompel, « An architecture for the aggregation and analysis of scholarly usage data », in *Proceedings of the 6th ACM/IEEE-CS joint conference on Digital libraries*, 2006, p. 298–307.

[95] K. Stock, A. Robertson, F. Reitsma, T. Stojanovic, M. Bishr, D. Medyckyj-Scott, et J. Ortmann, « eScience for Sea Science: A Semantic Scientific Knowledge Infrastructure for Marine Scientists », in *e-Science, 2009. e-Science'09. Fifth IEEE International Conference on*, 2009, p. 110–117.

[96] K. Stock, V. Karasova, A. Robertson, G. Roger, M. Small, M. Bishr, J. Ortmann, T. Stojanovic, F. Reitsma, L. Korczynski, et others, « Finding Science with Science: Evaluating a Domain and Scientific Ontology User Interface for the Discovery of Scientific Resources », *Trans. GIS*, vol. 17, n° 4, p. 612–639, 2013.

[97] Q. He, D. Kifer, J. Pei, P. Mitra, et C. L. Giles, « Citation recommendation without author supervision », in *Proceedings of the fourth ACM international conference on Web search and data mining*, 2011, p. 755–764.

[98] L. Rokach, P. Mitra, S. Kataria, W. Huang, et L. Giles, « A supervised learning method for context-aware citation recommendation in a large corpus », *Invit. Speak. Anal. Perform. Top-K Retr. Algorithms*, p. 1978, 2013.

[99] Q. He, J. Pei, D. Kifer, P. Mitra, et L. Giles, « Context-aware citation recommendation », in *Proceedings of the 19th international conference on World wide web*, 2010, p. 421–430.

[100] C. Caragea, A. Silvescu, P. Mitra, et C. L. Giles, « Can't see the forest for the trees?: a citation recommendation system », in *Proceedings of the 13th ACM/IEEE-CS joint conference on Digital libraries*, 2013, p. 111–114.

[101] D. Zhou, S. Zhu, K. Yu, X. Song, B. L. Tseng, H. Zha, et C. L. Giles, « Learning multiple graphs for document recommendations », in *Proceedings of the 17th international conference on World Wide Web*, 2008, p. 141–150.

[102] D. M. Pennock, E. Horvitz, S. Lawrence, et C. L. Giles, « Collaborative filtering by personality diagnosis: A hybrid memory-and model-based approach », in *Proceedings of the Sixteenth conference on Uncertainty in artificial intelligence*, 2000, p. 473–480.

[103] W. Huang, S. Kataria, C. Caragea, P. Mitra, C. L. Giles, et L. Rokach, « Recommending citations: translating papers into references », in *Proceedings of the 21st ACM international conference on Information and knowledge management*, 2012, p. 1910–1914.

[104] Gaillard J., El Bèze M., Altman E., Ethis E.. « Well-Argued Recommendation: Adaptive Models Based on Words in Recommender Systems. EMNLP », in - *Conference on Empirical Methods in Natural Language Processing,* Oct 2013, Seattle, Washington, United States. pp.1943-1947.

CHAPTER 3: RELATED WORK

-Social Media, Social Networks, Social Recommenders and Social Context-

Social media is more about sociology and psychology, than it is about technology"

- Anonymous

http://blog.shelbymcquilkin.com/2014/07/13/social-networking/

1. Introduction:

Based on the interaction of users on the web, social media have tremendously evolved in the last few years, resulting in a fundamental change in the way in which people live and consume information. The power is no longer with marketing or public relations teams; it is democratized so that every user can play a role in the web today.

Online social networks belong to social media site families and they become indispensable in people's daily lives; people from different countries and age groups use them on a daily basis. While people are spending time on social media sites, like social networks, they are leaving valuable information attracting researchers' attention.

One of the most attractive aspects for researchers resulting from this phenomenon is how to enhance recommendation based on this social information. A variety of works discusses and proposes different approaches for social recommendation; this chapter explores main works in this domain.

In this chapter we will discuss social media, online social networks, as well as social recommenders as the three disciplines are highly connected together; where the social recommendation can be considered as a part of the social media game. We will refer briefly to the fundamental works, focusing on those which we will need in next chapters, where we will build a social machine of recommendation. Mainly our work is related to social recommendation, but understanding the whole environment of social media and its relation with social recommendation helped us first to prove that our model can be useful in real cases when it transforms the general purpose social networks into a social machine that is first, second this understanding helped us to accomplish live evaluation using the social media channels (Twitter Avatars, and Facebook Applications).

From the other side context awareness refers to integrating information into different applications in order to trigger or perform actions based on changing context. Contextual information varies from the changing of time, location, mood weather, human surroundings, or any other conceivable and measurable change. The domain of recommendation is one branch that adopted contextual information in order to enhance the way in which it works. Integrating contextual elements into recommenders has proven to enhance these systems.

In this chapter we will also investigate context awareness and context-aware recommenders. We introduce the levels of context and define the situation as a high-level interpretation of context, as all these elements help us later in the iAmélie system, which uses one aspect of context which is the social context.

Some contextual information can be measured (by sensors for example) but others are more difficult to obtain, such as user mood and internal mood or surroundings; that is why we give equal attention to the following question: can social networks be a source of contextual information?

2. Social Media:

2.1. Social Media overview:

Social media refer to the social interaction among people in which they create, share or exchange information and ideas in virtual communities and networks [1]. Social media is based on conversations [2], user generated content, some conservation can spread and reach millions especially if the content was attractive, or surprising for people [3], [4].

Social media sites began very early on the web. For example, in 1997, Sixdegrees allowed users to create profiles, create friends list, and add friends-of-friends to one's own lists [2]. Actually different types of social media sites already exist in the web today. Some of them are for general mass like Hi5, Friendster, Facebook, Pinterest. While others are specialized in some domains like: LinkedIn for professional profiles, MySpace, YouTube, and Flickr for sharing videos and photos, FarmVille for social games, and TripAdvisor for places and travels opinions.

Social media users can create social media profile cards, like what they do in the business card concept, using tools like Retagger in order to promote their accounts on different platforms which will encourage an increase of followers on those accounts.

Blogging sites raised in late 1990, to become extremely popular. Their authors range from ordinary users to expert users or even celebrities. Blogging sites are easy to create and maintain. In 31/07/2014 the number of blogs was 751 million[8]. Blogs become a source of public opinions and many engines like Technorati were created to search blogs.

Social media also opened the door to rank sites by users' votes on the content, using the bookmarking sites like Reddit, Digg, and Delicious[9].

Micro blogging is another aspect of social media that offers real time updates; the most popular of which is Twitter. Users send short messages (Tweets) that are limited to 140 characters and are mostly short status updates of what users are doing, where they are, how they are feeling, links to other sites, photos, or short videos.

Sites like Foursquare utilize a blend of real-time updates and location specific information by rewarding users for 'checking in' to at any location worldwide and for leaving their feedback for that specific location for others to view.

We can conclude that with the rise of social media, the corporate communication has been democratized. Marketing and public relation no longer hold the power, it is now in the hands of individuals and communities that create content over the different social media channels with or without the permission of companies. Companies can no longer ignore the effect of social media. One classical example is in the broken guitar of Dave Carroll by United Airlines. Carroll is a country music Canadian musician. Based on the subsequent reaction from the airline company towards the broken guitar Carroll recorded a music video, and then he shared it on YouTube at the time and gathered more than 14 million views. The financial impact of the United Breaks Guitars is estimated at $180m![10]

To conclude: the positive participation of people in the web towards a brand can be considered free marketing that leads to growing brand recognition, increasing sales and so on [5] while negative feedback of people can cause costly damages [6][7].

As BBC Business Editor Tim Weber (2010) explains: "These days, one witty tweet, one clever blog post, one devastating video——forwarded to hundreds of friends at the click of a mouse——can snowball and kill a product or damage a company's share price."

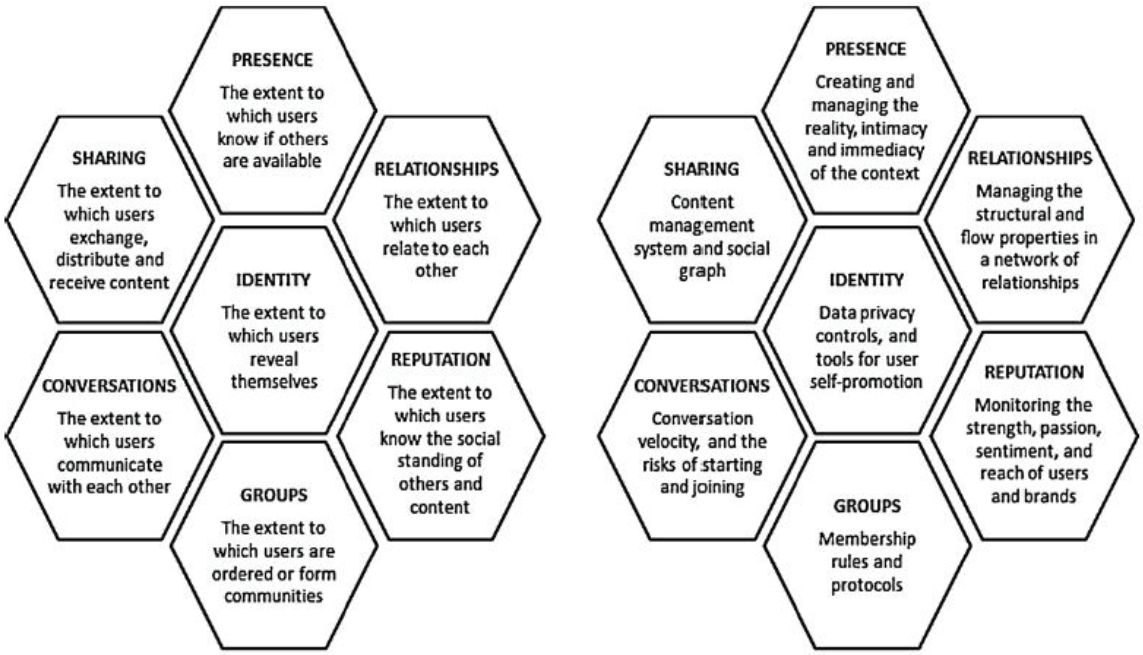
Some works have questioned opportunities and threats of social media [8], trying to understand it better in order to better benefit from its existence. Calling this effect of people's interaction, eWoM (electronic Word of Mouth), as an electronic version of Word of Mouth (WoM).

[8] http://www.internetlivestats.com/
[9] formerly known as Del.icio.us
[10] Revenge is best served cold—on YouTube https://www.youtube.com/watch?v=5YGc4zOqozo

The original definition of WoM came about in 1967 as [9] as the "oral, person to person communication between a receiver and a communicator whom the receiver perceives anon-commercial, concerning a brand, a product or a service". Since 1969 [10] authors found that WoM has much larger effect on purchasing decisions than marketing tools and conventional advertising media.

Then many studies came to affirm the main assumption of [10] about the effect of WoM on the purchase decisions, like recently in [11] at 2009.The widespread use of social media sites and with the increase in people who use the concept, eWoM appeared first in 1998 in [12] and recently in 2009 in [13]. eWoM is less personal of course than traditional WoM, but it is more powerful because it is immediate, has a significant reach, is credible, and is publicly available. Some parameters may play a role in eWoM like demographics or psychographics of users studied [14].

Although eWoM influences purchase decisions, from which movie to watch to what stocks to buy but the way in which it work in not yet well understood [15].



Social media functionality *implications of the functionality*

Figure 3 from [16], the honeycomb of social media

In terms of social media in general, understanding eWOM can help to further benefit from it regarding audiences and their needs. In [16] "Introduce the honeycomb of social media" Figure 3, as a framework of seven social media building blocks. These seven blocks can help to understand both the existing social media sites, as well as the needs of organizations.

Figure 3 presents the seven blocks of the honeycomb [16][1]. Which are: identity, conversations, sharing, presence, relationships, reputation, and groups. Each block focuses on a specific facet of social media user's experience and its implications for firms and organizations. These blocks vary in their presence from one social network to another.

2.2. Social media blocks:

- **Identity:** It represents the limits to which users put in social media information related to their identities. Like the name, age, gender, profession, location, or even some internal information like feelings, likes, and dislikes. However social media sites offer different possibilities in adding identity information.
- **Conversations:** It is a main block in social media; people are here to communicate with other users and groups. People Tweet, post, blog, look to meet new likeminded people to find true love, to find an ideal book, or even to build self-esteem or to discover new ideas or trending topics.
- **Sharing:** Sharing is about the breadth to which people exchange, distribute, and receive content. However sharing alone is a way to connect in social media, the question is can this lead to build a relation between people resulting from this interact? This depends between many elements on the functionality of the social media sites, and the shared component, like photos on Flickr, and Music on MySpace.
- **Presence:** Presence means to what end users can know about each other if they are accessible, the user is connected, not connected, busy or hidden… are some examples of how the case would be. As virtual and real worlds are highly connected in our days, this involves also where people are in the real world. Where were they or where are they going to. For example, Facebook offers the possibility to add all the places that the user had visited with the time in which he has visited each place.
- **Relationships:** Relationship represents to what level users are connected to each other's, which means they have a relation provided by the system, like being friends in Facebook, or follow each other on Twitter, in relationship an important aspect is to what extent the relation is real, in other words do two friends on Facebook really communicate there? Or they are simply only one of the friends list.
- **Reputation:** Reputation represents to what extent users can determine the standings of others (including themselves) using social media. Although reputation can be understood differently, but in most of cases it is considered as the trust. Trust is not an easy concept to measure; that is why social media relies on some metrics to automatically aggregate the user-generated information to determine trustworthiness. For example Katty Perry (@katyperry) with 54 million followers on Twitter, or Justin Bieber (@justinbieber) with 53 million followers have without any doubt a good reputation. LinkedIn works differently; the reputations of individuals are based on endorsements from others.
- **Groups:** The groups block indicates to what extent users can build communities and sub communities. When the network is more social, users can build more and bigger groups of friends, followers, an example is in Google + circles, or in Facebook groups of friends (school friends, work friends …).

Briefly we scanned the blocks of social media, in the next sections we'll focus on the online social networks as a rich branch of social media.

3. Social Networks:

Social networks are a fundamental part of the social media family sites. Online social networks like Facebook and Twitter continue to grow. Figure 4 shows the number of users

using some social networks within a month in the United States only. Facebook has 141 million users and Twitter has 93 million users compared to a mere 8 million on Flickr and 1 million on epinions. Amazon is not considered as a social network but instead is a recommender system. However it has been added in order to compare figures.

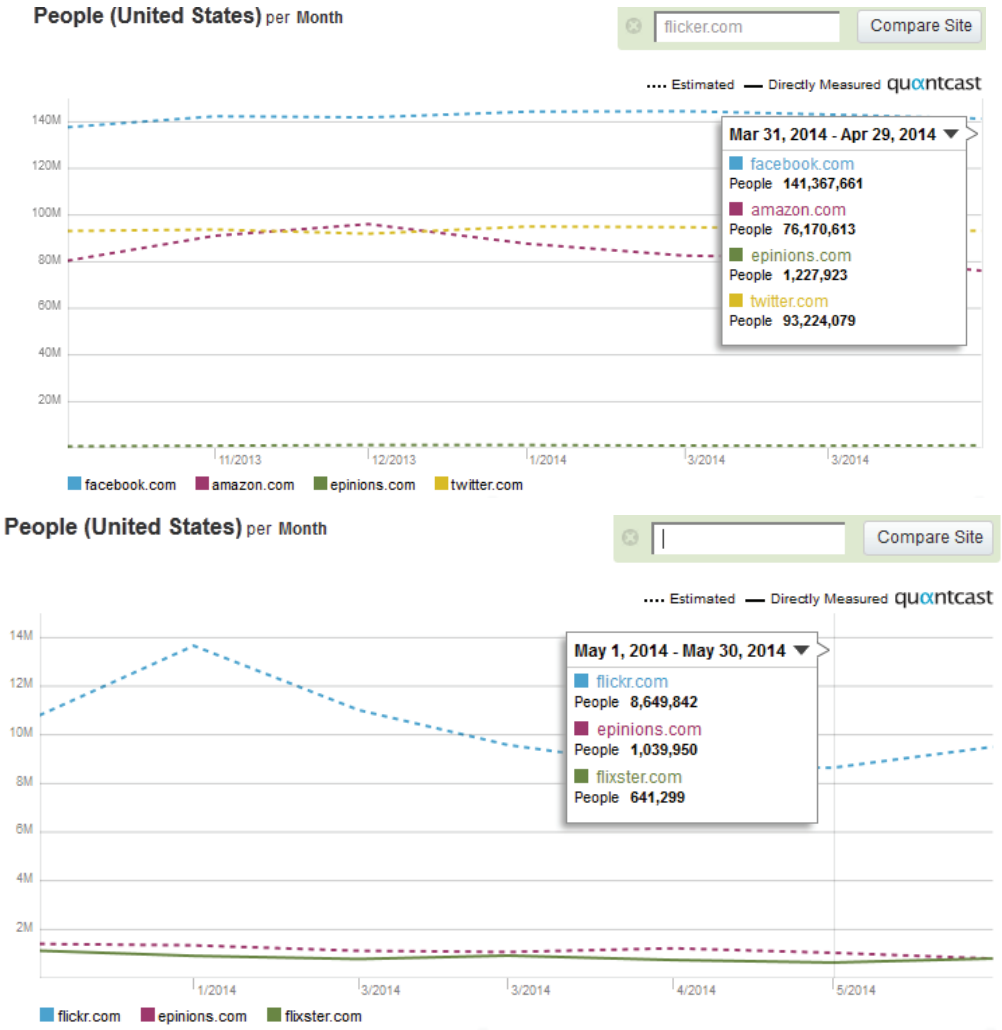


Figure 4 A comparison between different sites over one moth use in United States
https://www.quantcast.com/

Online social networks are divided into **G**eneral **P**urpose **S**ocial **N**etworks GPSN like Twitter and Facebook, towards **D**omain **B**ased **S**ocial **N**etworks DSSN like epinions for product recommendation, Flickr for photo, LinkedIn for professional relations and so on.

3.1. General Purpose Social Networks GPSN:

They are the social networks that do not have a domain associated with them (like films, products…). Users can do different actions in this kind of networks in different domains. Many GPSN exist like Classmates, MySpace, Facebook and Twitter. In [17], the author claims that one main factor for which users use social networks is looking for the lost physical space around people, as people live in small homes and they are very busy, so they try to overcome this lack of physical space and time in social networks which are out of the space & time limits.

- **Facebook:**

Facebook was founded in 2004; one decade later, it has the global rank of 2[11] and nearly 1.3 billion active users per month, 25% of which are from the United States[12]. Since there are 7.2 billion humans on Earth[13] ,this would mean that for every five humans, one would have a Facebook account! In addition, 48% of these users log in every day and each user has an average of 130 friends. More than 50 percent of the users broadcast information and knowledge via Facebook. And Finally 70 percent log in every time they start their computer or web reader.

According to [18] Facebook turned into a habit-forming activity and is a part of people's daily routine. Nearly half of the respondents announced that it is not easy to keep updated on top of things without Facebook. Facebooking (the use of Facebook) is sometimes considered as an unconscious habit. A majority of the respondents stated that they log in to their Facebook accounts every time they launch their web browser. Leif Denti, a doctoral student of Psychology at the University of Gothenburg stated that "this may even developed into an addiction."

The use of Facebook is somehow related to their commercial status, as people with low income and low-educated individuals spend more time on Facebook, and the worst fact is that women who use Facebook more are also report feeling less happy and less content with their lives. Facebook was not the first social network created but it is the most widely used to date. In Facebook, users can post phrases, comment on friends' post, and share videos and photos. They can also like items any kind, like books, films, restaurants. Additionally people can invite friends, and they can add all the places, countries and villages where the user has visited.

In his work [17] in 2009, the author tried to answer some opened question related to Facebook, for him there are three major elements for which Facebook is considered advanced:

- *Identity:* In Facebook users can write posts, like items, do actions that can reflect their identity, and they can see directly the feedback of friends
- *Relationship:* Facebook permits to users to maintain their relation with their friends, while simultaneously seeking new relations.
- *Community:* Since the user has established his relations of old and new friends, he now has a community in which he can establish his social position. The basic desire is simple and age-old: to be recognized as a valued member of one's various communities.

Responding to these three needs might be a main factor for why do people continue to join Facebook, to log in every day, moreover to put their important information about them, at the risk of losing privacy. As Grimmelmann investigated in [19], social networks are open for participation of anyone, so any information that people try to keep offline might find its way online by a friend by way of a tag in a photo, or a post on the user's wall. Like the author we refer to Google's Eric Schmidt when he said "If you have something that you don't want anyone to know, maybe you shouldn't be doing it in the first place." In legal, filings, his company has argued that "even in desert, complete privacy does not exist."!

[11] Alexa Rank
[12] http://www.statisticbrain.com/facebook-statistics/
[13] http://www.worldometers.info/world-population/

- **Twitter:**

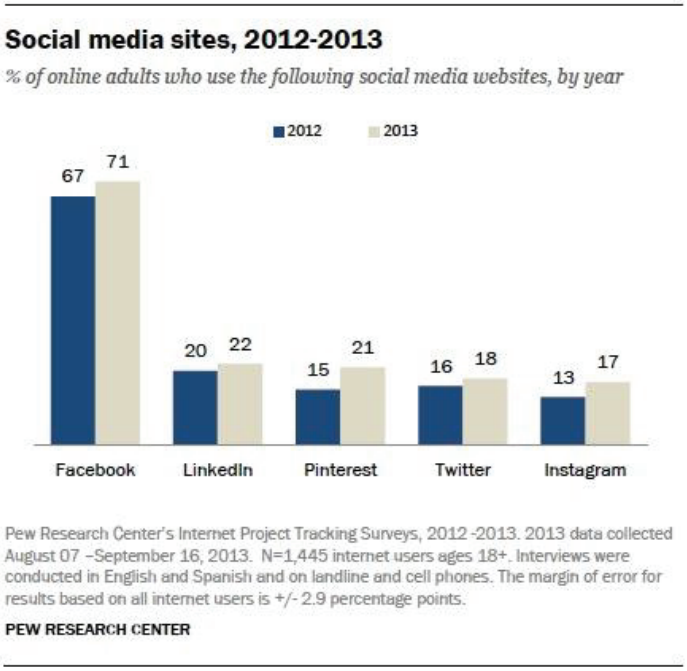
Twitter has nearly 645 million active users[14], 115 million are active every month, taking 5 days to produce 1 billion tweet! Twitter was founded in March 21, 2006 and is currently ranked #8 by Alexa.

Twitter is different from Facebook in the way in which it works; in Twitter a user can post short messages (limited to140 characters), he can follow other users, as well as he can be followed by other users. This is how the community is created on Twitter. Furthermore the user can share photos and videos in addition to "retweeting", replying, and "favoriting" tweets from other users.

Tweets can contain any information the user wishes to share. In general people use tweets to defuse information of what they are doing, where they are, how they are feeling, links to other sites, photos, or short videos, even for political and technical information.

Twitter reflects the trending topics instantly, not only by offering the trending topics but also by the participation of users who create tweets towards any news. Examples include the results of a sports match or an election and even news of Facebook being down for a few minutes.

The amount of information that exists on Twitter has caught the attention of researchers and they actually study these short messages.



[14] http://about.twitter.com/company

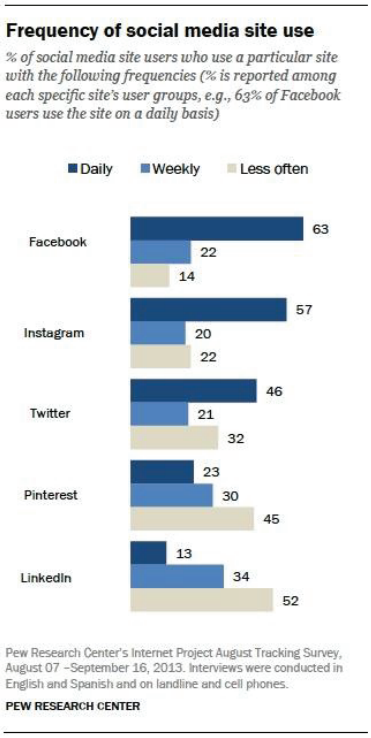


Figure 5 Social media sites usage evolution between 2012-2013, Facebook stays at the head of used sites (63%) of users log in to it daily![15]

- **Pinterest:**

Pinterest was founded in March 2010 and has an Alexa rank of 27 today (within only 4 years)! The concept of Pinterest is to offer to its users a visual discovery tool, combined with social properties. Pinterest provides users with the ability to collect ideas related to their interests. They can create and share collections called "boards", of visual bookmarks (called "Pins"). They can use these boards to meet any goal like collecting interesting recipes, or even for projects like planning for a trip.

From a social aspect, the site permits users to invite other users and share pins and interests with them.

3.2. Domain Based Social Networks DBSNs:

- **Epinions**:

Epinions was established in 1999 and was bought by Shopping.com in 2003 which sold it to eBay in 2005. Epinions is a consumer review site, where users can read, write, and review different items so that they can make a better decision based on others opinions as well as support other users. Users also can rate reviews which will lead to building a community by trusting others. However in March 2014, all community features have been disabled. As epinions has users, relations and ratings, so it has been widely used in most of the social recommender experiments.

[15] http://www.pewinternet.org/2013/12/30/social-media-update-2013/

- **Flickr**:

  Flickr is an image hosting and web services suite that was created by Ludicorp in 2004 and acquired by Yahoo in 2005. In addition to being a popular website for users to share and embed personal photographs, and effectively an online community, the service is widely used by photo researchers and by bloggers to host images that they embed in blogs and social media. In Flickr users can build their communities as well as they can rate items, making it with Epinions the most used social networks in order to evaluate the social recommenders.

As Figure 4 shows, the domain based social networks have much less users that the GPSNs. Many other domain social networks exist, like Flikster for films and LinkedIn for professional profile; coverage of these social networks is out of the scope of this study.

While in Figure 5 we refer to use of social network usage evolution between 2012 and 2013, in (a) there hasn't been a substantial change in those two years: Facebook maintained its position as the most used social network with a slightly increase in the percentage of usage. In (b) in the same figure we find the usage frequency of social network (daily, weekly, and monthly). The study was over 1445 adult users over 18. Interviews were conducted in Spanish and English.

The main conclusion so far is that if there was a way to transform this treasure of information in social networks into recommendation engine this might help users.

So far we have presented the main information of the domain of social networks, and now we will discuss the social recommenders which will link the recommendation domain with social networks.

4. Social recommenders:

Social recommenders combine the two domains of social network and recommendation. Both are interesting topics, used daily by users, and respond to users' needs. This is why researchers say it took some years to combine the two together, trying to enrich recommendation systems by social networks and vice versa, building what is called social recommenders, or community based recommenders[20][21].

The idea is to recommend based on the preferences of the users' friends. Following the epigram "Tell me who your friends are, and I will tell you who you are" [23], [24]. The underlying concept is simple: people prefer to have faith in recommendations from their friends more than recommendations from similar but anonymous individuals [25], because their tastes are similar to, and/or influenced by their trusted friends in social networks.

The main participation of social recommenders in general is to integrate the information extracted about the user, their relations, and friends into a matrix of trust or similarity between friends that feed the collaborative filtering algorithm to find the recommended items.

Usually the social network is denoted by the graph $G(U, F)$, where U is the set of users, and F is the set of friendship links. This graph is translated into a matrix S, where each user u has a set of F_u^+ who are the users trusted by user u, as well as F_u^- who are the users who trust the user u. When relation in the social network are symmetric, meaning that is user u is friend with user v then user v is also a friend of user u, like the case in Facebook, then $F_u^+ = F_u^-$. The contrary is the case of non-symmetric social networks like Twitter, where user u can follow a user v, but the user v doesn't necessarily follow the user u. In this case $F_u^+ \neq F_u^-$. The relation between two users (know, trust, follow…) is represented by a positive value in

the interval$S_{u,v} \in [0,1]$. Where 0 means there is no relation and 1 means the relation is very strong. The value between reflect the level of the relation.

The value of relationship is either obtained explicitly by asking the user to provide the system by how much he trusts other users, or implicitly by observing the actions between the two users.

Usually S contains positive values, but it might have negative values refereeing to the conflicts in interests. An example of the relation matrix is in table 5.

	User 1	User 2	User 3
User 1		0,9	
User 2	0,8		0,7
User 3		0,2	
User 4	0,6	0,4	

Table 5 an example of a relation matrix

Different methods are adopted to achieve this goal: first combining collaborative filtering algorithms with the social information (social matrix), the two main approaches in this domain are Matrix Factorization and neighborhood based [26] discussed next.

4.1. Matrix Factorization based social recommenders

The general idea of Matrix Factorization (MF) is to model the user-item interaction with factors representing latent characteristics of the user and items in the system. After that the model is trained on the available data, to be used to predict ratings of users on new items. The model was widely used in linking social network with recommender systems. In the following we will scan the main works in the domain:

- **SoRec:**

  SoRec was proposed in [27]. SoRec authors aim to address the data sparsity and poor prediction accuracy problem by integrating Social Networks (SN) Recommendation Systems (RS) with In SoRec trust between users in a social network is integrated into a recommender system by factorizing the social trust matrix$S$.In this model the social trust model is slightly twisted on social spectral regularization matrix (SoRec).

$$S^*_{u,v} = S_{u,v}\sqrt{\frac{d_v^-}{d_u^- | d_v^-}} \qquad (12)$$

  Where $d_u^-$is the number of users who u follows/trusts. And d_v^- the number of users who follow/ trust user v. The final user item rating matrix is obtained from the model by the formula:

$$\hat{R} = r_m + QP^T \qquad (13)$$

  In the equation, P and Q contain the rank, and r_m contains the offset. The social information is added to the model by the following equation:

$$S^* = s_m + QZ^T \qquad (14)$$

  Where Z contains the social information in the model, the matrix Q is shared among the two equations. This is why this matrix should contain both: information

about the user-item, and about user-user in order to achieve accurate prediction for both. As the matrix Z is not needed for the prediction process. It can be calculated after the two matrixes P and Q are learned.

In another study [28] both the previous two equations were combined together in one equation in order to optimize RMSE:

$$\sum_{(u,i)obs} \left(R_{u,i} - \hat{R}_{u,i}\right)^2 + \sum_{(u,v)obs} \left(S^*{}_{u,v} - \widehat{S^*}_{u,v}\right)^2 + \gamma(\|P\|_F^2 + \|Q\|_F^2 + \|Z\|_F^2) \quad (15)$$

In the overall concept of SoRec two users Q_u,Q_v become more similar if they were friends. In the original model only positive trust was considered. However the model allows also negative values of $S^*{}_{u,v}$ representing the distrust between two users. Such a function is optimized using the gradient descent method.

- **Social Trust model**

The approach in [29],[30] introduce the social trust model STM. It is a linear combination of basic matrix factorization approach and a social network approach.

Prediction is obtained from comprising the two Matrix$Q$ and P in one formula:

$$\hat{R}_{u,i} = r_m + \alpha Q_u P_i{}^T + (1 - \alpha) \sum_{u \in F_u^-} S_{u,v} Q_u P_i{}^T \quad (16)$$

Where F_u^- is the set of directs friends of user u. So the prediction as the equation shows contain 3 elements, the first two are the same in traditional collaborative filtering methods: global offset r_m and prediction based on user u and item‘s latent features. The last term is the weighted sum of the predicted ratings for item i from all the friends of user u. This is how STM integrates the social influence in the prediction process. α represents the level of the social influence in the prediction process, $\alpha \in [0,1]$. The social influence is ignored with α equal to 1, and it has the highest weight when is equal to 0.

Evaluation was done by optimizing the RMSE by training on the equation:

$$\sum_{(u,i)obs} \left(R_{u,i} - \hat{R}_{u,i}\right)^2 + \gamma\,(\|P\|_F^2 + \|Q\|_F^2) \quad (17)$$

- **Social MF Model**

Social matrix factorization model, proposed in [31], was found to outperform both SoRec and STE (considering the RMSE metric). This model addresses the transitivity relation of trust in the social network. As the dependence of the user’s feature vector on his neighbors feature vectors can broadcast in the social network, making the user’s feature vector depending on possibly all users in the network. Each row in the user’s matrix $S$is normalized to 1, so for each user we have the following equation:

$$\sum_{u} S^*{}_{u,v} = 1 \quad (18)$$

For each user. The prediction in this model is obtained as follows:

$$\hat{R} = r_m + QP^T \qquad (19)$$

While the equation to minimize RMSE is the following:

$$\sum_{(u,i)obs} \left(R_{u,i} - \hat{R}_{u,i}\right)^2 + \beta \sum_{all\ u} \left\| Q_u - \sum_{v \in F_u^-} S^*{}_{u,v} Q_u \right\|^2 + \gamma \left(\|P\|_F^2 + \|Q\|_F^2\right) \qquad (20)$$

In this equation the second term forces the user's latent factor Q_u to be similar to the weighted average of his friends' profilesQ_u while β controls the tradeoff between the rating, and the social features. If $\beta = 0$, then the social aspect is ignored, while if it is equal to 1, this means the social aspect is maximized.

A variation of social FM was proposed in [28] in which the training function was modified so that the ranking (top-k hit ratio) was learned instead of RMSE metric.

- **Circle based recommender:**

Most of existing works in the social recommenders take the social network as a whole in order to produce a recommendation while some works consider another aspect of using recommender systems. in [32] Yang proposed a circle based recommender. The assumption is that humans are multi-faceted, online, or offline, that is the case. To say it differently, in the real life, we refer to different groups of friends in the different domains; for example the technology domain (PC, or camera to buy) towards the clothes domain. Assuming that we know we're able to know the users' trust circles over the different domains, then we can use the information in only this circle in order to recommend to the user in that domain. Nonetheless this definition of circle is not the same with Google +, or Facebook, where the circle might contain friends over different categories.

Yang propose a set of algorithms to infer category-specific circles of friends, then to infer the trust values of each link based on the users ratings. They construct circles of friends from their rating behavior. He follows the assumption that user trusts a friend over a specific category but not in everything, and adding every friend's information result in a noise in finding predictions.

To find an inferred circle of user u: in each category, a user v is in the inferred circle of user u, if and only if the two conditions hold:

- $\mathbf{S_{u,v}} > \mathbf{0}$in the (original) social network: **u** and **v** are friends.
- $\mathbf{n}(\mathbf{M_u^{(c)}}) > \mathbf{0}$and $\mathbf{n}(\mathbf{M_v^{(c)}}) > \mathbf{0}$ in the rating date.

Which implies that both of the users u and v should have rated items related to the same category in order to be in the same circle.

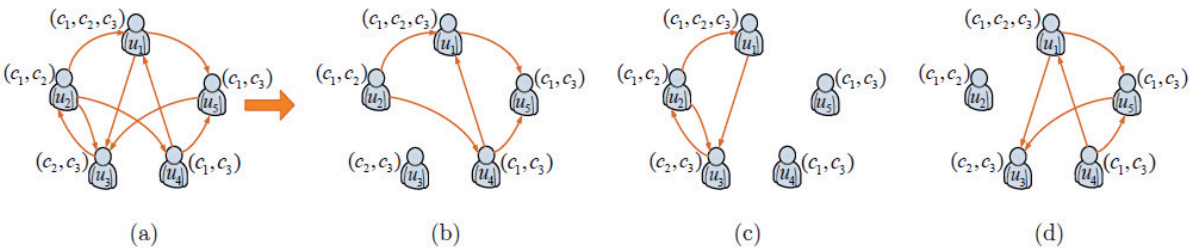


Figure 6 an illustration of circles, each user is labeled with category in which he has rating a) Is the original social network, and b), c), d) are the inferred circles according to the categories c_1, c_2, c_3

Recommendation then will be based on the chosen circle and the friends in this circle. An example of how circle based model work is in figure 6 (from the same reference). Although their solution is new and interesting it has been evaluated in epinions and not on other types of social networks.

In this context we refer to the works of Xu [33]. Xu did something similar to extracting circles, but in his work he does not consider the social aspect, instead he proposed to enhance the clustering on the collaborative filtering matrix. In his method he assigns the user to different clusters, and then the recommendation will be based on the appropriate cluster. This solves the sparsity problem as dividing the matrix like that result in having different matrix which is not smaller and they are not sparse like the main ones. Their method can be considered as a preprocessing of the data, whenever this is done, any recommendation method can be applied over the resulting matrix. At the same time this enhances the recommendation accuracy. Their test was on data set from Last.fm; they compared their model over a variety of methods.

The work	**Domain**	**Evaluation Met**	**Evaluated over**	**Compare to**
SoRec[27]	factor analysis approach (Probabilistic MF)	MAE	Epinions	MMMF (MaximumMargin MF), PMF(Probabilistic MF), CPMF (Constrained Probabilistic MF)
[28] Trust CF-ULF	Latent factor + CF	RMSE, recall	Epinions, Flixster	SocialMF model, SoRec model, STE model
STM[29],[30]	Probabilistic matrix factorization frame-Work	MAE, RMSE	Epinions, doubon,	PMF[34], Trust, SoRec.
Social FM (outperform SoRec, STE)[31]	Similar to STE	RMSE	Epinions, Flixster	BaseMF, STE, CF (memory based)
Adapve social similarities [35]	Matrix factorization with social constraint regualizer	RMSE	Epinions,	PMF, SR, SRPCC[30], ASS
Cir based[32]	Products (MF)	RMSE, MAE	Epinions	BaseMF, SocialMF, circle 1, circle2
FIP[36]	Probabilistic factor-based random walk model MF	AP (average precision), AR(average recall), nDCG	Yahoo! Pulse	item oriented neighborhood (SIM), regression based latent factor (RLFM), neighborhood based latent factor (NLFM),
[37]Matchbox	social spectral regularization method	User click on links, user satisfaction	Facebook	KNN, SVM,

Table 6 A comparison between main social recommender systems using matrix factorization technique

Many other models integrate social networks with recommendation systems, in this sections we scanned the most effective ones –table 6-, and we refer to the survey [26] for more details about the other solutions.

- **Remarks:**

As a general remark in most of these works, the accuracy is becoming better when integrating the social aspect in recommenders. Another remark, we see clearly that Epinions

has been widely used in order to prove social recommendation approaches. Its data is easily accessible and it contains both, the recommendation information (user, item, ratings) as well as the social information. Only Matchbox has been tested on Facebook, but as Matchbox recommend links provided by Facebook users to each other's, it can be considered a closed recommender box. Fib also was applied on Yahoo! Plus and showed high accuracy values.

The most common evaluation metric is the MAE, and RMSE, over the RMSE metric Social MF was found to outperform both SoRec and STM. Fib was tested using the nDCG to evaluate the top N recommended items, besides the average precision and recall. Circles based recommender can be considered in some ways a preprocessing of recommendation in finding the circles in which to recommend.

4.2. Neighborhood Based social recommendation

The neighborhood based social recommendation systems use the available ratings directly into the prediction/recommendation process. The first kind is the social network traversal (STN) which traverse the source-user's neighborhood in the social network and query the rating of a targeted item [26] and the second of which is the nearest neighborhood based approach.

4.2.1. Social network Traversal Based Approaches

The basic concept in this type of recommenders is to query the ratings of user's direct and indirect friends towards an item in order to generate a prediction for a user on an item. Proposed models can be classified into:

- **Trust Weighted Prediction:**

  Some empirical studies have proved finding a correlation between trust and high similarity levels between users as in [38][39]. Trust has been proved to enhance the accuracy in recommendation systems in a many systems. Table 7 scans works that integrate trust with recommendation systems. The table contains some works that use the trust weighted prediction methods in different ways.

  In table 7, we notice that Epinions was used on large scale to evaluate the social trust recommenders, in two works movie Lens was used, TidalTrust used FilmTrust, and finally two works were evaluated by simulations as they use agents to work. The last model was tested on touristic information.

  Evaluating some works was performed by using MAE and RMSE, some other works created new metrics, some evaluated based on the coverage (the amount of items the system can recommend of the whole items).Some interesting finding like in MoloTrust is that adding trust can increase the coverage of Recommender Systems while preserving the quality of predictions. The greatest improvements are achieved for new users, who provided few ratings.

  Many of the proposed solutions were not compared with other ones; some of them were defined to be combined with traditional recommenders (the [40] MoloTrust). [41] Combined Ontology with agents in order to recommend items. However evaluations in the work are not mature enough.

  In TidalTrust[21] as an example of trust based recommenders, the trust value of two users u and v how are not directly connected is calculated in the following way: Tidal trust assigns the trust value of u's direct neighbors to v, the weight is the direct trust value from user u to his direct neighbor.

The work	Domain	Evaluation Met	Evaluated over	Compare to
TidalTrust[21]	Simple neighborhood trust recommendations	∂r, ∂a, ∂cf difference rating&(predicted rating, average rating, CF rating	FilmTrust	ACF : user-to-user nearest neighbor prediction algorithm based on Pearson Correlation
[42][43]TrustawareRecommender Systems	Weighted prediction	No metric	Epinions	No comparison
[44]Trustaware rec	Weighted prediction	MAE, coverage	Epinions	No comparison
[40]MoloTrust	Trust by backward exploration limit (max depth).	Trust metrics, MAE, coverage	Epinions, ebay	Combine with CF
[45]Trust Based Clustering	Add trust Clustering to memory-based collaborative filtering	MAE, RMSE	Movielens	ACF: Pearson similarity and uses Resnick's prediction Formula
[46] Trust based recommendation	Trust on profile, trust on items.	Prediction error	MovieLens	Combine with CF
[47] Agent trust recommender	agents use: trust relationships to filter info& social network to reach far knowledge.	-	Simulation	No comparison
[48] distributed recommender	distributed trust based recommender on SN	Cumulative Distribution Function (CDF against # queries	Simulation	No comparison
[41] trust in the semantic web	Integrate ontology + agents + trust	-	Tourism recommender system	Compared to manual recommendation provided by peers

Table 7 A comparison between main social recommender systems based on trust

- **Bayesian Inference Based Prediction**

  In a most recent work [49] (2013) authors use conditional probability distributions to capture the similarity between friends in social networks. Probability distributions carry richer information than trust values and allow one to employ Bayesian networks to conduct multiple-hop recommendation in online social networks. The experiments show that the accuracy of Bayesian inference based recommendation is better or comparable to that of centralized CF based approaches and trust-based approaches, and can flexibly trade off the amount of recommendations against the recommendation accuracy -Table 8-.

  Evaluations were held on Epinions and MovieLens; the approach was evaluated using the MAE metric, and well as discussing the coverage of the recommended items. Among others, the approach discuss how using trust it can overcome the cold start and sparsity problem.

The work	Domain	Evaluation Met	Evaluated	Compare to
[49]	Bayesian inference based recommendation	MAE, coverage	Epinions, MovieLens	KNN, SVD

Table 8 a social recommender systems using Bayesian networks

- **Random walk based Approaches.**

  Some other social RS algorithms employ random walks in online social networks in order to compute recommendation ratings. Authors of Trust Walker propose a random walk model in online social network. The model queries the ratings of user's friends (direct and indirect friends) for the target item, as well as the similar items. In order to find similarity, TrustWalker combine a trust based approach with an item-item similarity based approach. The item similarity can be calculated either by considering the content of items, like in the content based recommenders, or by considering the users' ratings over items like in the item-item based recommenders.

  Trust Walker requires two components in order to work: the random walk as well as the probabilistic item rating selection on each visited node.

  In the random walk processes the user's direct and indirect friends that are visited in the trust network. Whenever a user views and rates a target item, the rating is logged and if the user has not rated the target item, but has rated an item similar to the target item, the rating is logged with certain probability. The probability of using a rating of a similar item in place of a rating for the target item increases as the length of random walk increases. This probabilistic item rating selection aims to avoid going too deep in the network when no user in a close neighborhood has rated the target item. They employ the Pearson Correlation Coefficient of ratings expressed for two items to calculate the similarity value between them.

The work	Domain	Evaluation Met	Evaluated	Compare to
Trust Walker [50]	random walk	RMSE, Precision, Fmeasure, Coverage(%)	Epinions,	MoleTrust, TidalTrust, CF-User, CF-Item
[51] trust walker	random walk	Recall	Epinions	CF-User, CF-Item
[52]Crime Walker	recommendation in criminal Acts, memory based appoach	Recall	5 years of crime data	-

Table 9 A comparison between main social recommender systems using simulations

The same authors extended their model to recommend top-k items for a source user. Starting from user u, a random walk is performed in the trust network and each random walk stops at a certain user. Then the items rated highly by that user will be considered as the recommended items, ordered according to the ratings expressed by that user. Several random walks are performed to gather more information and compute a more confident recommendation rating. The estimated rating of each item is the average of ratings for that item over all sampled raters. In the end, items with the highest estimated ratings are chosen as top-k recommended items –Table 9-.

Once again Epinions was dominant in the evaluation, while for metrics, recall was used in 2 of the three discussed models.

4.2.2. Nearest Neighbor (NN) Method.

The basic idea is to identify what is called the neighbors of the source user. Then a prediction on a specific item or a recommendation list can be obtained. It is a combination between traditional collaborative filtering techniques with social neighborhood.

As mentioned earlier in [51], one example in which Trust-CF is proposed, in which a Breadth First Search (BFS) approach is applied in order to find the set of trusted neighbors. Then based on this neighbor network a collaborative filtering algorithm is proposed.

In [28] (previously discussed) also another example in which a Trust-CF-ULF method is proposed in order to use the social network information to find the top k recommendations by combining the latent factors with social networks in order to achieve this goal. In the same model it has been found that the technical approach which is used in order to combine the ratings with social network information works best for minimizing.

4.3. Social recommenders, Discussion

A variety of methods has been proposed in order to build social recommenders. As it is the case in traditional collaborative filtering, model based social recommenders perform well in both the rating prediction process, and in building the recommended list also, while neighborhood approaches are easy to implement.

Most of the work was mainly based on enhancing the similarity measures by adding trust or other social information extracted from social information, then performing prediction and recommendation.

Most of the solutions have been evaluated and tested on Epinions and Flixster data sets, using in most cases the MAE and RMSE metrics. Some works however use other social networks or metrics to be evaluated.

In certain works like [44][21], authors report that overall, social-network based recommendations are no more accurate than those derived from traditional CF approaches, except in special cases, such as when user ratings of a specific item is highly varied (i.e. controversial items) or for cold-start situations, i.e., where the users did not provide enough ratings to compute similarity to other users. However in most of the previous approaches, social recommenders were shown to yield better recommendations than profile similarity data.

As a social network continues to mature, the amount of information that it offers about its users increases. This is why there is still a need for a recommendation that can mine different links of newly available user's information in social networks.

Although social networks and recommendation systems are dynamic domains, most of the discussed solutions are trained and tested offline. Next steps might be to test and evaluate these systems in real time user experience, as well as to consider how to integrate the user continuous users' information in recommender systems.

Privacy is also a hot topic related to social recommender systems posing questions about if its users really agree to use their private data for recommendation.

5. Context-Awareness Review

The term "context-aware" first appeared in [53], where context was referred to as where you are, who you are with, and what resources are nearby. In the same paper, the authors observed that context is more than a user's location since other things of interest change as well, like a user's social situation.

In this sense, context-aware applications dynamically change or adapt their behavior based on the context of the application and the user. They either automatically execute a service, or present the information and services to a user, or tag the context to the information for later retrieval [54]. Context awareness is used for different domains, like social computing, intelligent ambient, user modeling, knowledge representation, and of course the focus of our work in the contextual recommendation.

In [55], context is defined as any information that can be used to characterize the situation of entities (place, people, and things), including the user and application and the interaction between them. In his work, Day considers the user context as the people nearby and social situations, in addition to other elements like the user profile and location.

Additional works classify context in different dimensions; in [56] and [57], there are external and internal ones, while in [58] they are presented in physical and logical contexts.

The external (physical) dimension signifies a context measured by hardware sensors, such as location, light, sound, movement, touch, and temperature or air pressure. The internal (logical) dimension relates to the user, either user-specified or captured, like the user's goals, tasks, work context, people nearby, social context, business processes, and the user's emotional state.

In [59] individual and group context are defined, as an internal and external vision. [60] defines the social context as a complement of the individual context. In this case, each node has two contexts: individual (its own view of itself), like its profile and preference, and a social one (being aware to be a part of the system), like people nearby and current social situation descriptions. This kind of context is in use by many systems that work through automatic recommendation and situation-based adaptation. Examples include using co-localization patterns to find common interests and recommend possible friends, while another infers the current situation, for instance, to switch off a cell phone ring tone during a business dinner. [61]

In [62], social context is referred to the concept of surrounding where interaction begins between a group. It captures the context and history around social interactions, which include social positions, social roles, customs, standards, values, kinship, ties fashions, and culture.

We stand with the position of [63], which states that definitions of context vary between those defining abstract concepts and others presenting context in a technical manner.

Returning to the different definitions of context, it is common to classify the context under two principle categories: the context acquired by hardware sensors concerning the surroundings and the context concerning the entities, or individuals themselves. The second latter depends on the former but it is more difficult to acquire and studied under the definition of situations. Many studies depend on it to be provided, while others work to predict it, either by supervised or non-supervised methods.

A number of surveys [55], [64], [65] show that location is the most-used type of context for domains as it is easily measured. Looking to [64] in 2000, where social context was defined as part of a user's context, the authors debate the importance of social context but see

obstacles in obtaining it. They suggest techniques like camera-based and image processing approaches, as well as utilizing agendas. In spite of this, they recognize the problem that users are not always willing to note everything. Thus, the third proposed solution is to recognize complex context from low-level sensors or what we call situation recognition. Here, much work has been done on the physical context but not much in the area of logic.

The difficulty of the context-aware application lies in finding the best balance and strategy between the different elements: context sensing, context modeling, and context processing. The goal application also has an important role.

5.1. Human Information as High Level Context (Situations)

High-level context includes human interactions and behavior information, meaning social context such as a user's profile, nearby people, and current social situations. This serves as a way to overcome problems of low-level context information acquired from the environment [66]. The information of low-level sensors is abstracted by a model layer that transforms the low-level context input to generate or trigger system actions. This external semantic interpretation of low-level context in context aware application is named Situation [67] [68], [69].

Situations add meaning into the application because they are more stable and easier to define and maintain than basic contextual cues. Change in situations cause adaptations in context-aware applications [66].

The situation is the physical, social and cultural space (context) in which the activity is carried out (see Chapter 2 in [70]). In the case of pseudo-simultaneous activities, the situation is characterized also by the other activities that are being carried out at the same time. We can consider the situation as being defined by what concepts are related to the user at the present time and how they are related. [71] To resume, the situation is a temporal state within context.

5.2. Defining Situations

Defining situations is a challenging task based on the extraction of human knowledge and interpretation. The techniques of situation recognitions have shifted from manually and logically based ones towards learning-based ones; the second focuses on the classification of basic human activities without considering a richer contextual description [72]. As an example in [73], the situation is characterized using the concepts of role and relation. Roles involve only one entity, describing its activity. An entity is observed to play a role. Relations are defined as predicate functions on several entities, describing the relationship or interaction between entities playing roles. [74] investigates in detail this shift from logical to learning techniques.

5.3. Relationships Between Situations:

Some approaches model the relationship between different situations. Often, situations claim that at least one situation must be active at a time. An example in [75] is where the situations of a lecture were defined for its auto recording using a Henri bet model. It may be that this concept, when examined from a social science perspective [76], is related to the recognition of "turns" in Activities which according to [76] gives valuable clues for an interpretation of social context. But according to them, the sensors' data can only be interpreted for this purpose in light of a well-defined domain.

6. Context Awareness and Users Modeling:

In his book, Brusilovsky [77] pointed out that user modeling is either based on feature modeling or on stereotype modeling. In feature modeling, the most popular and useful features when viewing a user as an individual are: the user's knowledge, interests, goals, background, and individual traits; he added the context of the user's work to these five features.

Indeed, the context modeling and user modeling are highly connected. Many user models include context features and vice versa. Context and user modeling inspire similar techniques; for user models the techniques to model the five features vary between:

User's knowledge: scalar model, metadoc, overlay model (subset of expert model), bug model, and genetic model. The last two are more theoretically based, and difficult to apply. The first three remind us of the key-value models of context modeling [65].

Interests: weighted vector of keywords, overlay model. Also reminds us of the key-valus model.

Goals: what does the user actually want to achieve? It is the most changeable feature of the user. There is the predefined goal catalog approach, where a user specifies the current goal and systems should adapt to new goals. Some models depend on the probability of the goal and data mining. This branch shows much relation with situation analysis, especially when trying to predict the user's goal from his/her situation.

Background: they are impossible to deduce so they are usually provided explicitly.

Individual traits: they are also difficult to deduce so they are provided by the user.

The last two topics are provided by users or collected by the system from different sources when dealing with context. Ontology modeling, of course, has a deep participation in both context and user modeling. In context awareness, the model of user context varies between what amounts of the feature they cover. As a result of the relation between the two branches, integrated frameworks were developed for modeling them both. Examples of such frameworks are in [78] [79]. In the first application, an ontology model is proposed. In [80] we find a study on the use of ontology in user modeling.

As a border between user and context modeling, Brusilovsky remarks that some information represented in the context models can hardly be considered information about a user in a pure sense. User modeling focuses mostly on long-term properties of the user that are distilled from observations, while context models attempt to represent the current features of the user and the environment, or what he names the affective state. As a solution to predict this state, he has presented the work of [81]. In this work, observing users' web log data can be used to detect their motivation. Figure 7 (from the same reference) shows the two visions of context dimensions, the user-centric and the device-centric views.

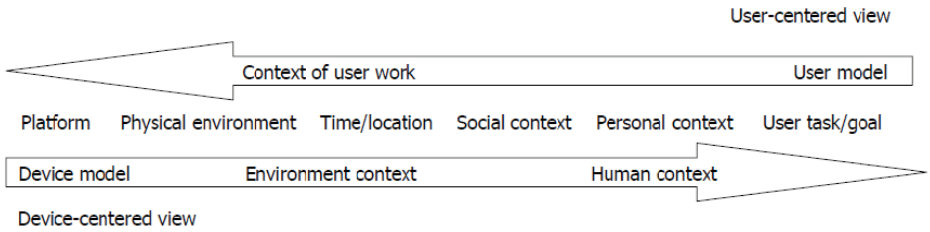


Figure 7 User- centric and device- centric view of context dimensions.

The assumption of the effective state was affirmed in [82] (published the same year of Bruvelnsky's book 07) where Brezillon noted that the discriminating factor between external knowledge and contextual knowledge (the user model and the context) is the focus. In the case of user application interaction, the focus is the user's attention, so that contextual information is that information which is related to the current focus, and everything else is external knowledge. This focus changes either according to external events or as a result of internal decisions. The active situation is another synonym for the same concept.

Dealing with this active situation marks the difference between the models of user modeling and context awareness. But it is still difficult to classify in detail the approaches to broader context and user modeling, since the most frequently used contexts are platform and location [55], [64], [65].

Stereotype modeling is based on assigning users to stereotypes (or roles). The adaptation is to change the assigned stereotype to the user. In this vision of user modeling we also see the relation with context awareness; for example, in some works the role of a user is predicted, while in others the role is defined and fixed but the resulting actions are modeled [83]. Linking the feature type and stereotype is the power for user modeling [77]; this can be applied for user context modeling as well.

7. Context-aware Recommenders

Context awareness enhanced the prediction accuracy of different domains containing recommendation systems [84]. All applications using context follow the assumption that context changes clearly affect the need of certain items or services. An example of this would be watching a film with a brother during the week versus watching a film with a girlfriend during the weekend.

Three different strategies can be adopted when integrating context in a recommender [85]: reduction-based (pre-filtering), contextual and post filtering, and context modeling. In the reduction-based methods, only the information that has been used in similar contexts is used to recommend in the current scenario, like the ratings of items evaluated in similar situations. In contextual and post filtering, the recommendation process itself does not change or take in consideration the contextual information. Instead, the output of the recommendation is filtered to include only the results related to the actual context of the user. Finally, in contextual modeling, which is the most complicated strategy, the contextual data is explicitly integrated in the prediction model. The problem with contextual modeling is that it is not easy to obtain, especially in the social context and contexts related to the users' internal feelings and desires [20].

Some examples of post filtering solutions are [86], [87], [88], which include contextual information into existing recommendation frameworks like the Matrix Factorization. In [89], the authors propose a multidimensional recommendation model based on multiple dimensions, i.e., user/item dimensions as well as various contextual information. With this setup, many statistical tests are used in order to determine the impactful context when providing recommendation [90].

Other families of contextual works focus on building models that integrate contextual information directly (context modeling models) into traditional user-item-rating relations. An example is found in [91], where a multiverse recommendation model has been proposed in order to model the data as a user-item-context N-dimensional tensor, then a Tucker decomposition is applied to factorize the tensor [92]. The shortcoming of this solution is that it is applicable only in cases of categorical contextual information. In [87], an enhanced

model was proposed to deal with all types of context but the model suffers from scalability issues if the user-item matrix is too large.

In [86], the authors propose a contextual collaborative filtering algorithm (called RPMF) to support context-aware recommendation. The assumption behind this model is that contextual information is encoded in or reflected by the user-specific and item-specific latent factors. Based on this, tree-based random partition is applied to split the user-item-rating matrix by grouping users and items with similar contexts, and then matrix factorization is applied to the generated sub-matrices. Similar to this, Soco [84] uses a tree decomposition in order to integrate context with recommender systems, but Soco goes farther in linking social information with contextual information in order to get higher accuracy. Both solutions integrate Matrix Factorization on the tree, and while RPMF applies it on each node of the tree Soco applies the matrix factorization to the whole tree.

As discussed in the previous chapter, social information clearly affects recommendation, and we also discussed that social information can be considered as contextual information that affects the whole recommendation process.

Soco [84] offers one solution in an attempt to link the two aspects (social and contextual) together in the recommendation system that it offers. Before Soco other solutions had been proposed, which unfortunately did not efficiently combined different types of contextual information (e.g., contexts with discrete values versus contexts with continuous values [91]) or suffered from high computational complexity (e.g., the matrix factorization model is impractical for extremely large data sets, or multiple matrix factorization operations are needed [86]).

Other parts considered only social contexts like group-aware friendship models [90]. [32] introduced both in the previous chapter. In other instances, only specific contextual information such as time or user mood when an item is rated is considered for social recommendation [93], [94].

iAmélie, proposed in the following chapters belongs to the family of recommenders that link contextual social information in order to enhance recommendation.

8. Conclusion

Social media has opened the door to a variety of questions and issues that were too difficult to be considered before. As social media opened the door to everyone, wherever they would like to do any action he wants on the web.

Social networks, as a branch of social media, continue to evolve in an interesting way; people are constantly joining as well as uploading valuable information regularly.

The common sense indicated that users classify their relations according to different social networks. In other words, users do not have the same groups of friends over different social networks, as in LinkedIn they can have a different group that those in Facebook, Twitter, or Flickr. However, there is no study that investigates this issue.

Social recommendation is based on linking recommender systems with social networks in order to enhance the quality of recommendation and reach more user satisfactions, a variety of solutions has been proposed in the work, but much of them are based on Epinions and Flixster to be tested, and are evaluated with accuracy metrics like MAE, and RMSE.

To what extend does the added value in accuracy indicate a good recommendation? Does it really affect the user if the MAE has been evolved from a model to another with 0,03 for

example? Especially that, when recommending it has been proved that the problem when considering accuracy is not in how good an item is to recommend it, but on the contraire, how bad an item is not to recommend it [95][96]. And however a bad item is bad, either it had 1, or 2 rating that will not change the behavior towards it.

Besides most of the proposed models were evaluated on Epinions, Flixster or other domain based social networks, these results are similar to the general evaluation discussion in the last chapter where evaluation in general in recommendation systems has always been performed offline on available data sets.

What would be an interesting idea is to investigate the data in GPSN, and try to use them to build a recommender which has not been done in the past. The main difficulty in working with GPSN is in translating the raw data extracted from these social networks into useful information for recommendation, as these social networks are not recommendation oriented, and they do not contain explicitly the triple (ratings, user, items) like in Epinions and Flixster. Another problem is how to compare the performance with state of the art works. As for now there is no works addressing this problem. And that would be the focus of our first contribution iSoNTRE.

Logically it is much easier to understand the effect of a neighbor on a decision if the user clarifies that the trusts such neighbors in a social network such as Epinions. But trying to infer this from a GPSN is much more difficult because in such sites relations are open as a user u might be a friend with user v, but they do not trust at all of the commercial tastes. To say it differently, a user might have totally different relations in a commercial site than in a GPSN site like Facebook. That would be the focus of our second contribution iAmélie. Now we move to introduce the context based recommenders that will have a role in our iAmélie system.

Context awareness from another side adds a layer on applications that makes the life of users much easier. People tend to have interests, goals, and desires, but these constitute relative information, relative to their context, internal mood, surroundings and many other elements; as such, a recommender that considers these facets of context for its recommendations responds better to users' needs and enhances the overall accuracy of its own system.

From the evidence, we see that these two aspects of contextual and social, when purposely put together, can result in a much better recommender. Contextual solutions evolved due to geographical and temporal information; However, one main drawback of these solutions is that they take into consideration mainly one aspect of the social context when performing recommendation, which is the friends or groups of friends based on a common interest as in Soco [84] and [32] Circle based. Though this fact might not always be correct, as in some cases users might like to be recommended based on their friends' information, whereas in other cases they would prefer recommendations based on their own interests or interests of experts in a domain.

In introducing iAmélie we will hone in on this problem and discuss in detail the assumptions of different recommenders before using context and social information to respond to users' needs.

9. References:

[1] J. H. Kietzmann, B. S. Silvestre, I. P. McCarthy, et L. F. Pitt, « Unpacking the social media phenomenon: towards a research agenda », J. Public Aff., vol. 12, no 2, p. 109–119, 2012.

[2] N. B. Ellison et others, « Social network sites: Definition, history, and scholarship », J. Comput.-Mediat. Commun., vol. 13, no 1, p. 210–230, 2007.

[3] K. Plangger, « The power of popularity: how the size of a virtual community adds to firm value », J. Public Aff., vol. 12, no 2, p. 145–153, 2012.

[4] R. Chakrabarti et P. Berthon, « Gift giving and social emotions: experience as content », J. Public Aff., vol. 12, no 2, p. 154–161, 2012.

[5] P. Longart, « What drives word-of-mouth in restaurants? », Int. J. Contemp. Hosp. Manag., vol. 22, no 1, p. 121–128, 2010.

[6] M. Khammash et G. H. Griffiths, « 'Arrivederci CIAO. com, Buongiorno Bing. com'—Electronic word-of-mouth (eWOM), antecedences and consequences », Int. J. Inf. Manag., vol. 31, no 1, p. 82–87, 2011.

[7] J. H. Kietzmann et I. Angell, « Generation–C: creative consumers in a world of intellectual property rights », Int. J. Technol. Mark., vol. 9, no 1, p. 86–98, 2014.

[8] A. M. Kaplan et M. Haenlein, « Users of the world, unite! The challenges and opportunities of Social Media », Bus. Horiz., vol. 53, no 1, p. 59–68, 2010.

[9] J. Arndt et A. R. Foundation, Word of mouth advertising: a review of the literature. Advertising Research Foundation, 1967.

[10] J. F. Engel, R. D. Blackwell, et R. J. Kegerreis, « How information is used to adopt an innovation », J. Advert. Res., vol. 9, no 4, p. 3–8, 1969.

[11] E. M. Steffes et L. E. Burgee, « Social ties and online word of mouth », Internet Res., vol. 19, no 1, p. 42–59, 2009.

[12] F. A. Buttle, « Word of mouth: understanding and managing referral marketing », J. Strateg. Mark., vol. 6, no 3, p. 241–254, 1998.

[13] M. Huang, F. Cai, A. S. Tsang, et N. Zhou, « Making your online voice loud: the critical role of WOM information », Eur. J. Mark., vol. 45, no 7/8, p. 1277–1297, 2011.

[14] D. L. Williams, V. L. Crittenden, T. Keo, et P. McCarty, « The use of social media: an exploratory study of usage among digital natives », J. Public Aff., vol. 12, no 2, p. 127–136, 2012.

[15] M. Bulearca et S. Bulearca, « Twitter: a viable marketing tool for SMEs? », Glob. Bus. Manag. Res. Int. J., vol. 2, no 4, p. 296–309, 2010.

[16] J. H. Kietzmann, K. Hermkens, I. P. McCarthy, et B. S. Silvestre, « Social media? Get serious! Understanding the functional building blocks of social media », Bus. Horiz., vol. 54, no 3, p. 241–251, 2011.

[17] J. Grimmelmann, « Facebook and the social dynamics of privacy », Iowa Law Rev., vol. 95, no 4, 2009.

[18] « Sweden's largest Facebook study: a survey of 1000 Swedish Facebook users - University of Gothenburg, Sweden ». [En ligne]. Disponible sur: http://www.gri.handels.gu.se/english/latest-news/news/d/sweden-s-largest-facebook-study--a-survey-of-1000-swedish-facebook-users.cid1073014. [Consulté le: 07-nov-2012].

[19] J. Grimmelmann, « Privacy as product safety », Widener LJ, vol. 19, p. 793, 2009.

[20] F. Ricci, L. Rokach, et B. Shapira, Introduction to recommender systems handbook. Springer, 2011.

[21] J. Golbeck, Generating predictive movie recommendations from trust in social networks. Springer, 2006.

[22] D. Goldberg, D. Nichols, B. M. Oki, et D. Terry, « Using collaborative filtering to weave an information tapestry », Commun. ACM, vol. 35, no 12, p. 61–70, 1992.

[23] O. Arazy, N. Kumar, et B. Shapira, « Improving Social Recommender Systems. », IT Prof., vol. 11, no 4, p. 38–44, 2009.

[24] D. Ben-Shimon, A. Tsikinovsky, L. Rokach, A. Meisles, G. Shani, et L. Naamani, « Recommender system from personal social networks », in Advances in Intelligent Web Mastering, Springer, 2007, p. 47–55.

[25] R. R. Sinha et K. Swearingen, « Comparing Recommendations Made by Online Systems and Friends. », in DELOS workshop: personalisation and recommender systems in digital libraries, 2001, vol. 106.

[26] X. Yanga, Y. Guob, Y. Liua, et H. Steckc, « A Survey of Collaborative Filtering Based Social Recommender Systems ».

[27] H. Ma, H. Yang, M. R. Lyu, et I. King, « Sorec: social recommendation using probabilistic matrix factorization », in Proceedings of the 17th ACM conference on Information and knowledge management, 2008, p. 931–940.

[28] X. Yang, H. Steck, Y. Guo, et Y. Liu, « On top-k recommendation using social networks », in Proceedings of the sixth ACM conference on Recommender systems, 2012, p. 67–74.

[29] H. Ma, I. King, et M. R. Lyu, « Learning to recommend with social trust ensemble », in Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval, 2009, p. 203–210.

[30] H. Ma, D. Zhou, C. Liu, M. R. Lyu, et I. King, « Recommender systems with social regularization », in Proceedings of the fourth ACM international conference on Web search and data mining, 2011, p. 287–296.

[31] M. Jamali et M. Ester, « A matrix factorization technique with trust propagation for recommendation in social networks », in Proceedings of the fourth ACM conference on Recommender systems, 2010, p. 135–142.

[32] X. Yang, H. Steck, et Y. Liu, « Circle-based recommendation in online social networks », in Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining, 2012, p. 1267–1275.

[33] R. Pan, G. Xu, et P. Dolog, « Improving Recommendations in Tag-based Systems with Tag Neighbor Expansion », in the Proceedings of the 3rd International Conference on Computer Science and its Applications (CSA-11), 2011, p. 12–15.

[34] A. Mnih et R. Salakhutdinov, « Probabilistic matrix factorization », in Advances in neural information processing systems, 2007, p. 1257–1264.
[35] L. Yu, R. Pan, et Z. Li, « Adaptive social similarities for recommender systems », in Proceedings of the fifth ACM conference on Recommender systems, 2011, p. 257–260.
[36] S.-H. Yang, B. Long, A. Smola, N. Sadagopan, Z. Zheng, et H. Zha, « Like like alike: joint friendship and interest propagation in social networks », in Proceedings of the 20th international conference on World wide web, 2011, p. 537–546.
[37] J. Noel, S. Sanner, K.-N. Tran, P. Christen, L. Xie, E. V. Bonilla, E. Abbasnejad, et N. Della Penna, « New objective functions for social collaborative filtering », in Proceedings of the 21st international conference on World Wide Web, 2012, p. 859–868.
[38] C.-N. Ziegler et G. Lausen, « Analyzing correlation between trust and user similarity in online communities », in Trust management, Springer, 2004, p. 251–265.
[39] C.-N. Ziegler et J. Golbeck, « Investigating interactions of trust and interest similarity », Decis. Support Syst., vol. 43, no 2, p. 460–475, 2007.
[40] P. Massa et P. Avesani, « Controversial users demand local trust metrics: An experimental study on epinions. com community », in Proceedings of the National Conference on artificial Intelligence, 2005, vol. 20, p. 121.
[41] P. Bedi, H. Kaur, et S. Marwaha, « Trust Based Recommender System for Semantic Web. », in IJCAI, 2007, vol. 7, p. 2677–2682.
[42] P. Massa et B. Bhattacharjee, « Using trust in recommender systems: an experimental analysis », in Trust Management, Springer, 2004, p. 221–235.
[43] P. Massa et P. Avesani, « Trust-aware recommender systems », in Proceedings of the 2007 ACM conference on Recommender systems, 2007, p. 17–24.
[44] P. Massa et P. Avesani, « Trust-aware collaborative filtering for recommender systems », in On the Move to Meaningful Internet Systems 2004: CoopIS, DOA, and ODBASE, Springer, 2004, p. 492–508.
[45] T. DuBois, J. Golbeck, J. Kleint, et A. Srinivasan, « Improving recommendation accuracy by clustering social networks with trust », Recomm. Syst. Soc. Web, vol. 532, p. 1–8, 2009.
[46] J. O'Donovan et B. Smyth, « Trust in recommender systems », in Proceedings of the 10th international conference on Intelligent user interfaces, 2005, p. 167–174.
[47] F. E. Walter, S. Battiston, et F. Schweitzer, « A model of a trust-based recommendation system on a social network », Auton. Agents Multi-Agent Syst., vol. 16, no 1, p. 57–74, 2008.
[48] K. Sarda, P. Gupta, D. Mukherjee, S. Padhy, et H. Saran, « A distributed trust-based recommendation system on social networks », in Proceedings of the 2nd IEEE Workshop on Hot Topics in Web Systems and Technologies, 2008, p. 1–6.
[49] X. Yang, Y. Guo, et Y. Liu, « Bayesian-Inference-Based Recommendation in Online Social Networks », Parallel Distrib. Syst. IEEE Trans. On, vol. 24, no 4, p. 642–651, 2013.
[50] M. Jamali et M. Ester, « TrustWalker: a random walk model for combining trust-based and item-based recommendation », in Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, 2009, p. 397–406.
[51] M. Jamali et M. Ester, « Using a trust network to improve top-N recommendation », in Proceedings of the third ACM conference on Recommender systems, 2009, p. 181–188.
[52] M. A. Tayebi, M. Jamali, M. Ester, U. Glässer, et R. Frank, « Crimewalker: a recommendation model for suspect investigation », in Proceedings of the fifth ACM conference on Recommender systems, 2011, p. 173–180.
[53] B. Schilit, N. Adams, et R. Want, « Context-aware computing applications », in Mobile Computing Systems and Applications, 1994. WMCSA 1994. First Workshop on, 1994, p. 85–90.
[54] A. K. Dey, « Providing architectural support for building context-aware applications », Georgia Institute of Technology, 2000.
[55] A. K. Dey, G. D. Abowd, et D. Salber, « A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications », Hum.-Comput. Interact., vol. 16, no 2-4, p. 97–166, 2001.
[56] P. Prekop et M. Burnett, « Activities, context and ubiquitous computing », Comput. Commun., vol. 26, no 11, p. 1168–1176, 2003.
[57] R. M. Gustavsen, « Condor–an application framework for mobility-based context-aware applications », in Proceedings of the Workshop on Concepts and Models for Ubiquitous Computing, 2002.
[58] G. Kappel, W. Retschitzegger, E. Kimmerstorfer, B. Pröll, W. Schwinger, et T. H. Hofer, « Towards a generic customisation model for ubiquitous web applications », in 2nd International Workshop on Web Oriented Software Technology, 2002.
[59] P. Brézillon, « Context and Explanation in e-Collaborative Work », Handb. Res. Methods Tech. Stud. Virtual Communities Paradig. Phenom., vol. 1, p. 285, 2010.
[60] P. T. Eugster, B. Garbinato, et A. Holzer, « Middleware Support for Context-Aware Applications », Middlew. Netw. Eccentric Mob. Appl., vol. 1, p. 305, 2009.
[61] P. BELLAVISTA, A. CORRADI, M. FANELLI, et L. FOSCHINI, « A Survey of Context Data Distribution for Mobile Ubiquitous Systems », ACM Comput. Surv., vol. 45, no 1, 2013.
[62] A. Rakotonirainy, S. W. Loke, et P. Obst, « Social awareness concepts to support social computing », in Computational Science and Engineering, 2009. CSE'09. International Conference on, 2009, vol. 4, p. 223–228.
[63] M. Perttunen, J. Riekki, et O. Lassila, « Context representation and reasoning in pervasive computing: a review », Int. J. Multimed. Ubiquitous Eng., vol. 4, no 4, 2009.
[64] G. Chen et D. Kotz, « A survey of context-aware mobile computing research », 2000.

[65] M. Baldauf, S. Dustdar, et F. Rosenberg, « A survey on context-aware systems », Int. J. Ad Hoc Ubiquitous Comput., vol. 2, no 4, p. 263–277, 2007.
[66] C. Bettini, O. Brdiczka, K. Henricksen, J. Indulska, D. Nicklas, A. Ranganathan, et D. Riboni, « A survey of context modelling and reasoning techniques », Pervasive Mob. Comput., vol. 6, no 2, p. 161–180, 2010.
[67] P. Gray et D. Salber, « Modelling and using sensed context information in the design of interactive applications », Eng. Hum.-Comput. Interact., p. 317–335, 2001.
[68] A. K. Dey, « Understanding and using context », Pers. Ubiquitous Comput., vol. 5, no 1, p. 4–7, 2001.
[69] S. Dobson et J. Ye, « Using fibrations for situation identification », in Pervasive 2006 workshop proceedings, 2006, p. 645–651.
[70] G. Riva et G. Riva, Ambient Intelligence: the evolution of technology, communication and cognition towards the future of human-computer interaction, vol. 6. IOS Press, 2005.
[71] A. Olaru, « A context-aware multi-agent system for AmI environments », Technical report, University Politehnica of Bucharest, University Pierre et Marie Curie Paris, 2010. February, 2010.
[72] A. Ranganathan et R. H. Campbell, « An infrastructure for context-awareness based on first order logic », Pers. Ubiquitous Comput., vol. 7, no 6, p. 353–364, 2003.
[73] J. Crowley, J. Coutaz, G. Rey, et P. Reignier, « Perceptual components for context aware computing », UbiComp 2002 Ubiquitous Comput., p. 117–134, 2002.
[74] J. Ye, S. Dobson, et S. McKeever, « Situation identification techniques in pervasive computing: A review », Pervasive Mob. Comput., 2011.
[75] G. Shafer, A mathematical theory of evidence, vol. 1. Princeton university press Princeton, 1976.
[76] J. Barwise et J. Perry, Situations and attitudes. 1983.
[77] P. Brusilovsky, « Adaptive navigation support », Adapt. Web, p. 263–290, 2007.
[78] A. Zimmermann, M. Specht, et A. Lorenz, « Personalization and context management », User Model. User-Adapt. Interact., vol. 15, no 3, p. 275–302, 2005.
[79] D. Heckmann, T. Schwartz, B. Brandherm, M. Schmitz, et M. von Wilamowitz-Moellendorff, « Gumo–the general user model ontology », User Model. 2005, p. 149–149, 2005.
[80] S. Sosnovsky et D. Dicheva, « Ontological technologies for user modelling », Int. J. Metadata Semant. Ontol., vol. 5, no 1, p. 32–71, 2010.
[81] H. Prendinger, J. Mori, et M. Ishizuka, « Recognizing, modeling, and responding to users' affective states », in User Modeling 2005, Springer, 2005, p. 60–69.
[82] J. Brézillon et P. Brézillon, « Context modeling: Context as a dressing of a focus », Model. Using Context, p. 136–149, 2007.
[83] P. Brusilovsky et E. Millán, « User models for adaptive hypermedia and adaptive educational systems », Adapt. Web, p. 3–53, 2007.
[84] X. Liu et K. Aberer, « SoCo: a social network aided context-aware recommender system », in Proceedings of the 22nd international conference on World Wide Web, 2013, p. 781–802.
[85] G. Adomavicius et A. Tuzhilin, « Context-aware recommender systems », in Recommender systems handbook, Springer, 2011, p. 217–253.
[86] E. Zhong, W. Fan, et Q. Yang, « Contextual Collaborative Filtering via Hierarchical Matrix Factorization. », in SDM, 2012, p. 744–755.
[87] S. Rendle, Z. Gantner, C. Freudenthaler, et L. Schmidt-Thieme, « Fast context-aware recommendations with factorization machines », in Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval, 2011, p. 635–644.
[88] L. Baltrunas, B. Ludwig, et F. Ricci, « Matrix factorization techniques for context aware recommendation », in Proceedings of the fifth ACM conference on Recommender systems, 2011, p. 301–304.
[89] G. Adomavicius, R. Sankaranarayanan, S. Sen, et A. Tuzhilin, « Incorporating contextual information in recommender systems using a multidimensional approach », ACM Trans. Inf. Syst. TOIS, vol. 23, no 1, p. 103–145, 2005.
[90] B. Xu, J. Bu, C. Chen, et D. Cai, « An exploration of improving collaborative recommender systems via user-item subgroups », in Proceedings of the 21st international conference on World Wide Web, 2012, p. 21–30.
[91] A. Karatzoglou, X. Amatriain, L. Baltrunas, et N. Oliver, « Multiverse recommendation: n-dimensional tensor factorization for context-aware collaborative filtering », in Proceedings of the fourth ACM conference on Recommender systems, 2010, p. 79–86.
[92] L. R. Tucker, « Some mathematical notes on three-mode factor analysis », Psychometrika, vol. 31, no 3, p. 279–311, 1966.
[93] N. N. Liu, B. Cao, M. Zhao, et Q. Yang, « Adapting neighborhood and matrix factorization models for context aware recommendation », in Proceedings of the Workshop on Context-Aware Movie Recommendation, 2010, p. 7–13.
[94] N. N. Liu, L. He, et M. Zhao, « Social temporal collaborative ranking for context aware movie recommendation », ACM Trans. Intell. Syst. Technol. TIST, vol. 4, no 1, p. 15, 2013.
[95] B. Marlin, R. S. Zemel, S. Roweis, et M. Slaney, « Collaborative filtering and the missing at random assumption », ArXiv Prepr. ArXiv12065267, 2012.
[96] B. M. Marlin et R. S. Zemel, « Collaborative prediction and ranking with non-random missing data », in Proceedings of the third ACM conference on Recommender systems, 2009, p. 5–12.

CHAPTER 4: ISSUES IN GPSNS AND IN RECOMMENDATION SYSTEMS ASSUMPTIONS

1. Introduction:

In this chapter, we will introduce to the 2 models that we will present later: iSoNTRE in chapters 5 and iAmélie in chapter 6. We will begin by discussing GPSNs towards DBSNS, and then we will move to consider the different assumptions in recommendation systems, and how the real need is to create a system that responds to all of them. The main goal of this chapter is to understand how the two models collaborate in order to offer better recommendation to users. The first model belongs to social recommenders' techniques; while the second model iAmélie can be defined as a contextual recommender. iAmélie can be considered as an extension of iSoNTRE which is a social one, knowing that both systems belong to the hybrid family or recommenders.

1.1. GPSNs towards DBSNs:

The social recommenders discussed in Chapter 3 are mainly applied over Domain Based Social Networks (DBSN) like Flixster and Epinions, which already contain users, items, ratings as well as relations. The methods operating on DBSN's rely on the specificity of the network for the recommendations to be meaningful. They also do not have to deal with ambiguity resulting from the network having multiple purposes or general purposes. Thus these social recommenders are usually not applied to General Purpose Social Networks (GPSN) like Facebook and Twitter, neither are they easily applied to them, because GPSNs contain raw information about users, which need to be extracted then processed and modeled before being exploitable in social recommenders.

As discussed in chapter 3 GPSN are much common than DPSN: with 1,310 million of monthly active Facebook users , and 284 million monthly active users on Twitter, compared to 50 million on Flixster and 1 million on Epinions. People are here in GPSN! They spend nearly 75 minutes per day on Facebook (81 minutes for women and 64 minutes for men) and 67% of users use these networks as a hobby! [1]. According to the same reference Facebook turned into a habit-forming activity and is a part of people's daily routine. In Twitter it takes 2 days to produce 1 billion tweets!

Based on these facts, we believe using GPSN information for recommendation could have several advantages, like:

- Overcoming the cold start problem for users (chapter 2) which is a perennial problem with the growing number of users joining recommendation systems every day;
- Avoiding the burden to users who have to provide their personal information and preferences over and over again in recommendation systems while it is already there in social networks,
- Permitting to recommend any type of resources (or items) including Short Life Resources (SLR). SLR are the resources (like news in a newspaper or offers on products in a commercial site) that live only for few days and that are usually not easy to recommend using collaborative filtering techniques, since they cannot gather enough ratings to be recommended;
- Using raw data (implicit data) from users benefit from its availability and the fact that it does not vary significantly over time like the explicit ratings that may vary (e.g. a user sees a better film so he/she changes from consistently bad reviews to suddenly favorable reviews). It is a well-established wisdom that explicit ratings are often hard to obtain and getting information passively (e,g, as in Amazon) leads to richer results for users.

- In any case, iSoNTRE can join any working recommender. When the matrix is built it can be added to any existing recommendation matrix in order to enhance the recommendation quality in the existing systems and produce recommendations for new users.

To summarize, using GPSN as information goldmine may facilitate the use and the combination of recommendation systems, which in turn may enhance the variety and the satisfaction of items consumption as shown in [2]. The questions to answer in this area of GPSN exploitation are:

Question 1: Do GPSN contain representative information about the user or are they instead only a self-idealization that is less informative for effective recommendation?

Question 2: Can we predict form the users' spontaneous actions in social networks information that was not provided explicitly by them?

Question 1 is not the focus of this work as it was addressed by prior work in the social network domain. However, Question 2 is a focus and we address it via designing and algorithm as well as a methodology for collecting or eliciting data, and then evaluate it via surveys of user groups. In summary, 63 Facebook surveyed users said they did not join Facebook for commercial purposes; however we were able to predict for most of them a commercial profile that they acknowledged as representing their commercial interests accurately. As well as in Twitter the profiles of 12000 users contained around 3,000,000 of commercial concepts as will be shown later.

Based on the results and previous discussion we introduce the following contributions in the next chapter 4:

Contribution 1: Define iSoNTRE, a hybrid social recommender machine that is designed to transform the GPSN into a recommendation engine. It transforms the raw data of GPSN into a useful data for recommendation. To the best of our knowledge, iSoNTRE is the first recommender that addresses such questions. iSoNTRE gathers both collaborative filtering with content filtering techniques using an external conceptual source to process as a middle layer between the users and items to be recommended.

Contribution 2: Evaluating iSoNTRE, with both offline evaluation using the metrics (MAE, RMSE), as well as an on-line evaluation, using avatars (Sirens) on Twitter and Facebook applications.

We see iSoNTRE as a step forward in building the social machines of the future introduced by Tim Berners Lee in his book [3], and in his talk in the www conference at Lyon (2012).

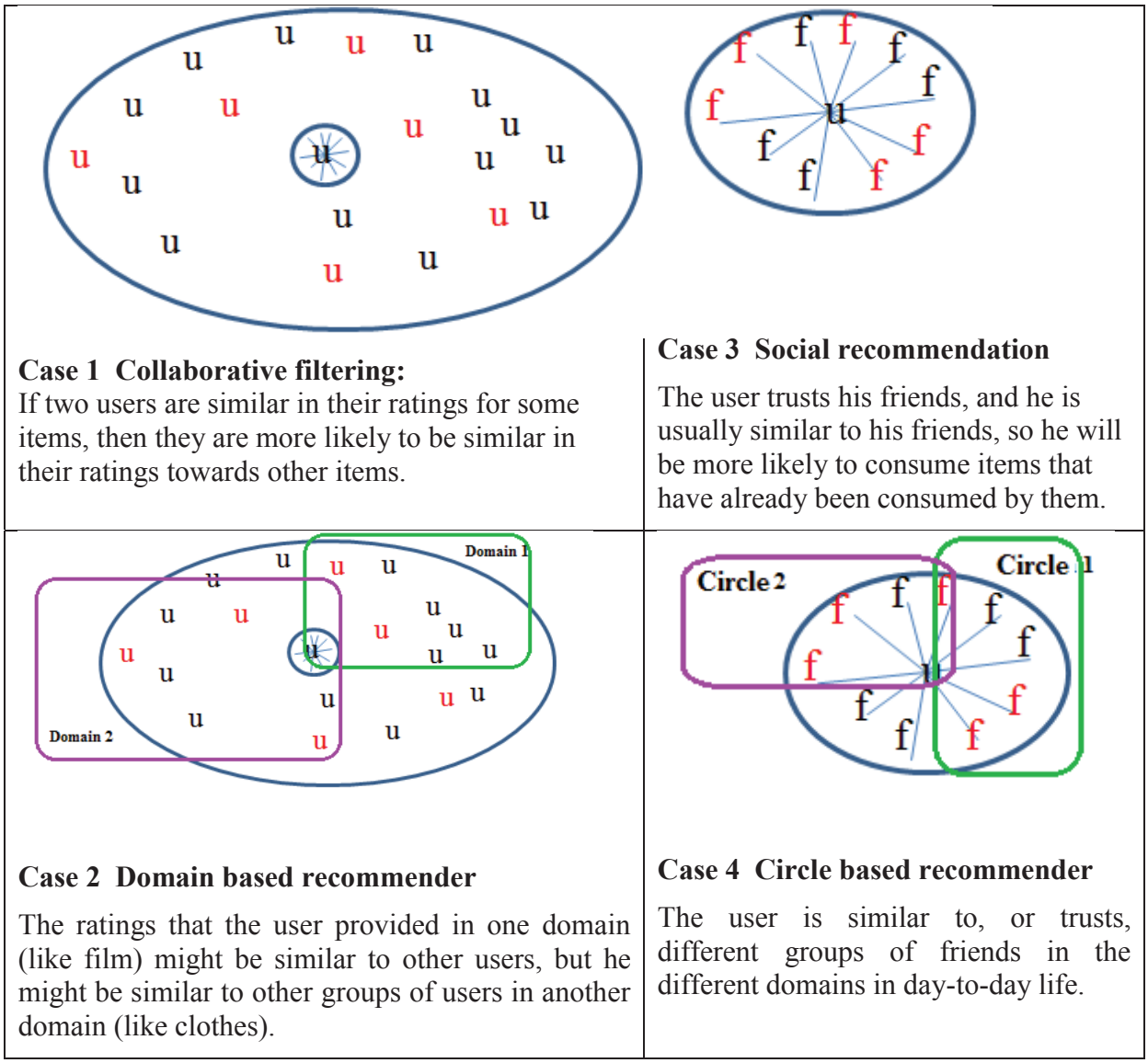
1.2. Recommendation assumptions:

Each recommendation system lies upon an assumptions, the iAmélie system's main contribution is that it makes a step backwards and tries to answer different assumptions at the same time. Where in the chapter 6 we will explain how iAmélie works, we will classify the main recommendation strategies into four families, which are based on recommendation systems history and the works introduced in related works' chapters. The goal of this classification is to discuss the claims and assumptions that are behind them which will be the basic of iAmélie system. The assumptions that we will detail in chapter 6 are summarized in figure 8. The main idea in iAmélie is that the same user usually needs different assumptions in different cases, so that a system that answers different assumptions at the same time is much more preferable by users. These assumptions can be seen as a kind of context related to user's

goal of using the system that is why we classify iAmélie as a contextual hybrid recommender. Besides the discussed assumptions iAmélie considers a more assumption that no existing recommender system takes in consideration:

Some of my friends might not be like me, but they know me well.

Later in chapter 6 the model will be introduced and evaluated in details.



Case 1 Collaborative filtering:
If two users are similar in their ratings for some items, then they are more likely to be similar in their ratings towards other items.

Case 3 Social recommendation

The user trusts his friends, and he is usually similar to his friends, so he will be more likely to consume items that have already been consumed by them.

Case 2 Domain based recommender

The ratings that the user provided in one domain (like film) might be similar to other users, but he might be similar to other groups of users in another domain (like clothes).

Case 4 Circle based recommender

The user is similar to, or trusts, different groups of friends in the different domains in day-to-day life.

Figure 8 The different recommendation systems assumptions

1.3. iSoNTRE and iAmélie together:

We decided to separate iSoNTRE and iAmélie into two systems to give the possibility to focus on each system, present its model and evaluate it in an appropriate way, as well as to give the possibility to adopt each one alone in other systems where it is suitable to be adopted. After all the two systems are totally compatible together:

When building iSoNTRE based on global world knowledge and a GPSN we will have a collaborative filtering recommender (case 1) in figure 8, while if specific world knowledge was used like shopping or film world knowledge, then a domain based recommender will be build (case 2). At the same time, as we will build iSoNTRE from a GPSN then we do have the social relation of a user, his friends and trusted people, so if we modify the similarity in iSoNTRE over a global world knowledge to integrate friendship and trust relations then the social recommendation will be obtained (case3). Finally if we include friendship and trust relation over a specific world knowledge then even the circle based recommender will be obtained (case 4).

The rest of the chapter will cover important issues to discuss before building to two models; first, we'll discuss if social networks contain real information about users or are they self-idealization about them. And then o what is there in GPSN (Twitter, Facebook); can we extract more information that was not provided explicitly from users' spontaneous actions? Then we will refer to social networks as a source of contextual information.

2. Issues to address:

2.1. Are social networks a Self-idealization of users?

In DBSN this question may be ignored, as users use these sites in order to solve a specific problem like finding a film (Flixster) or a product (Epinions). In these cases, users are expected to provide exact information in order to get good recommendations.

Contrary to this are the GPSNs, the focus of our work. Common opinion is that people do not show their true personality on social networks; instead, they draw a picture that they want others see on their Facebook accounts [4]. However, recent studies showed that users do place true parts of themselves onto social networks [5]. In [6], five personal traits were extracted only by taking into consideration some information of users' profiles on Facebook. For Twitter, in [7] authors affirm that people use this micro blogging site to talk about their daily activities and to seek or share information about themselves and their activities. Thus, transforming GPSNs into a source of recommendation is a meaningful procedure that may help users.

2.2. Can More Information Be Extracted from Users' Spontaneous Actions in GPSNs?

First, we tried to consider a domain that was not the focus of users when they joined a GPSN. We held a questionnaire on 63 Facebook users (most of them were students, aged between 19 and 20 years) that posed only one question: why did you join Facebook? There were five answers to choose from: 1.) curious about others; 2.) fun; 3.) keep in touch with friends and family; 4.) shopping purpose; 5.) keep up to date with latest news. Users were able to choose one or more answers. The results are in Table 10.

In the table we notice that none of the 63 users joined Facebook for commercial or shopping purposes. Although this group of 63 users is not a representative sample but the fact that they all declared not joining Facebook for commercial purposes is significant.

This is why we chose the commercial domain to investigate: <u>can we predict users' commercial profiles from their actions in GPSNs (Facebook and Twitter).</u>

Why have you joined Facebook	
Curious about others	49
Fun	35
Keep up to date with friends & family	43
keep update with last news	26
Shopping purpose	**0**

Table 10 From 63 Facebook users no one joined Facebook for shopping purpose

The procedure that we propose to investigate this question is the following:

Commercial profile building Procedure:

1. *Data Extraction:* Extract information about users from GPSN (Facebook/Twitter).
2. *Data Cleaning:* For each user we applied the a cleaning [8]:
   - Convert all characters to lowercase characters.
   - Remove 334 stop words from the source code of the Gensim framework.[16]
   - The punctuation marks were cleaned.
   - Every link in the tweets was reached and the main keywords of the link (in the title) were added to the user profile.
   - Stem (Porter stemer) all the tweets, re-tweets, and replays.
3. *Knowledge Extraction:* Use an external world knowledge (in the shopping domain) to see if the extracted information contain commercial information from the external world knowledge
4. *Commercial Profile Building:* See how many times commercial concepts appeared in an extracted user profiles. In order to build a commercial user profile like a cloud of commercial concepts. For example, if a profile contained 5 times the concept Nike and 3 times a concept Amazon this is an indication of how much such a user is interested in these concepts.

2.2.1. World Knowledge Sources:

The ODP (Open Directory Project) of Wikipedia is general, covers a large variety of concepts, and offers links between them. In some specific domains, we find other well-known knowledge domain concepts. In the branding and shopping domain eBay -Figure 9- and amazon are excellent examples of shopping domain knowledge. Although amazon is well known for its recommendation by item-item collaborative filtering, we cannot ignore the high role that the categories play in the methodology of the system. The branding knowledge used by eBay can be downloaded[17] and it contains the categories and relations between them. Usually it is linked by items, so that every item belongs to a certain category, exactly like the article that belongs to different concepts in Wikipedia.

In the commercial profile extraction, we use Blue Kangaroo Tree (BKT) world knowledge. It is similar to the eBay one, it has been provided to us by the Blue Kangaroo team[18]. BKT contains the brands -mainly American-, numbering 2,000 brands and 17,000 categories that are related to each other.

[16] http://radimrehurek.com/gensim
[17] http://www.cgmlab.com/ebay-category-tree-download/
[18] it is available from the authors by mail request

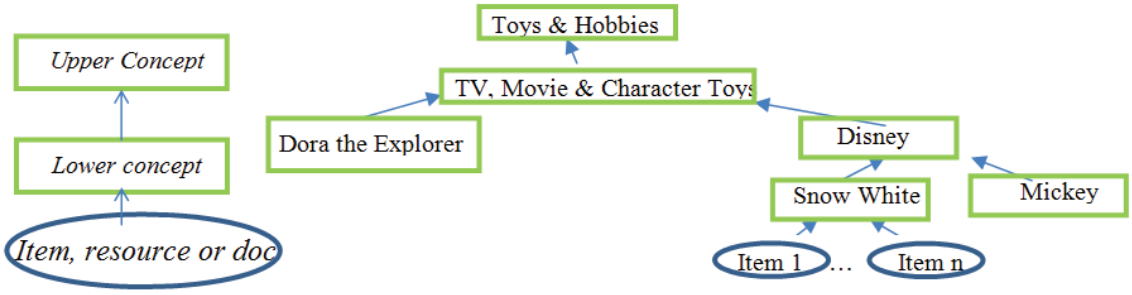


Figure 9 Part of the branding domain knowledge tree of eBay which is similar to Blue Kangaroo, Amazon and the other commercial sites.

2.2.2. Is there commercial information on Facebook?

To the best of our knowledge no available study has considered the commercial aspect of Facebook. Since we could not find the available Facebook datasets that contain enough information for this test, we had to collect the data from Facebook another way. Because Facebook is not an open network like Twitter, we couldn't collect the user information unless users accepted this practice. Since data collection can be done through Facebook applications, we designed one for that purpose (Roo-are-you -Figure 11-). We added some questions that will help us in the application.

1. We asked the user if he thinks that we can extract commercial information from his Facebook profile. (Yes/ No)
2. If the user gave permission to our application, it began to extract all the user's likes in order to build his commercial profile.
3. We showed the extracted profile to the user and collected his feedback about it – Figure 12-

The procedure then is the ***Commercial profile building Procedure*** introduced before, an example of a commercial profile is in Figure 10. Where the users information is the Facebook likes, the world knowledge used is the BKT, based on both the user commercial profile is built, then it is evaluated by the user:

4. Ask the user about his Feedback about the extracted profile Table 11.
5. Give the user, the possibility to delete/add commercial concepts, or the change the size of a concept, knowing that the bigger a concept is, the more important it's meaning to the user Figure 12.
6. See the difference between the extracted profile and the changes provided by the user.

Figure 10 an example of a commercial cloud

Evaluation of the extracted profile lays in the last three steps (4-6), based on the difference between the proposed profile and the updated one, in addition to the live feedback from the sessions. Two sessions of evaluation were conducted with the 63 Facebook users (university students and friends). Based on these 63 users, we were able to collect information of around nearly 2,000 Facebook users who are their friends (some users asked us to delete their profiles after the test, and we respected their wish; others provided us only with their information and not their friends' information).

Figure 11 the welcome page in the roo are you Facebook application

Questions during the Roo-are-you Facebook app	**Yes**	**No**	**I don't know**
Can we predict your commercial profile of FB	8	23	32
Do you agree this shopping profile describes you?	49	10	4
Were you surprised?	45	18	0
Do you think it is useful in real life?	52	6	5
Do you think there is a privacy issue here?	44	9	10
Do you remember you did likes on brands on Facebook?	19	29	15

Table 11 the Facebook test over 63 Facebook users

2.2.3. Discussion on Facebook commercial profiles

User Motivation for the App: As a general remark, users were motivated for the test; some of them were astonished during the test and were interested in knowing more and trying other applications. Users who did not have any profile information were the passive users [6]; those who pearly do actions on Facebook they only look at other actions and see the news there.

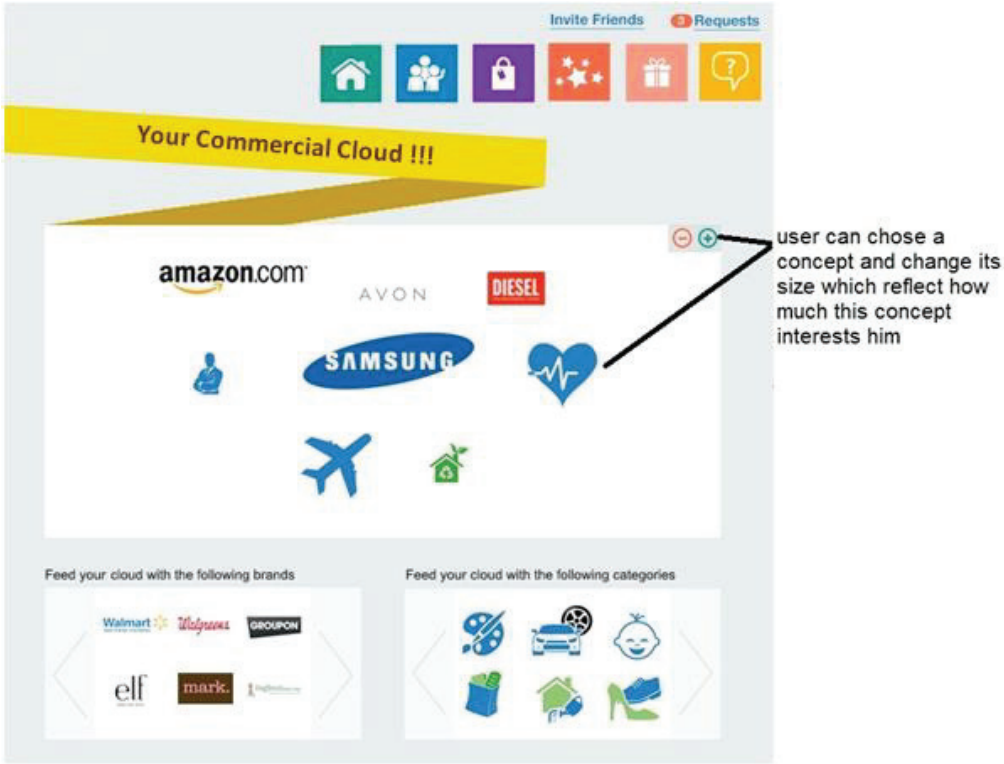


Figure 12 the user cloud user can add concepts, delete ones or even change the size of a concept which reflects a correction to the extracted profile.

No Shopping purposes and 49 commercial profile: As described before none of our 63 users has joined Facebook for a commercial purpose. We were able to have 49 participants who agree that the commercial profile described them.

Not remember liking commercial brands: 29 users didn't remember clearly that they had liked commercial concepts, or brands on Facebook, but most of them in fact did. We did not investigate this issue in detail, so it remains an area of interest to explore.

The tendency to add commercial concepts: in (4-6) steps users can add or delete concepts to their clouds, we noticed a tendency to add concepts, categories and brands to their profile. This is a good indication that what we extracted from Facebook reflected them, but it is not all about them. The added information to profiles was estimated to be about 30% of commercial concepts, some users deleted some items that was about 7%.

Privacy issues: with the 63 surveyed users we noticed that they consider Facebook itself is an attack on privacy, and when they put data on it, this data could be used for plenty of different goals. This affirms the assumption that was discussed years ago in [4], where the author affirmed that users are aware of the risks that they might be exposed to due to using Facebook, but their desire to share and join others leads them to ignore this risk. However we did not investigate into detail the privacy issues in our work.

Over all commercial information: Based on the 63 users we extracted the friends' information for those who gave us the permission, we had nearly 2000 Facebook account. In these 2000 profiles we had around 538 commercial concepts that has been repeated 30,105 times where the user cloud contained between 15 and 25 commercial concepts as a mean.

2.2.4. Is there commercial information on Twitter?

In our study we have analyzed the accounts of nearly 12,000 users on Twitter to examine if we can find commercial information in their profiles. Since Twitter is an open social network, users' information is accessible through the Twitter API, -unless the user changes the default options of privacy-. The data collection was performed using PHP programing language and the data was collected in a mongoDB. The computer used was a Lenovo Intel Core i7 with 8 GB RAM. The crawled information contains users' tweets, re-tweets, mentions, links, hash tags, replays, and the country information of each user. One issue that had to be dealt with was that some people created an account then didn't use it. In order to avoid using unused accounts in our study we applied the following rules:

- The user's account should have at least 150 tweets.
- The user should be from the U.S. (this is because the world knowledge that we use BKT contains mainly American brands). The test was done on the country and location.
- The account should not be a company account. As the test will be conducted in the shopping domain, the preference was to have accounts belonging to women. In order to achieve both requests, the crawled account name should contain a woman's name because most of people use their name as a part of their accounts.[19]

The same steps used to clean the users information (step 2 in section 2.2. from the commercial profile extraction procedure) was applied over BKT concepts. Knowing that every link in the tweets was reached and the main keywords of the link (in the title) were added to the user profile. The next step was to extract every mention of commercial concepts from BKT from the users' tweets and actions, (step 3 in the section 2.2. in the same procedure).

Twitter data view:

We found nearly 1030 commercial concepts in the extracted dataset; these concepts were mentioned 3,184,788 times, as shown in Table 12.

Number of commercial concepts in users profiles towards how many times concepts were mentioned			
Number of concepts that was mentioned less than 1000 times in the data set	768	# times these concepts were mentioned	2,421,790
Number of concepts that was mentioned more than 1000 & less than 10,000 times in the data set	200	# times these concepts were mentioned	648,222
Number of concepts that was mentioned more than 10,000 times in the data set	62	# times these concepts were mentioned	114,776
# concepts	**1030**	# total of	**3,184,788**

Table 12 Number of times in which shopping related concepts were mentioned

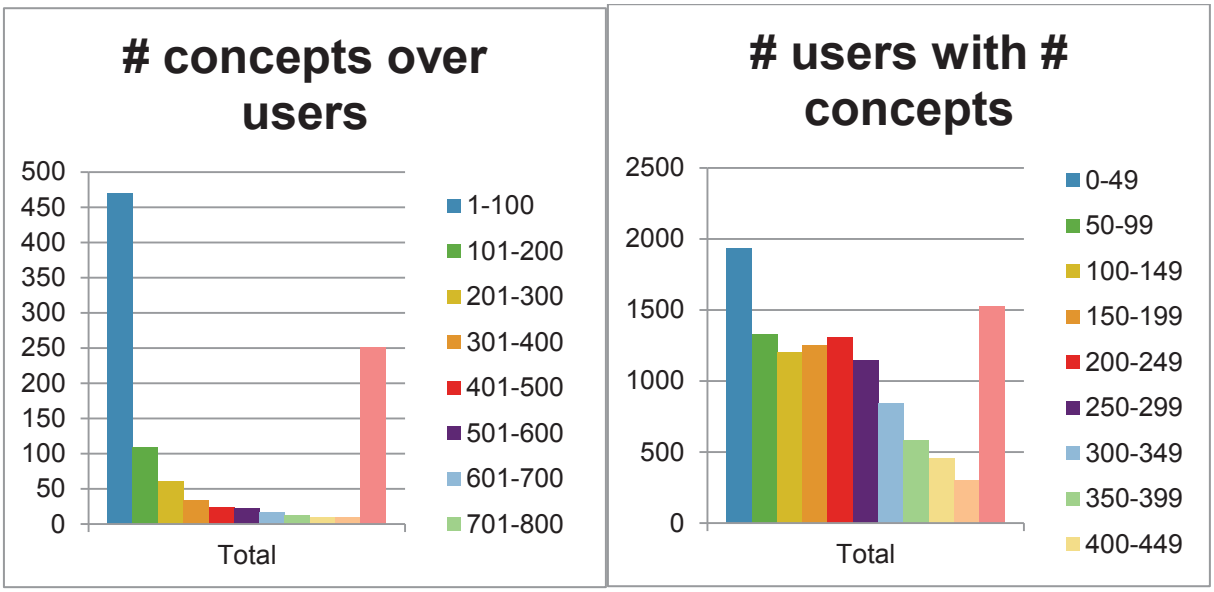
From this pool of information, we listed the most mentioned concepts in our data set in table 12.

[19] http://www.babynameguide.com/categoryamerican.asp

# times they were mentioned	Concept	# times they were mentioned	Concept
202976	Summer	46933	Oral Hygiene
165926	Amazon	45939	Coffee Tea
146990	RV Camper	42590	Fab
126261	Greeting Cards	42373	Walmart
122186	Books	41261	Visa
96884	Spring	36683	eBay
95790	Toys Games	35031	Food Groceries
70210	Apple	34259	Starbucks
63842	Private Flash Sales	27582	Makeup
62437	GUESS	23587	Flowers
61173	Office	20992	iTunes
58177	Graduation	17883	Samsung
49435	Kids Shoes	14826	Laptops
49275	Thanks giving	14624	Vacation Packages
		13331	Hotels Resorts

Table 13 Top mentioned commercial concepts and how many times each concept has been mentioned

In order to better see the data in Figure 13, we show the distribution of users over the 1,000 most mentioned concepts. In Figure 13 (A) we have nearly 450 commercial concepts that have been mentioned by between 1-100 users, and 100 concepts that have been mentioned by 101 -200 users, while in Figure 13 (B) we notice that nearly 2,000 users have between 0 and 49 commercial concepts in their profiles, and nearly 1,500 users have between 50 and 99 commercial concepts in their profiles.



A B

Figure 13 Distribution of users among commercial concepts in the twitter extracted data.

In order to better understand these extracted numbers we compared them with numbers extracted from two data sets from epinions, knowing that epinions is a product-oriented social network; unlike Twitter, which is an open social network. We wanted to compare the number

of commercial actions done by users in Epinions (reviews on products) with commercial actions in Twitter (mentions of commercial concepts).

The first epinions data set contained 2,805 users. Nearly 2,100 users had between 0 and 49 reviews; about 300 users had between 50 and 99 reviews. Figure 14 shows the results of this data set.

The second data set [11] comprised of 75,891 user data, where we had 27,947 users who did not provide any review; nearly 36,000 users had provided between 1 and 49 reviews; and about 13,000 users provided between 51 and 100 reviews. The total number of reviews done by all users was 681,280. The distribution is shown in Figure 15.

We conclude that Twitter is a GPSN but the spontaneous actions in it contain a lot of commercial information that need to be exploited and understood. Even when compared to shopping social networks like epinions, the amount of commercial information in Twitter is not at all negligible.

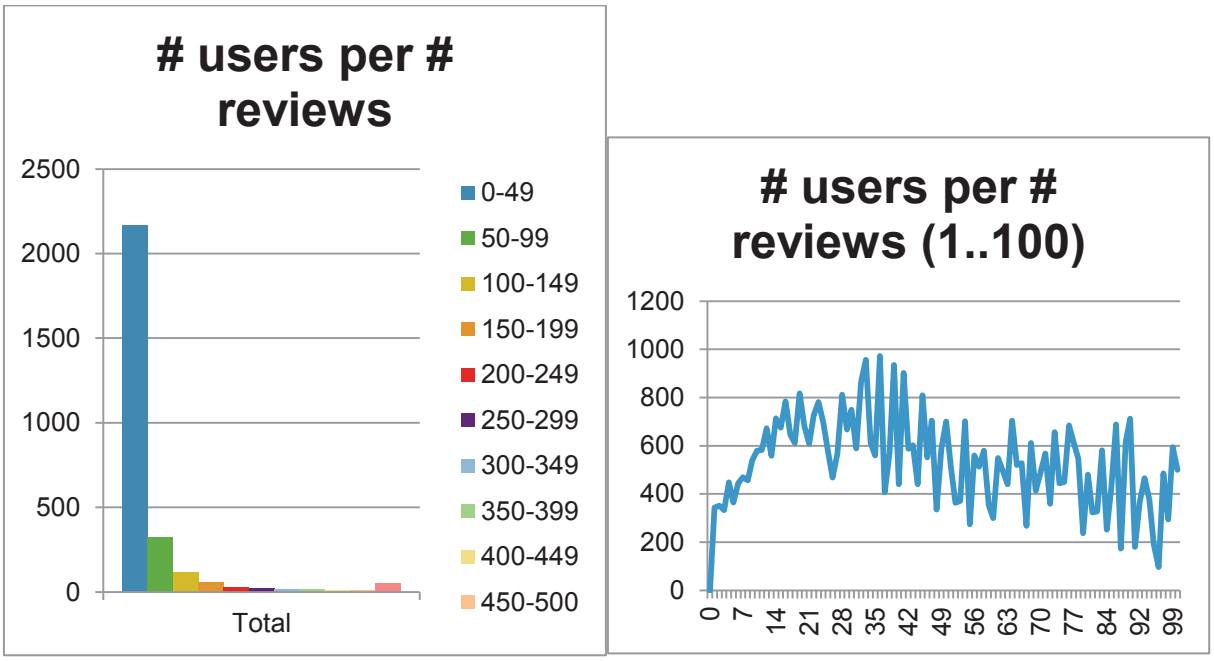


Figure 14 Over a 2805 users a data set of epinions, nearly 2100 users have between 0 and 49 review, although epinions is a shopping social network.

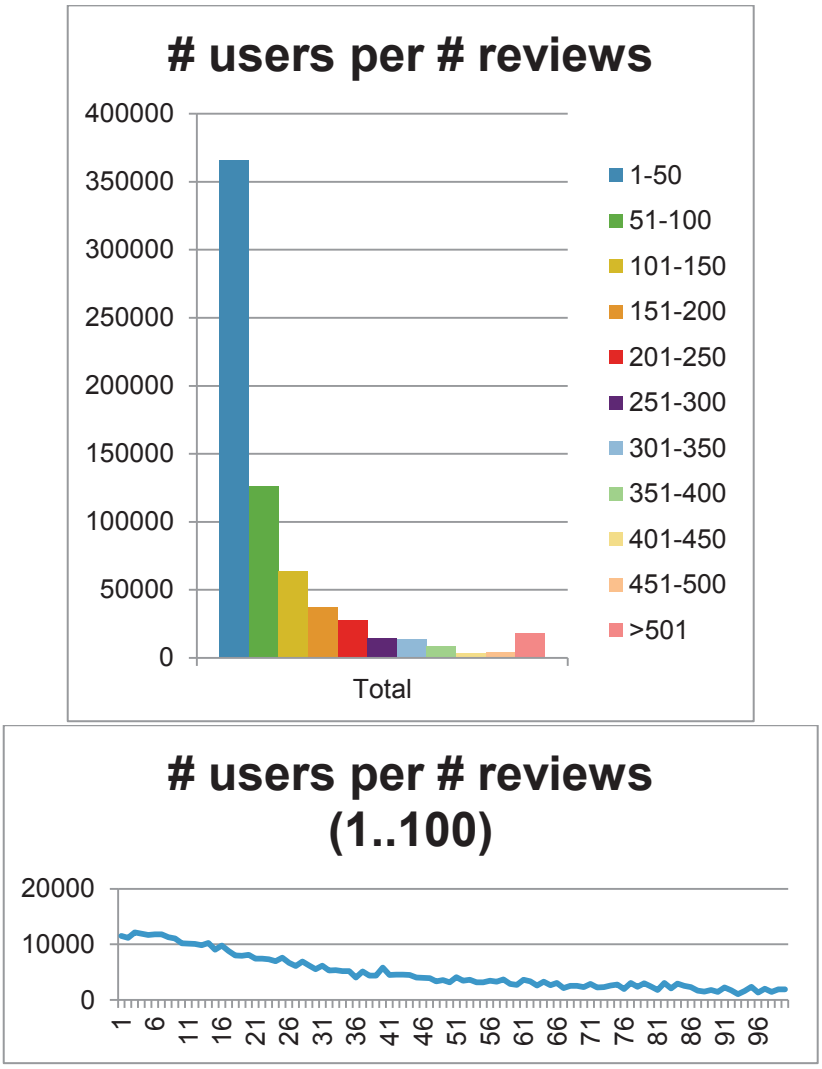


Figure 15 of a dataset of 75,891 epinions users. A lot of users with only little participation.

3. Social Networks as a Source of Context?

Physical context can be extracted from social networks if the user has explicitly added it, like the places that he/she has visited. The example in Figure 16 shows the places provided by a user on Facebook. The temporal information can be extracted from the time the user completed the actions on the social networks, as in Figure 17. The same process can be done on Twitter where the time of tweet is provided within the tweet's information.

Figure 16 Places visited by a user posted on Facebook, with time stamp

Figure 17 Time stamp on actions.

With the examples above, it is evident that we can extract the social context from social network; in other words, we can extract the user's friends and even classify these friends in groups based on the common interests as shown in [32] for example.

In some instances, the user can express his/her feelings explicitly on the social networks. To date, some studies have extracted emotion from social network actions, allowing the feeling of users on social networks to be known as shown in in figure 18. However, feelings extraction from actions on social networks is out of the scope of our work.

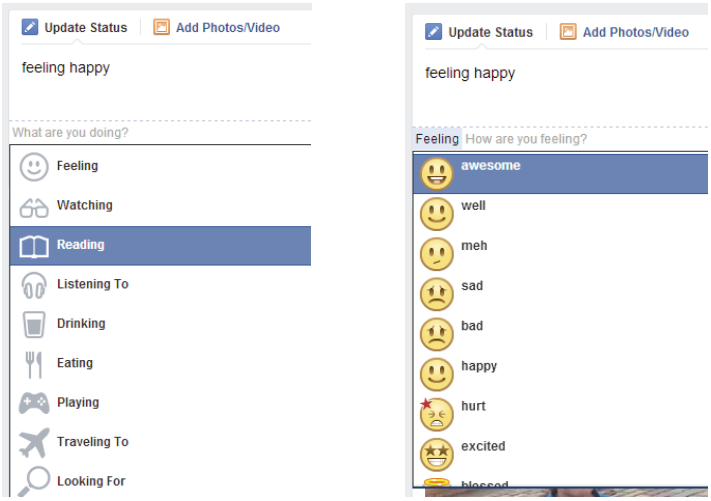


Figure 18 User can even provide his/her internal state clearly in social networks., Example from Facebook

4. Conclusion:

People use online social networks such as Twitter and Facebook for different goals and carry out different actions, which contain a variety of information, therein. In this section we were able to investigate if we could find information in a domain that is not explicitly defined as a goal for GPSN users, which is the commercial profile. Surprisingly, our conclusion is yes; in users' social information there are commercial trends that can be extracted, even if social networks serve a general purpose and are not intended to be commercial networks.

One reason for this might be the widespread use of social media and social marketing, which invite people to communicate around commercial issues and brands in online social networks. This would mean that people like, comment or conduct commercial actions without being aware of them.

At the recommendation level we conclude that GPSNs contain information about users that can be useful in the recommendation domain. Clearly Facebook can be useful in recommending books and movies, which are among the defined categories within it. And based on the previous discussion, we determined that GPSNs can be useful in also recommending commercial items, as we were able to extract users' commercial information from these social networks. Now, we can move to introduce iSoNTRE, the social network transformer into recommendation engine, which can be applied over sopping domain, or any other one.

CHAPTER 5: ISONTRE MODEL

Towards a Social Machine for Recommendation

"Computers can help if
we use them to create abstract social
machines on the Web: processes in which
the people do the creative work and the
machine does the administration"

Berners-Lee, Weaving the Web, 1999

1. Introduction:

GPSNs contain a lot of users' information over different domains, such as movies they like, and books they read; as well, it's possible to extract useful information from different domains, like the commercial ones based on the users' spontaneous actions on these GPSNs. iSoNTRE is a social recommender that is designed to transform all the richness in GPSNs into robust recommendation systems. It can be seen as a social recommender that can be applied over different domains, transforming users' information into recommendation information. Therefore, iSoNTRE saves users efforts from having to enter what they like in different systems, and saves them from the cold start problem in new systems. In this way, there is no need for users to provide their information in different systems if they did that without request in a GPSN.

iSoNTRE is a hybrid recommender; it combines content- and collaborative-based recommenders. It works as an opposite of the matrix factorization mechanisms. Instead of decomposing the tremendous recommendation matrix into a smaller matrix, iSoNTRE uses content-based techniques in order to build two matrixes: one containing users, concepts, and level of interest of a concept by the user matrix and a resource, concept, level of interest matrix, then it combines the two in order to get a whole matrix of user, resource, level of interest. The level of interest can be seen as a predicted value of how much the resource is interesting for the user. In other words, it is an equivalent to the predicted rating in recommender systems. The work in building the whole matrix based on the extracted ones is similar to the work in [8]. However, the authors of the paper proposed a model to enhance the film recommendation in movie systems and they do not handle the social aspect of recommendation, while iSoNTRE is about how to transform the GPSN in general into recommendation engine. From another perspective iSoNTRE uses a similar method to that mentioned paper in order to transform the GPSN into recommendation engine. Therefore, we will begin by introducing the matrix factorization algorithms.

2. Matrix Factorization Overview:

Perhaps, the most accurate model in the recommender field is matrix factorization [9], which is why it has been well studied in the recommendation solutions. Even in social recommendation models, matrix factorization is the most used technique [10]. Although this model is not practical in the industry because of the long time needed to prepare the model, the high accuracy level keeps attracting researchers to investigate its details. The most used approach in matrix factorization is the singular value decomposition (SVD) [10]; however, one of its drawbacks is that the model cannot be understood by users, as when reducing the size the detailed information is lost, making it impossible to explain the recommendation to users.

Matrix factorization's basic idea is to move from a recommendation matrix, which is usually very big and sparse, to a reduced matrix that has compacted the information. In this family of methods, the rating estimation $\hat{r}_{um}$ that a user u would give to an item m is estimated as an affinity measure between the user and the item, both characterized in the latent factor space with a pre-established dimensionality f:

$$\hat{r}_{um} = \vec{U}_{u \to R^f} . (\vec{M}_{m \to R^f})^T \qquad (21)$$

Where $\vec{U}_{u \to R^f}$ and $\vec{M}_{m \to R^f}$ represent the characteristics of user u and item m in the latent factor space R^f. The used affinity measure is the dot product. In the latent space R^f if the

components which characterize the user u is $\vec{U}_{u\to R^f} = [U_{u1}, U_{u2}, \dots, U_{uf}]$, and the item vector components is $\vec{M}_{m\to R^f} = [M_{m1}, M_{m2}, \dots, M_{mf}]$, so the dot product is the following:

$$\hat{r}_{um} = \sum_{i=1}^{f} (U_{ui} \,.\, M_{mi}) \qquad (21)$$

This characterization of the user and item has been found to minimize the prediction error e_{um}, which can be calculated like this:

$$e_{um} = \left(r_{um} - \sum_{i=1}^{f} (U_{ui} \,.\, M_{mi}) \right)^2 \qquad (22)$$

A regularization coefficient β is introduced to overcome overfitting, this β penalizes the norm of the user and item vectors. So that the regularized prediction error $\ddot{e}_{um}$ will be:

$$\ddot{e}_{um} = e_{um} + \beta \left(\|\vec{U}_{u\to R^f}\|^2 + \|\vec{M}_{m\to R^f}\|^2 \right) \qquad (23)$$

Then the user and item vectors have been found to minimize the regularized prediction error over a set of known ratings.

$$min_{\overrightarrow{U_u}, \overrightarrow{M_m}} \sum_{r_{um} \in R \,\wedge\, r_{um} \neq 0} \left(r_{um} - \sum_{i=1}^{f} (U_{ui} \,.\, M_{mi}) \right)^2 + \beta \left(\|\vec{U}_{u\to R^f}\|^2 + \|\vec{M}_{m\to R^f}\|^2 \right) \qquad (24)$$

In this expression, the matrix $R_{U \times M}$, of size $U \times M$, where U is the number of users and M is the number of items. Unknown ratings r_{um} are assigned to 0, while the known ones are usually in the interval [1..5].

3. iSoNTRE Model:

In traditional social recommender systems like those surveyed in [27], after having information from the recommendation-based social networks, different recommendation methods can be proposed and evaluated directly -Figure 19-. An example of work can be found on Epinions or Flicker as they are recommendation social networks. However, these works can't be automatically extended to GPSNs because in these social networks we need to deal with the raw data beforehand. That is what iSoNTRE do.

In order to work, iSoNTRE extracts users' information, preprocesses it, and translates it to a source of recommendation. In addition, it is based on world knowledge in different domains, each world knowledge containing concepts that will work as a middle layer between resources and users: this world knowledge is needed to filter the extracted profiles over a specific domain so that user will end up having a group of concepts that are interesting for him in a specific domain, such as shopping or films. This leads to having a recommendation engine over each domain (film domain, products domain, books domain, etc.)

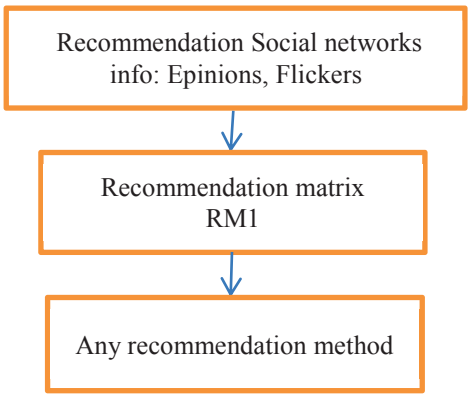


Figure 19 the process in traditional social recommenders, although works propose theatrical methods, they can't be applied on GPSNs, which need to deal with raw data

The Open Directory Project (ODP) web directory or Wikipedia encyclopedia is a very common example of world knowledge used in some works [20][22]. Like ODP, any other source of related concepts or ontology can be used as world knowledge. In the shopping domain, e-Bay one or Blue Kangaroo Tree (BKT), introduced in the previous section, can be used.

The way in which iSoNTRE works can be divided into two main tasks: first, iSoNTRE handles the raw data in order to build the recommendation engine; second, iSoNTRE offers the recommendation.

3.1.1. The Social Network Transformer: Handling The Raw Data:

iSoNTRE first extracts users' information from a GPSN. The extracted information varies from one social network to another. So in Twitter iSoNTRE extracts the tweets, re-tweets, and replays. In Facebook, iSoNTRE extracts all the likes of a user, his main information, and information from his wall.

The second step is to clean the extracted data using the same steps as in the previous section 2-2 in chapter 4 so that each user u will end up by having a global profile $\overrightarrow{G_u}$. The global profile is a bag of meaningful concepts that can be repeated in the profile many times. An example of a general profile of a user u is:

$$\overrightarrow{G_u} = (sport, football, France, sport, romance, video, hokey, Italie, love \ \ldots)$$

Third step is to filter the global profiles over concepts of domain(s) $\vec{D}$.. Each domain $\vec{D_i}$ contains a number of concepts X in order to filter the user's global $\overrightarrow{G_u}$ profile over the domain $\overrightarrow{D_i^X}$ an intersection function is used.

$$\overrightarrow{G_u} \cap \overrightarrow{D^X_{i:Sport}} = \overrightarrow{U_D} \qquad (25)$$

In the previous example, if we apply an intersection between the user global profile $\overrightarrow{G_u}$ and a sport knowledge domain $\overrightarrow{D_i^X}$ we'll end by having the following sport profile of the u:

$$\overrightarrow{U_{D:sport}} = \overrightarrow{G_u} \cap \overrightarrow{D^X_{i:Sport}} = (sport, football, sport, hokey \ \ldots)$$

So far we have elaborated on the decomposition of the user's general bag of words into different bags of concepts using different world knowledge. Now comes the step of finding the each user's level of interest toward each concept, which will either be extracted then normalized to be considered as a predicted level of interest, or be an extracted rating if the user was to give a rating to this concept.

For each concept for each user the frequency of how many times the concept is addressed in the user profile is taken into consideration as an interest measure. In the previous example we have for the user in the sport domain the following frequency:

$(sport: 2, football: 1, hokey: 1 \ \ldots)$.

3.1.2. User Concept Matrix:

In each domain $\overrightarrow{D_i^X}$ we have X concepts, so we define a domain matrix that contains rows of all the users U, and as columns all the concepts X. The values of this matrix in its $u(i,j)$ elements are either the extracted rating of the user i towards the concept j if it exists or 0 otherwise. As the data will be normalized (Z-Score) the 0 will not inter group the data. The user can be expressed by a vector $\vec{U}_{1\to R^X}$.

$$U_{U\times X} = \begin{bmatrix} U_{11} & \cdots & U_{1X} \\ \vdots & \ddots & \vdots \\ U_{U1} & \cdots & U_{UX} \end{bmatrix} = \begin{bmatrix} \vec{U}_{1\to R^X} \\ \vdots \\ \vec{U}_{U\to R^X} \end{bmatrix}$$

Figure 20 shows the main steps in the transformation process from the general social networks into this user-concept matrix. The use of world knowledge in order to extract concepts from users' profiles is extracted from the content-based family of recommenders.

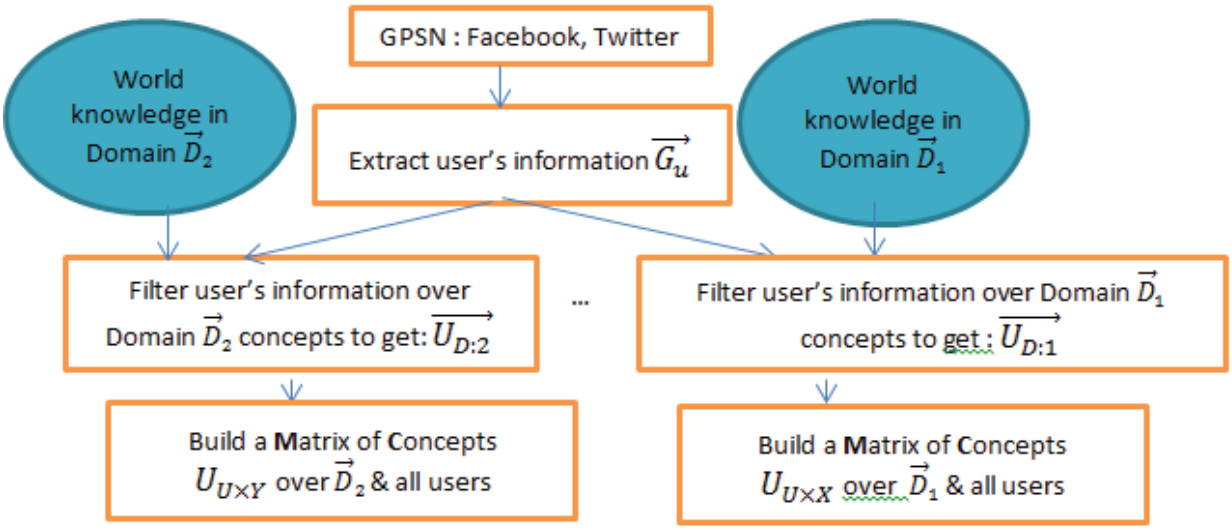


Figure 20 The way to transform GPSN raw data into user, concept matrix over the different domains

3.1.3. Resource Concept Matrix:

The goal of any recommendation system is to recommend resources; this process involves a treatment on the resources in order to be recommended.

iSoNTRE performs on resources $\vec{M}$ the same filtering operation done on the users. Logically it uses the same world knowledge used over users $\overrightarrow{D_i^X}$ which is related to the resources kind. This means that if the resources are in the shopping domain the world knowledge is in this domain, while if the resources were in the movie domain then a world knowledge in the movie domain should be used; in both cases, we need to have users' profiles on the chosen world knowledge.

The first step is to clean each resource *m* using the same steps as in the previous sections. Each resource *m* will end up having a vector, or an item profile $\vec{M}_m$ that is a bag of meaningful words, which can be repeated in the item profile many times. An example of an item vector is:

$$\overrightarrow{M_m} = (sport, clothes, outdoor, indoor, sport, football)$$

The second step is to filter the item profile over concepts of domain(s). The domain $\overrightarrow{D_i^X}$ is the same one that has been chosen in order to build the user-concept matrix, containing the number *X* of concepts, in order to filter the user global profile $\overrightarrow{G_u}$ over the domain $\overrightarrow{D_i^X}$ an intersection function is used.

$$\overrightarrow{M_m} \cap \overrightarrow{D_i^X} = \overrightarrow{M_D} \qquad (26)$$

In the previous example, if we apply an intersection between the user global profile $\overrightarrow{M_m}$ and a sport knowledge domain $\overrightarrow{D_{i:Sport}^X}$ we will end by having the following concepts of the resource *m*:

$$\overrightarrow{M_{sport}} = \overrightarrow{M_m} \cap \overrightarrow{D_{i:Sport}^X} = (sport, football, \dots)$$

To find the level of importance of each concept in this resource we count how many times each concept has appeared in the item profile, then we apply the same normalization process adopted in the case of users.

In the previous example we have for the resource $\overrightarrow{M_m}$ in the sport domain the frequency as following: $(sport: 2, football: 1 \dots)$.

In each domain $\overrightarrow{D_i^X}$ we have *X* concepts, so we define a domain matrix that contains as rows all the resources (*M* ones), and as columns all the concepts *X*. The values of this matrix are either the extracted normalized value of interest if it exists or otherwise 0. As the data is normalized with Z score, the 0 will not inter group the data. Each resource profile can be seen as a vector $\vec{M}_{1\to R^X}$

$$M_{M\times X} = \begin{bmatrix} M_{11} & \cdots & M_{1X} \\ \vdots & \ddots & \vdots \\ M_{M1} & \cdots & M_{MX} \end{bmatrix} = \begin{bmatrix} \vec{M}_{1\to R^X} \\ \vdots \\ \vec{M}_{M\to R^X} \end{bmatrix}$$

3.1.4. Building the Recommendation Engine (the Recommendation Matrix)

We will use the canonical form of matrix factorization to express the matrix of estimated ratings $\hat{R}_{U\times M}$ as an affinity measure between the extracted user profile matrix $U_{U\times f}$ and the item profile matrix $M_{U\times f}$ both characterized in the same latent factor space.

$$\hat{R}_{U\times M} = U_{U\times f} \cdot (M_{M\times f})^T \qquad (27)$$

In our case we generalize to the domain *D* with *X* dimension so that we end up with the matrix of user-resources, formally by the expression:

$$\hat{R}_{U\times M} = U_{U\times X} \cdot (M_{M\times X})^T \qquad (28)$$

As we have elaborated before how we can build $U_{U\times X}$ as the User-Concept matrix and $M_{U\times X}$ is the Resource-Concept matrix, then the recommendation matrix $\hat{R}_{U\times M}$ can be built. For each value $\hat{r}_{um}$ it is calculated with the expression:

$$\hat{r}_{um} = \sum_{x\epsilon\{1,\ldots,|X|\}} U_{ux}.M_{mx} \qquad (29)$$

Thus if we adopt a decision making vision, this method has a well-known canonical form of the s*imple additive weighting method* (SAW) for multi-attribute decision making [12], which is a common aggregation method in the decision making domain. Many studies in this domain [13], [14], [15] have shown that the intuitiveness of the SAW method makes it more preferable for user direct interaction over other less interpretable, non-linear methods.

Therefore iSoNTRE behaves like a SAW in order to build the overall recommendation matrix. The main advantage of this adaptation is the ability of users to understand and interact with it.

The recommendation matrix has been built, normalized, and the recommendation engine is now ready to recommend; Figure 21 shows an example of the resulting matrix and Figure 22 shows the details of the process discussed in this section.

$r_{1,1}$	$r_{1,2}$	$r_{1,3}$	..	..	$r_{1,m}$
$r_{2,1}$					
$r_{3,1}$					
..					
$r_{u,1}$					$r_{u,m}$

$\hat{R}_{U\times M}$

=

u_1	..	u_x
u_2		
..		
..		
u_u		

$U_{U\times X}$

·

m_1	m_2	..	..	..	m_m
..					
m_x					

$(M_{M\times X})^T$

Figure 21 the recommendation matrix

By building the extracted recommendation matrix $\hat{R}_{U\times M}$, the process of transformation of social network information into a recommendation engine is finalized. iSoNTRE offers the methodology to move from the raw data in these GPSNs into a recommendation matrix, which is the point where other social recommendation systems depending on recommendation social networks begins. At this step, we can move to use any recommendation algorithm and to compare different variants, including CF algorithms.

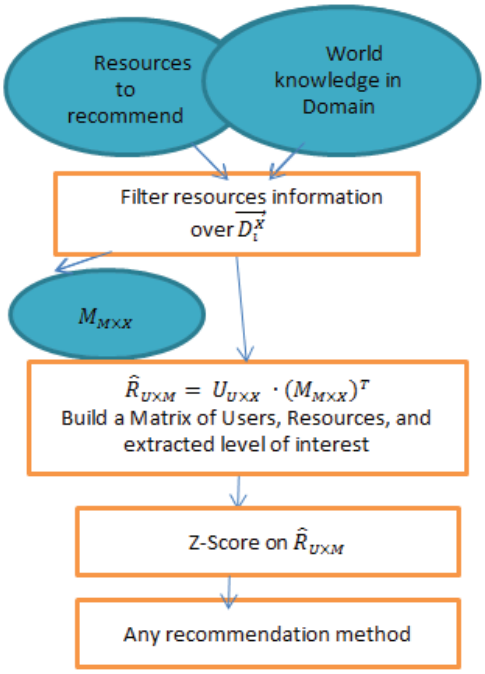


Figure 22 the process to build the recommendation matrix $\widehat{R}_{U\times M}$

Matrix Normalization:

In traditional recommendation systems, ratings values are in the interval [1..5]. In the systems that use implicit data this is not the case; as in the extracted data, an extracted value of any interval might be found [0..N]. This is because the idea is to count how many times the concept has appeared in the profile. For this reason, we need data normalization, meaning that we need to translate all the extracted values into a meaningful interval for recommendation.

In [9], the authors argued that the Z-Sore in similar cases can work very well and indeed sometimes better than other normalization scores. Thus, we adopted the Z-Score in order to do the normalization, where μ is the mean of the population and σ is the standard deviation of the population.

$$Z = \frac{x - \mu}{\sigma} \qquad (30)$$

The absolute value of Z represents the distance between the raw score and the population mean in units of the standard deviation. Z is negative when the raw score is below the mean and positive when it is above, which is a very good point in our case, as it permits us to keep the 0 in the matrix whenever there is no explicit extracted rating as 0 doesn't affect the calculation of the score.

A key point to keep in mind is that calculating z requires the population mean and the population standard deviation, not the sample mean or sample deviation. Normalization can also be done by a simple centering of the matrix values.

3.1.5. iSoNTRE in Action:

The resulting recommendation matrix contains the users, items (resources), and extracted level of interest. In order to recommend, two approaches can be elaborated: the first assumption is that the resulting matrix, as it contains the extracted level of interest of users towards items, is ready to recommend directly—so for each user it is enough to rank his row

and recommend to the resources that are on the top of the list (Top N recommender); the second assumption is that the resulting matrix is a recommendation matrix (user, resource, rating) and then the recommendation will be performed based on any recommendation method such as memory based or model based.

However when iSoNTRE is actively working two methods are needed. The first method will start the system, but when the system is working (when users are responding to the recommended resources and real ratings are being added to the matrix) the second method will be adopted. Both of the methods are tested in the next chapter. After that, we will introduce how to add new users and new items into the system.

New User Recommendation?

Adding a new user to the system is done like in matrix factorization methods: first by finding the vector of the user $\overrightarrow{G_u}$, then by finding the intersection with the related world knowledge $\vec{U}_{1\to R^X}$, and finally by using his vector to find his level of interest in resources. Then he can easily join the overall matrix. The same discussion can be applied on new resource(s).

$$\hat{r}_{um} = \sum_{x\epsilon\{1,\dots,|X|\}} U_{ux}.M_{mx} \qquad (31)$$

Different Recommendation Engines Over Different World Knowledge:

As discussed before, the user/resources information was extracted over the same domain $\overrightarrow{D_i^X}$ in order to get a recommendation matrix $\widehat{R^i}_{U\times M}$ over this domain; in the previous work we have ignored the *i* in for simplicity. This means that for each user there are different profiles over the different domains, and for each domain there are resources that will be recommended.

In other words, this means that the transformation process results in a different recommendation matrix over the different domains -Figure 23-. When we work on one domain, we'll ignore the *i* indices of the domain.

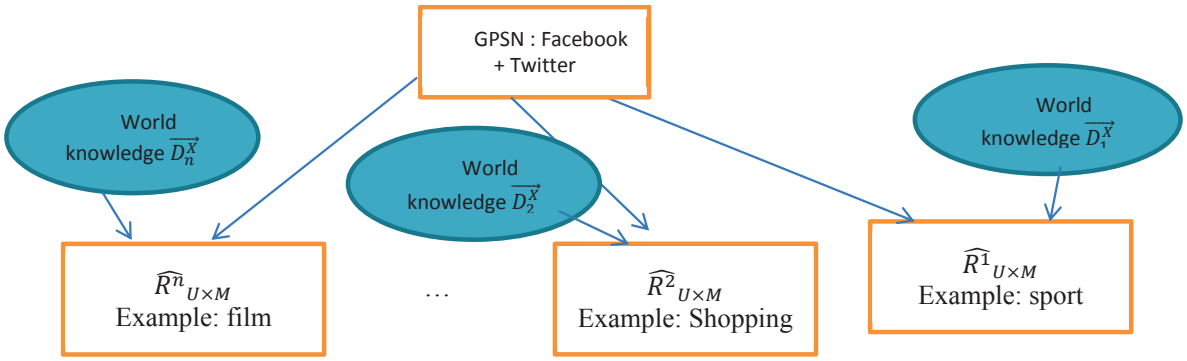


Figure 23 Many recommendation engines over the different domains.

The iSoNTRE method of building the recommendation engine is a combination of matrix factorization (based on the user-concept matrix $U_{U\times X}$, and the resource concept matrix $U_{U\times X}$) the $\vec{X}$ dominion of the domain $\overrightarrow{D_i^X}$ and the resource, user extracted profiles based on a specific domain $\overrightarrow{D_i^X}$. In practice, the iSoNTRE matrix (like the other recommendation

matrixes) should be calculated again within time, such as every day, in order to include the new changes and actions—knowing that the first time will be the most difficult as in the rest of the time it is enough to add only the new actions.

Including the User Feedback:

We outlined that one advantage of the adopted SAW model is that it can add the user feedback to the extracted user profile: by using the *X* latent space of the domain $\overrightarrow{D_t^X}$ the user can edit his profile and update it. In turn, these changes can be reflected in his new recommendations. This can be done both in the beginning of the process or any time during the work based on the desire of the user. The algorithms to include the users' feedback are:

1. The user profile is extracted from the social network $\overrightarrow{U_D}$ over a domain $\overrightarrow{D_t^X}$ and a first ${U^o}_{U\times X}$ is built.
2. The user is shown his profile in a graphic cloud and feedback is collected. The feedback possibly includes an addition of a concept, a suppression of a concept, or the change in the size of the concept which reflects the level of interest in the matrix $E_{U\times X}$. The changes are added to the user main matrix $U_{U\times X} = {U^o}_{U\times X} + E_{U\times X}$
3. A recommendation matrix $\hat{R}_{U\times M}$ is obtained over the corrected user-concept matrix $U_{U\times X}$

$\hat{R}_{U\times M}$ is the matrix of extracted ratings of users *U* towards the items *M*. In the recommendation words, this matrix contains the predicted ratings of users to these resources. To conclude, iSoNTRE uses the SAW method based on the extracted information in order to predict the ratings of users towards resources. This is what was used in the Facebook application when the user was invited to update the extracted commercial profiles, then the updates that he provided us with were captured and added to his profile as discussed in the previous algorithm.

4. iSoNTRE discussion:

In this chapter we introduced iSoNTRE to transform GPSNs into recommendation engines through a recommendation matrix and by using a world knowledge as a middle layer. iSoNTRE adopts a methodology based on the inverse of matrix factorization algorithms and gives users the possibility to update their profiles in order to get better recommendations.

iSoNTRE also offers numerous contributions: it is a hybrid solution that gathers content and collaborative based techniques in one recommender in order to handle the raw data of GPSNs, it uses the data that users have entered with their free desire to help them where they find it difficult, to provide information. This overcomes the cold start problem on new users and items as new items can be recommended right away based on its concepts.

The lay over the raw data of users makes iSoNTRE one recommender that benefits from the advantage of implicit information, like its availability, and that it reflects the real users' internal information, which is not like the explicit ratings that may vary over time (e.g. a user sees a better film) and are sometimes hard to obtain.

Besides, prediction values will be good as they result from a matrix factorization. More than that, the user profile in the initial step of iSoNTRE is extracted from his actions in the GPSN, which reflects the user's personality and needs, giving him the benefit of his spontaneous actions in order to enhance his recommendation.

Also of note, the resources that can be recommended may be any kind of resource, only under the condition that the relevant world knowledge must be found, even with the SLiR resources introduced in the previous chapter.

iSoNTRE's main purpose is to use the chaos of daily human interaction to meet their needs, all without requiring users to enter and re-enter their interests again and again. When iSoNTRE works, it is a living social machine of a future where people do the interesting work and computers do the administration. Next we will show iSoNTRE's evaluation through both offline and online cases.

5. Evalaution

In order to assess a recommender, it is usually evaluated using available data sets over metrics like those discussed in the first chapter; however, iSoNTRE is a hybrid recommender that transforms a GPSN into a recommendation engine, thus it is not possible to evaluate iSoNTRE on available data sets like those of Epinions, Flixster, or movie data sets. To evaluate iSoNTRE, it should be built it first, and then it can be evaluated. In this chapter we will show how to build then evaluate iSoNTRE both in Facebook and Twitter. We used short-life resources (offers) in the evaluation process, as well as we used the commercial domain. As discussed in the previous chapter we build the matrix $\hat{R}_{U\times M}$ (Twitter, Facebook) and evaluate the recommendation resulting from it through two levels:

First, compared to the traditional solutions, using the RMSE and MAE metrics through a cross validation process.

Second, a live test, which we considered critical for our system. It is based on feedback from real Twitter and Facebook users. Its goal was to see the reaction of people towards the recommended items. As the test is based on two different social networks, this meant that we needed to build two recommendation matrixes $\widehat{R}_{U\times M}$, one for Twitter and the other for Facebook.

6. Building iSoNTRE:

We used the Twitter data introduced in the previous chapter (12,000 users who had 1030 commercial concepts) to build iSoNTRE so we had a $U_{12000\times1030}$ matrix. We normalized the matrix with the Z-Score and the world knowledge was the same BKT used in the previous chapter.

We had nearly 10,000 offers from Blue Kangaroo in order to try the methodology. These offers varied over the 1,030 commercial concepts resulting in the $M_{10000\times1030}$ matrix; combining the two matrixes resulted in a complete matrix:

$$\hat{R}_{12000\times10000} = U_{12000\times1030} \cdot (M_{10000\times1030})^T$$

In this matrix, we have 88,446 values of estimated ratings.

On Facebook the matrix was built over the 2000 users and the 538 commercial concepts.

$$\hat{R}_{2000\times10000} = U_{2000\times538} \cdot (M_{10000\times538})^T$$

7. Evaluation of iSoNTRE:

The recommendation matrix $\hat{R}_{12000\times10000}$ and $\hat{R}_{2000\times10000}$, contained the offers that might be interesting to the user based on his extracted commercial concepts from his actions in the social networks, we used the two assumptions to assess them:

The first assumption: the matrix is considered ready to recommend, as it contains the predicted calculated values of how interesting an offer is to a user. According to this process, the matrix is ready to directly recommend offers. The second step in this case is to evaluate to what end the recommended list was interesting to the user. Top N item evaluators as we didn't have the truth ground in order to choose another evaluation metric like RMSE and MAE. Top N can better answer the question: how many recommended items are good and responded to the user needs? The Facebook application designed asked the user about how many offers they liked from the top N recommended offers for them.

The second assumption: the matrix is a rating matrix and found predictions based on it. The claim behind this assumption is based on the live test with Facebook in the previous chapter, when we asked to evaluate their predicted commercial clouds; we found that users added concepts to the cloud but they did not delete much, which meant that the predicted cloud was correct but not totally complete.

The second assumption is essential when iSoNTRE works well because with time people will carry out actions and ratings that will be used to find new predictions and recommendations. This means that in real-life cases the iSoNTRE recommendation matrix will contain both the real ratings and extracted ratings, with the possibility of a user changing his profile information.

7.1. Assumption 1: Matrix Ready to Recommend:

7.1.1. Facebook Live Test:

In this scenario, we found the most interesting offers for every user by ranking his rows in the matrix so that the offers that had the highest values were at the head of the recommendation list (Top N recommendation). Then, the resulting ranked vectors of offers were presented to the user. As the data had been extracted from scratch, we didn't have the truth ground in order to find the RME and RMSE on the predicted data, which is why this test had to be done on real users.

As a list had to be recommended to users, a good indicator in this case was to find from the Top N item how many items are interesting to users. A Facebook application (roo are you) was created for this purpose; -Figure 24-. It adopts a simple principle, which was a like or dislike of a recommended offer. The application permits to each user to classify a collection of offers as interesting, not interesting, or normal within a timeframe of two minutes (the short time encouraged users to try our application and easily complete its process). Results of this application are in Table 13. 143 users tried our application over two trials. The precision (percentage of liked offers to the whole recommended offers) is in Figure 25.

Facebook trail	# offers in Top N list	# users	Mean of Precision (liked of the 60 offers)	MAP
Trial 1	60 offer	74	82%	0.832
Trial 2	60 offer	143	80%	0.811

Table 14 Results of Facebook application over two trials

The high precision and MAP values in Table 14 and in Figure 25 shows that the recommended offers based on the Facebook information were interesting to users.

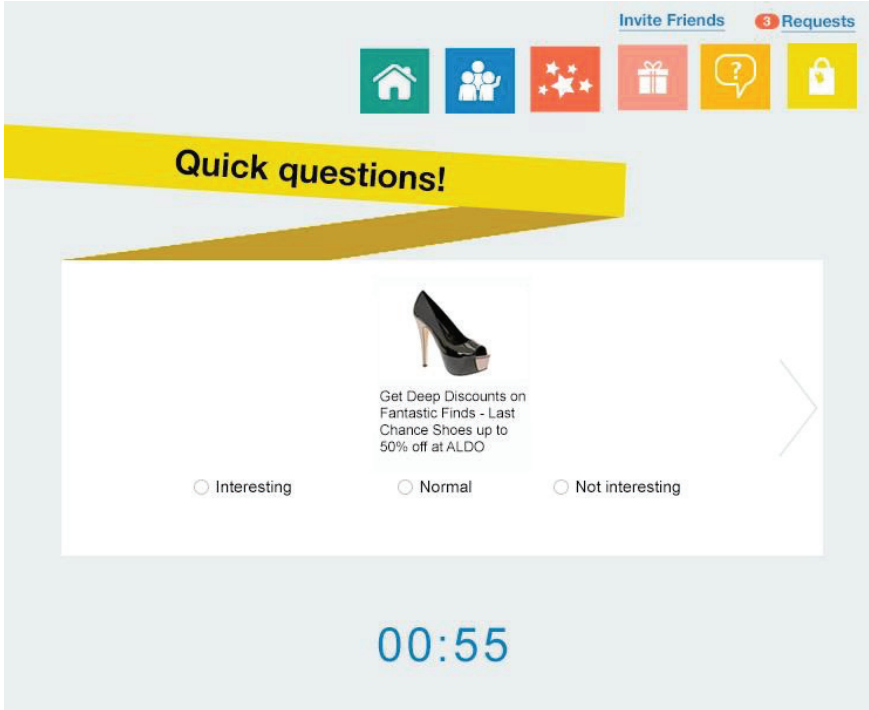


Figure 24 The Facebook application evaluated if the user had found the recommended items interesting or not interesting or if they were normal to him.

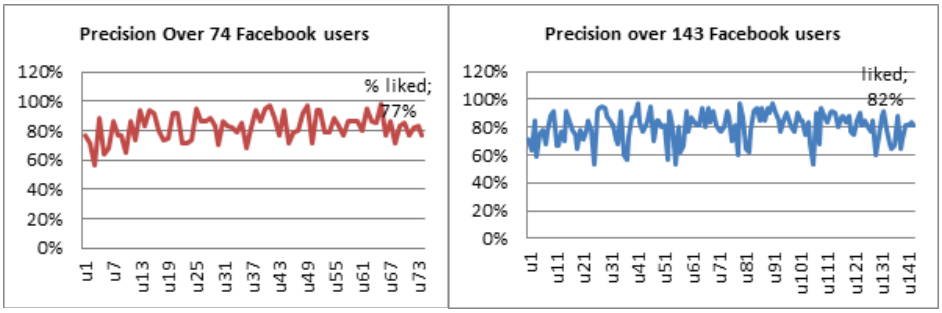


Figure 25 The precision over two trials; each user evaluated 60 recommended offers, precision around 80%!

Results of both applications show that recommendations based on the first assumption gave relatively high results.

7.2. Twitter Live Test :

Like in Facebook we wanted to study the feedback of real users on the recommended offers. The main idea of the test was to send offers to users and see if they would like them or not. “Like an offer” in the Twitter case meant to click on it or to buy it. In this scenario, however, we did not have the information for a bought offer, but the “click on offer” information was provided by a tracking system adopted by the company. Thus, we considered a click on an offer as an indication of a good recommendation, like in [23] where the authors considered the click on a song as proof that the user liked the song.

To begin we scanned the database of our 12,000 users, chose 2400 candidates and sent them offers. Since we operated with Twitter, the dedicated user offers were sent through tweets.

Because of the limits of a tweet and because we could not send a lot of links in one tweet, we had to send the best offer, which was the first offer from the recommended list, to the user. The body of the tweet held both the summary of the offer and a link to the offer, as well as a mention of the user using a handle where we could reach him, like in Figure 26. In order to give meaning to the tweet, we added a prefix before the link to the offer, followed by the title of the offer -Figure 26-, a list of examples of possible prefix is in –Figure 29-

In addition to using the click of a user as a feedback, other user actions were assigned as an indicator for positive preference toward the offer, such as a re-tweet, sending a related tweet or a replay or any other feedback.

Figure 26 An example of a tweet targeting a user, the prefix is "Safety first, Savings next", followed by the offer.

So far we have elaborated how we sent the offers and target the users, the next portion to address is who sent the tweets.

7.3. Sirens Creation:

In order to be able to send messages or tweets to targeted users, we had to create twitter accounts that we called Sirens, accounts which are supposed to behave as human: they "work" during the day and "sleep" through the night, which is normal human behavior. In addition, Sirens "worked" more during the weekends when they had more free time and did not "work" Wednesday or Thursday based on the same study in order to imitate normal human tweeting behavior [20]. With this behavior established, these accounts were meant to send tweets that contained the recommended offer that we wanted a user to receive so we could collect the feedback -Figure 26-. Because of the limit regarding the number of tweets that could be sent daily, we created 80 Sirens, each one tasked to send offers to a group of our targeted users -Figure 27-.

[20] http://www.marketingpilgrim.com/2013/04/how-to-maximize-your-tweets-infographic-no-joke.html

Figure 27 An example of a Siren.

The sirens held American names since our 12,000 users were based in the United States; they also lived in different states therein, had a family, and displayed different personalities and interests -Figure 28-. An interest in our case was translated into tweeting in the domain; for example, a Siren interested in funny content would follow, tweet, and re-tweet funny content. In order to do that we defined for each domain parent accounts, which are accounts from Twitter interested in each domain. For privacy reasons, we avoid mention of the parent accounts in this study, however they are available based on a mail request to the authors and an example is in -Figure 30-.

SOCIAL TOOL Home twitter facebook Contact Login

Home / Twitter / Avatar Peronality

Avatar	Sports	Women	Dancing	Food	Music	Shopping	Travel	Diet	Fan of Life	Writing	Fan of Movies	Optimist	Intelligent	Pets	Funny	Romantic	christmasfan
ChTaylor80		1	1		1												1
ShoppingMomAR		1			1	1							1				
MamaEvaShops			1			1					1						1
Dealz4Kidz		1				1											
iFindDeals2Shop				1						1							

Figure 28 An example of Siren personalities.

SOCIAL TOOL Home twitter facebook Contact Login

Prefix ADD

Prefix
Here's a holiday keeper 4u:
Pick out your holiday deal
Special gift ideas
Oh you are a lucky woman with:
How goes the shopping?
Your Top deal:
What a deal:
STOP!!!
so clever. Must try:
Deal of the day:
VIP deal ;)
Christmas deal :)
VIP Christmas deal ;)
be happy with:
Snow super deal:
your new favorite deal:
Great deal on:
You can save with:

Figure 29 Some of the prefixes used by Sirens in the period of December

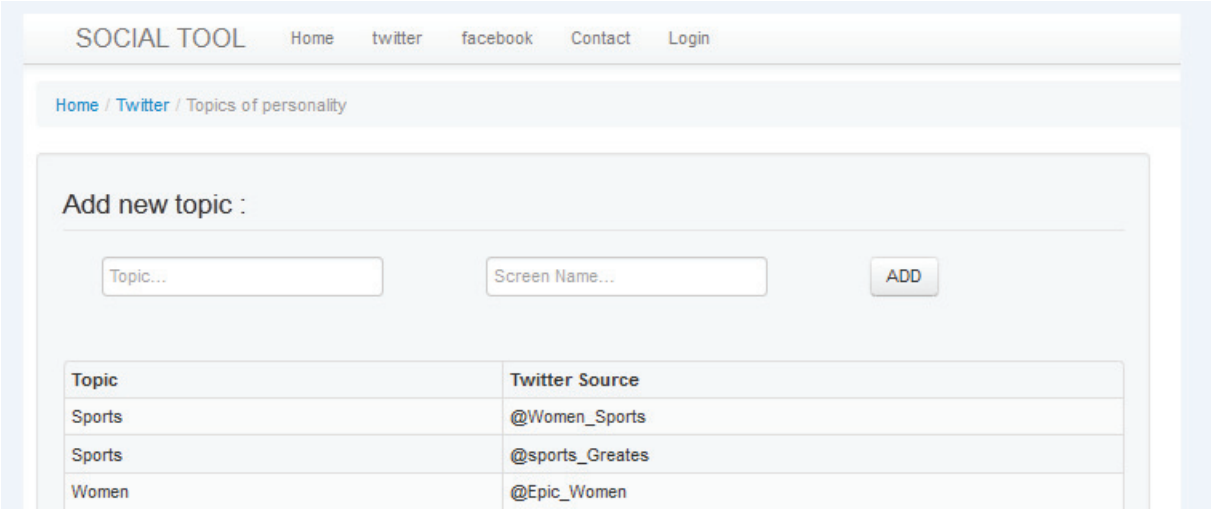


Figure 30 An example of parent accounts.

7.3.1. iSoNTRE in Twitter (with Sirens)

Two trials were elaborated over the Sirens. In the first one 2,000 users were targeted over a period of six days in October 2013, while in the second trial 2,400 users were targeted over a period of five days in December 2013. Each day, 40 Sirens sent out general tweets related to their personalities, (20 general tweets) and each of them targeted ten users. Before the trials, each account tweeted nearly a week's worth of general tweets without any mention to establish legitimacy. -Table 15- summarizes the trials setups:

Twitter users	**targeted users**	**# Sirens**	**#Target Tweet /Day**	**# Days**	**General tweets related to personality**
Trial 1 (December)	2400	40	10	6	20
Trial 2 (October)	2000	40	10	5	20

Table 15 the Sirens overall information in the two trials

Table 16 contains the results of the targeting process where Sirens received different actions: clicks on the offers, thanks for tweets, positive interactive tweets, and followings. There were also some negative tweets from people who did not like the offer. The most surprising result was the fact that our Sirens received offers from other Twitter accounts which were not ours. Other Sirens were already around during the study!

	Click	**%**	**Thanks**	**Followings**	**Normal tweets**	**Negative tweets**	**Received offers**
Trial 1 (December)	294	14,7%	16	62	18	12	9
Trial 2 (October)	450	18,7%	11	78	11	16	15

Table 16 an average of 16.7% of user click on the proposed offers.

iSoNTRE was able to get a click-through rate of 17,3% on the recommended offers. These users saw the offer description in the tweet and then clicked on it, which is a high indication of user satisfaction compared to the general click-through rate on Twitter of 2.8% [27].

The fact that the Sirens received a lot of actions (the re-tweets, normal tweets, followings and thanks) indicates that the Sirens were able to behave as humans. Of note, we witnessed a

difference of clicks between the two trials, we posit that this difference might be due to the timing of the two trails, as in December people are preparing for the Christmas and holiday season and are more open to receiving offers for items that they plan on purchasing and that come with competitive and attractive offers.

7.4. Assumption 2: Matrix to Build Recommender System on It:

This assumption is based on the result of our Facebook live test introduced in the previous chapter, where we remarked that users have the tendency to add brands and categories to their commercial profile extracted from a social network, but not to delete concepts from them, meaning the extracted commercial profile in most cases was relatively right but not complete. Based on this assumption, we consider that the extracted rating matrix is a core of the system and we should provide recommendations based on it.

At the same time, this assumption is important due to the fact that iSoNTRE includes users' ratings in addition to the extracted ratings, meaning that the system needs to use a recommendation method to predict and recommend items.

As a first step we considered all the 0's in the matrix as non-determined values, or ratings that needed to be predicted using a recommendation method. We compared results varied over SVD, a user-user collaborative filtering, and an item-item collaborative filtering algorithms. For each of which we compared to the data itself, on the data normalized by centering the values and by the Z-Score. The values are in -Table 17-. We noticed that the values in the SVD recommender usually gave the best recommendation results, and the best among all was the SVD with the Z-Score normalization.

Recommendation method	**MAE**	**RMSE**
User based CF	4.82	30.26
SVD	6.26	24.56
User based CF (normalized center)	5.08	16.58
SVD (normalized center)	3.95	12.38
User Based CF (Z score)	1.02	1.82
Item based CF (Z score)	4.18	4.28
SVD (Z score)	***0.62***	***0.99***

Table 17 A comparison between different recommendation methods; SVD with Z-Score shows the best results

In addition, based on the previous table we found the SVD with Z-Score to give the better results in the MAE and RMSE metrics. In SVD, the choice of the number of categories affected its performance, which is why we tried to find the better value for the number of categories in -Table 18-.

SVD With Z-score	**MAE**	**RMSE**
10 categories	6.26	24.56
20 categories	3.95	12.37
50 categories et plus	***0.62***	***0.99***

Table 18 A comparison between different category values in SVD with Z-Score; the 50 value is the best choice over this data, as more trails showed as the same MAE, RMSE values.

System in Action

In the previous discussion we built the rating matrix from social networks in the shopping domain. When the system works, the first step is to begin with assumption 1 and then to move to assumption 2. This means that the system has its first round based on offers predicted to be good, and then it will continue working based on assumption 2 as in any recommendation system. However, from time to time a synchronization process has to be done between new social network information and recommendation systems.

8. Conclusion:

In designing iSoNTRE, we were inspired by the social machines of the future that Berners-Lee mentioned in his book Weaving The Web. The evaluation of iSoNTRE shows that it is an effective methodology to transform social networks into a recommendation engine, as it was able to recommend short like resources items (SLiRs, or offers) to users effectively. At the same time, it recommended many other kinds of items. We discussed how iSoNTRE can work directly after building its matrix, or it can adopt any recommendation method that can be applied over the resulting matrix. We compared different methods and normalization methods and found that Z score normalization with SVD gave the better results.

In the proposed evaluation we used the BKT world knowledge, but any other world knowledge could be used as well. iSoNTRE can be applied over different domains, such as films or books; however, the choice of a world knowledge is critical to effectively running the system. Using the ODP of Wikipedia, for example, permits one to build a complete system produced by users at the level of world knowledge and actions.

Like any recommendation system, iSoNTRE needs to be updated from time to time in order to include users' latest actions, both the actions at the level of social networks and within iSoNTRE itself.

9. References:

[1] Denti, Leif, Isak Barbopuolos, Ida Nilsson, Linda Holmberg, Magdalena Thulin, Malin Wendeblad, Lisa Andén, and Emelie Davidsson (2012), « Sweden's Largest Facebook Study ». *Gothenburg Research Institute, available at: http://gupea.ub.gu.se/bitstream/2077/28893/1/ gupea_2077_28893_1.pdf),* May 24, 2013.

[2] M. D. Ekstrand, F. M. Harper, M. C. Willemsen, et J. A. Konstan, « User perception of differences in recommender algorithms », in Proceedings *of the 8th ACM Conference on Recommender systems,* 2014, p. 161–168.

[3] Tim Berners-Lee. « Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web by its Inventor » . *Harper San Francisco,* 1999.

[4] J. Grimmelmann, « Facebook and the social dynamics of privacy », *Iowa Law Rev., vol. 95, no 4*, 2009.

[5] M. D. Back, J. M. Stopfer, S. Vazire, S. Gaddis, S. C. Schmukle, B. Egloff, et S. D. Gosling, « Facebook profiles reflect actual personality, not self-idealization », *Psychol. Sci., vol. 21, no 3, p. 372–374,* 2010.

[6] Y. Bachrach, M. Kosinski, T. Graepel, P. Kohli, et D. Stillwell, « Personality and Patterns of Facebook Usage », in: · *Proceedings of the 4th Annual ACM Web Science Conference, WebSci '12, Pages 24-32 ACM New York, NY, USA* 2012

[7] A. Java, X. Song, T. Finin, et B. Tseng, « Why We Twitter: Understanding Microblogging Usage and Communities », in Proceedings *of the 9th WebKDD and 1st SNA-KDD 2007 Workshop on Web Mining and Social Network Analysis, New York, NY, USA,* 2007, p. 56–65.

[8] A. Beccera, S. Jimenez, A. Gelbukh, « towards user profile based interfaces for exploration on large collections of items» in *Proceddings of Rec Sys workshop, Hong Kong, Chine,* 2013.

[9] X. Su et T. M. Khoshgoftaar, « A survey of collaborative filtering techniques », *Adv. Artif. Intell., vol.* 2009, p. 4.

[10] X. Yang, Y. Guo, Y. Liu, et H. Steck, « A survey of collaborative filtering based social recommender systems », *Comput. Commun., vol. 41, p. 1–10,* 2014.

[11] J. Tang, H. Gao, H. Liu and A. Das Sarma. “eTrust: Understanding Trust Evolution in an Online World”, Proceedings *of the Eighteenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* , 2012.

[12] G.-H. Tzeng et J.-J. Huang, "Multiple attribute decision making: methods and applications". *CRC Press,* 2011.

[13] G. Adomavicius, N. Manouselis, et Y. Kwon, « Multi-criteria recommender systems », in *Recommender system handbook*, Springer, 2011, p. 769–803.

[14] C.-H. Yeh, « A Problem-based Selection of Multi-attribute Decision-making Methods », *Int. Trans. Oper. Res.*, vol. 9, no 2, p. 169–181, 2002.

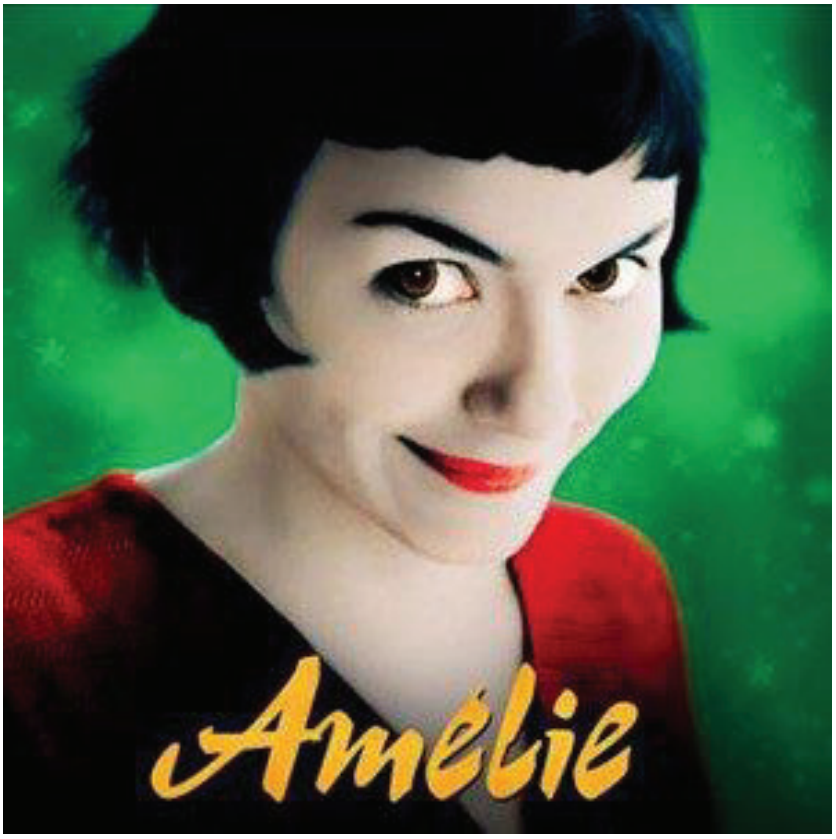
[15] C. Zopounidis et M. Doumpos, « Multicriteria classification and sorting methods: A literature review », *Eur. J. Oper. Res.*, vol. 138, no 2, p. 229–246, 2002.

CHAPTER 6: IAMÉLIE

-A Hybrid Recommender Based On A Reflection In Recommendation Systems -

« Savez-vous quel est le point commun entre tous ces personnages ?
C'est Amélie, et elle va changer leurs vies.
Et si elle changait votre vie !!! »

Le fabuleux destin d'Amélie Poulain, réalisé par Jean Pierre Jeunet (2005)



https://www.youtube.com/watch?v=N0rnLZN5r6w

1. Introduction

Before the introduction of modern computer and search technologies librarians were the masters of surprise and suggestion. With expert eyes and an expansive memory, they built strong relationships with their most frequent visitors and recommended books that corresponded to each one of their readers' needs, interests, and personalities. But most importantly librarians also explained their recommendations to these curious consumers of stories and text. As a result, the librarian would recommend the same book to different readers, each having different tastes, and any number of these different readers might enjoy the same book for their own reasons: some wishing to be the first of their friends to have read it, others following the tastes and tests of others in order to read a something that they were sure would be good, while yet another reader picked the book up because she was familiar with and liked the book's subject despite the social aspects surrounding it.

For the attentive librarian of yesterday and today, the process was and is part of a respected profession. Today, however, most of us are not going to the library for our daily recommendations; instead, we browse the web for nearly everything we need. And where a person was once our trusted recommender, instead algorithms have come into being as the invisible actors of the online world, albeit with some limitations and risks.

To understand these limitations and risks, one must first understand a core concept of recommendation: that it is a long process that changes the recommendee, and even their community as well. This is due to the slight change in humans as they consume items (books, films, receipts, etc.) and their preferences change, become stronger, and ultimately become fixed. Although it is a *slight* change, it is one that can last indefinitely.

With this in mind, it is easier to understand the possible negative outcomes of the algorithms that drive the web's common collaborative filtering recommenders. As they currently operate, recommenders may introduce risk to individuals and communities over the long term: "If users are choosing items to consume based on personalized recommendation, when time passes they will end up in groups, or clusters of likeminded individuals. This divides the community into groups who seldom interact with people with whom they do not agree." [1]

In this sense, despite their service, social recommenders might be worse than they appear! They may give better recommendation based on evaluation metrics and short-term user satisfaction, but over the long term they divide the community into small groups of people who already know each other (usually a group of friends). Taken further, if a circle-based recommendation were adopted in social networks it would divide the group of friends into even smaller groups of overly like-minded people who continue to follow each other in the consumption process.

In this chapter, we will discuss an approach deals with these recommendation issues and propose a methodology of how to build a sensible adaptive recommender, which we have named iAmélie, a hybrid recommender that can be applied on iSoNTRE or on any social recommender.

The main idea behind iAmélie is to use the social context of users in order to enhance the way in which recommendation is done. iAmélie changes the recommendation from user to user, or even for the same user from one context to a different one.

2. Why iAmélie:

As the story goes, Amélie Poulain[21] was a little girl who was raised secluded because of supposed a heart illness (which she in fact did not have). With no schooling or a social life, she created her own world from which she watched others. Growing up, she wanted to positively change people's lives, and as she watched them and became familiar with them, she succeeded in making them happy. By watching people closely and understanding them she was able to surprise them, transforming their lives into something much better than before.

With this story as our inspiration, we knew what we wanted our iAmélie to do: like the original Amélie, the electronic one aims to observe and understand users in order to provide recommendations that will respond to their needs and enhance their lives.

iAmélie is a safe system that uses users' *available* social information to adapt its recommendations for them. It does this in a transparent and interactive way, provides explanations, corrects itself, and ultimately helps users save time and enhance their day-to-day life. Viewed as a whole, iAmélie has the intelligence of the librarian, and the magical, attractive spirit of Amélie Poulain.

While not a new recommendation platform, iAmélie is a hybrid adaptive recommendation platform that gathers different recommendation methods (which can be any recommendation method) into one platform.

3. iAmélie System:

The iAmélie system's main contribution is that it makes a step backwards in the domain of recommender systems, then based on the previous reflection over these system assumptions and the needs that they try to respond to.

In order to explain how iAmélie works, we will classify the main recommendation strategies into four families, which are based on recommendation systems history and the works introduced in the previous chapters. The goal of this classification is to discuss the claims and assumptions that are behind them which will be the basic of iAmélie system.

3.1. Classical CF algorithms:

This system recommends based on the overall users in the system. Algorithms—either model-based ones or memory based—find similar active users or items to the target one (practically around 25 similar users or items give good values [1]), then they recommend based on these similar users or items. The main assumption behind all works in this branch of recommenders is that:

Assumption 1: If two users are similar in their ratings for some items, then they are more likely to be similar in their ratings towards other items.

3.2. Domain/Cluster-based algorithms:

In this branch of recommenders, the recommendation matrix is divided into a smaller, compact matrix (less sparsity). Recommendation is done by finding the similar users

[21] http://en.wikipedia.org/wiki/Am%C3%A9lie

over the smaller matrix [2]. Clustering is one technique within this kind of family. The main assumption in this branch is that:

Assumption 2: Finding the similar users for the target user might be more meaningful in a smaller group of similar users.

Some works find many clusters for the user or groups and give predictions to the user based on the different clusters, then combines the results in one recommended list [3]. That is why recommendation is done over a smaller matrix. The main assumption of this family of recommenders is

Assumption 3: The ratings that the user provided in one domain (like film) might be similar to other users, but he might be similar to other groups of users in another domain (like clothes).

It is important to note here that even in collaborative filtering systems, such as those used on a website like Amazon, categorization plays a major role in the whole system, even though it is called a CF system.

In both of these recommender families, online social friends of the user are not integrated, as at the time when these families debuted social networks had not yet reached their current high level of usage.

3.3. Social recommenders or community recommenders:

In this family of recommenders the recommendation is based on the surroundings of the user in a social network, most often the user's relations within the social networks. The similarity (sometimes referred to as "trust") is calculated upon the users' friends so that the recommended items are those that are also consumed by the user's friends [4]. The main assumption in this family of recommenders is

Assumption 4: The user trusts his friends, and he is usually similar to his friends, so he will be more likely to consume items that have already been consumed by them.

3.4. Circle-based recommendation:

Circle-based recommendation divides the user's own social network into different groups according to shared activities, then tries to recommend items from the appropriate group. A good example of this is found in [5]. This division is claimed to decrease the noise of friends whom the user doesn't trust in each domain. The main assumption here ought to be clear

Assumption 5: The user is similar to, or trusts, different groups of friends in the different domains in day-to-day life. For example, one might trust Jack and Jane in film recommendation, while for travel one might prefer recommendation based on Lucie and John.

The social- and circle-based recommenders are special kinds of classical- and domain-based recommenders that are applied over users' social networks and have increased in use in alongside the rise of online social networks.

All four kinds of recommenders' strategies have been investigated and have been found to enhance accuracy in the evaluated cases. The evaluations have been made in most cases as recommendation systems are evaluated in the literature: on offline available recommendation

data sets. Figure 31 shows the social graph of user u, or his relations (friends) f extracted from social networks. Figure 32 shows the user, his social graph in the community of all the users, referred to by u.

Figure 33 shows in one graph the four recommendation strategies discussed before based on the previous two figures.

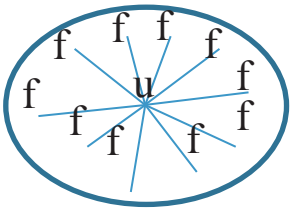


Figure 31 the user u social graph, f refers to his friends.

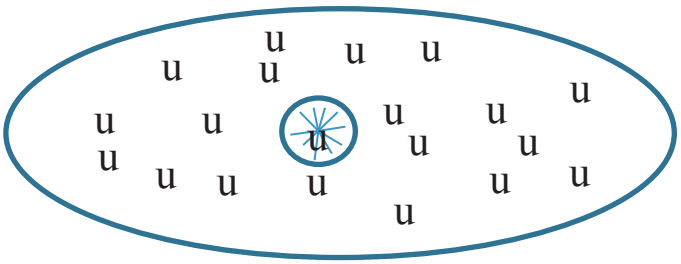


Figure 32 The user and his social graph in the community of all users.

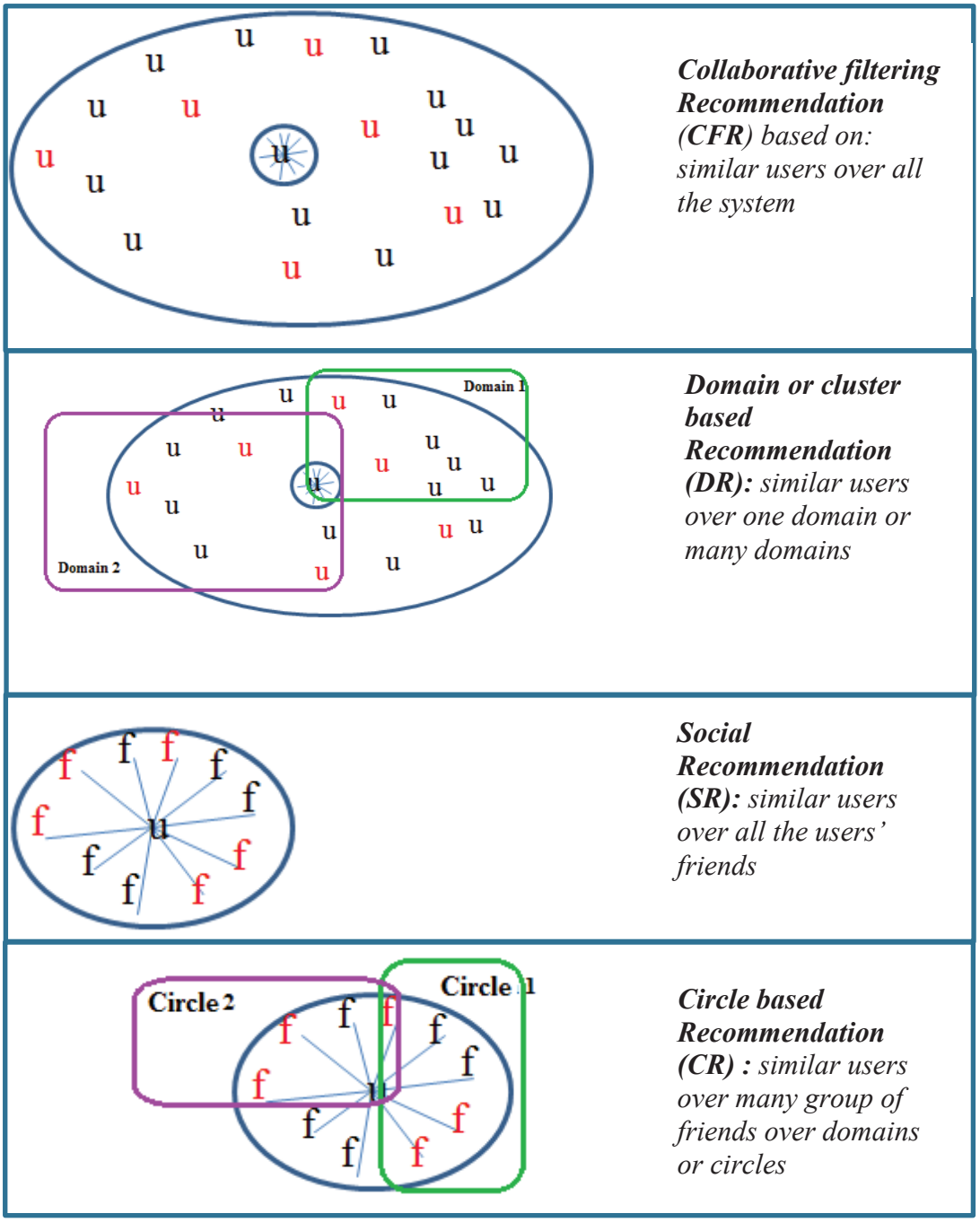


Figure 33 the four recommendation strategies

3.5. Discussion of recommendation assumptions:

We put in question the five previous assumptions and strategies in order to conclude that all of them are right. However, none represent or address the whole reality. In order to do that, we will consider a scenario that happens to most of us in our daily life:

"Alice has a clothes taste similar to Camellia; she can trust her in buying new clothes items.

At the same time, Alice likes to refer to her best friends Jane and Janet when buying clothes. They do not wear clothes that are similar to her style, but they know her tastes well, her form and what clothes she might like.

Camellia, however, likes to be ahead of her friends, she is the mode adopter, and she likes to be the first of her friends to adopt a new robe, shirt or haircut.

Poor Camellia and Alice know nothing about technology. In this domain they totally trust their friends Jack and Robert the most.

Alice likes to see films in the WE vacation, who better than Jack the film lover can advise her? However, Jack refers himself to the whole film-lover community all over the world in order to get film news and he really enjoys the role of being ahead amongst his friends.

Robert, the adventurer of the group, can't live without exposing new things to his group: new sports, new cameras, or new cars. For him, none of his friends can offer help; instead he is the reference for his friends for all new things and strange ideas.

Alice will travel into Paris next week; she refers to Sarah, who lives in Paris, to get information about where to stay and where to eat. As well, she looks into the recommendation sites like Trip Advisor for other people's experiences that can be useful to her."

This kind of complexity and complication is what we face each day of our lives, and it is repeated for each person on the planet. What existing recommender can respond to this group's contradicting needs, let alone all the needs of persons online?

In the example, all of the previous five assumptions appeared, in addition to one more assumption. Table 19 contains an analysis of the cases and the assumptions.

Case	**Assumption**
Alice has a clothes taste similar to Camellia	Assumption 4: Some of my friends are like me, similar to me, I trust them
Alice refers to Jane and Janet for advice in clothes although they are different from her	Assumption 6: Some of my friends might not be like me but they know me well.
Camellia likes to be ahead of her friends	Assumption 2, 3: What do similar people (not friends) have in one domain?
Camellia and Alice trust their friends Jack and Robert for technology advice	Assumption 5: What do my trusted friends in this domain have?
Robert likes to show new things to his group	Assumption 1: What people similar to me all over the world suggest for me to do?
Alice refers to Jack in order to choose a film in the WE.	Assumption 5: What do my similar friends have in this domain?
Jack refers to film lovers all over the world in order to get film news	Assumption 2, 3: What do similar people (not friends) have in one domain?
Alice will travel into Paris so she refers to Sarah, who lives in Paris, for information	Assumption 6: Some of my friends might not be like me (not live near me) but they know me well
Alice will refer to recommendation sites for travel	Assumption 2, 3: What do similar people (not friends) have in one domain?

Table 19 An analysis of the recommendation needs and the assumptions

We notice the appearance of a new assumption, which to our knowledge, has not been treated in the recommendation domain:

Assumption 6: Some of my friends might not be like me, but they know me well.

A recommender that responds to only one of the previous **assumptions** might satisfy a large number of users in many cases, but it risks resulting in the long-term undesirable consequence that we elaborated on in the introduction. This explains why none of the assumptions discussed before is enough to address the changing day-to-day human needs.

In the literature, the work of [6] has investigated the social aspect of recommendation in the sense of offering some intelligence through social recommendation in order to capture the cases in which the user might be affected by some other users' items (he likes/dislikes all what they did). The work proposes a learning recommender that can adjust its recommendation with time based on the users' information from social networks. Their work was tested on Facebook and it has enhanced link recommendation on Facebook remarkably over time[7].

However, their work did not take into consideration the context of the user and his/her individual case. It is a closed system that can enhance recommendation of shared links on Facebook but it cannot be used for other systems; in addition, the proposed system does not consider the assumptions and the effects as we did in the previous section. To resume, their work addresses the social and circle aspects of recommendation. To our knowledge no other solutions take into consideration this idea.

The need in recommenders now is to have a recommender that can adapt easily from one case to another whenever needed, which has yet to be addressed in the recommendation literature [8][9]. In this regard, iAmélie is one of the first hybrid social recommenders that addresses these different assumptions in one hybrid recommender system.

4. iAmélie Recommender:

iAmélie is an attempt to have one recommender responding to the different recommendation assumptions previously discussed. This is why it is a hybrid recommender; it brings together the different recommendation systems we discussed earlier: collaborative filtering, domain-based recommendation, social recommendation and circle-based recommendation—all in one. iAmélie utilizes users' contextual information, such as the social and temporal contextual information, to build its components. These four recommendation components, then, form the base of iAmélie and will be discussed in the next section, followed by a discussion of how the overall system works.

4.1. iAmélie components:

As mentioned above, iAmélie contains four recommenders in order to address different assumptions. The below components are implementations of the four recommendation methodologies that we introduced in Section 3:

4.1.1. iAmélie collaborative filtering recommender (CFR):

This component is the traditional collaborative filtering recommender. It offers the recommendations based on similar users in the whole community. and its goal is to build the recommendation matrix based on the whole branch of users so that similar

users will be unfamiliar with one another personally, but share similar actions, interests, and inspirations [1], which are shared in public over all domains. It can use any traditional collaborative filtering algorithm, model-based or memory-based, depending on the domain in which it is used. Finally, this component addresses the first assumption of recommendation.

4.1.2. iAmélie domain-based recommender (DR):

This component is based on dividing the whole recommendation matrix into different domains or categories. The user can belong to one or more groups of like-minded people in the different categories or domains who are not necessarily his friends but share his interests. One way to achieve this goal is by adopting the work of [3] of finding many clusters for the same user. This component addresses our second and third recommendation assumptions.

4.1.3. iAmélie social recommender (SR):

This component is the global social recommender where the recommendation is based on all the user's friends. Similarity will be applied over the friends of the target users[10], As is the case in most of the social recommenders. Recommendation is done over the user's social graph. The fourth assumption is addressed by this component.

4.1.4. iAmélie circle-based recommender (CR):

This component of iAmélie provides a way to find the user's circles of friends that are similar to him/her over the different domains. It adopts the methodology in [5] to find the user's domains as well as the friends who share the interest in each domain. The fifth assumption is answered by this component.

4.1.5. The sixth assumption:

None of the previous components of iAmélie address the sixth assumption where "*my friends might not be similar to me, but they know me well"*. However we see the answer of this assumption is a service provided by iAmélie, which we call: **A Friend's Advice.**

A Friend's Advice: *is a service that provides the user with the possibility to send a request to all of his/her friends and they can answer the request based on their knowledge about the user, even if they are different from him/her. This is similar to real life cases when we refer to friends to ask for information about an item. iAmélie ranks results based on the level of similarity or trust of a user towards other users.*

The social context is essential in building the DR, SR, and CR components, as this information is used to find the users who might be considered as neighbors of the main user. That is why we consider the system that we are building a social contextual recommender. As in Figure 34. [9]

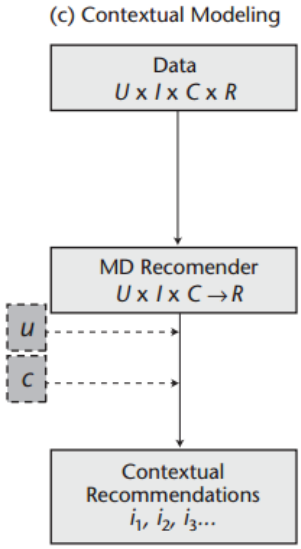


Figure 34 The contextual modeling of recommender systems.

4.2. The recommendation process:

We highlight again that iAmélie needs to have social contextual information about users in order to build its components. There are two ways to build this information. In the first case, iAmélie is built from scratch and the system needs time in order to reach all of its capacities. Without enough time, iAmélie will not have enough actions and relations information about users in order to build its components. This case can be overcome if iAmélie is built in the same way in which iSoNTRE has been built: by establishing recommendation systems from users' implicit information available on online social networks like Facebook and Twitter. However, as the discussion will be similar to iSoNTRE, this case will not be discussed in this chapter; instead, we will discuss the case in which iAmélie is built over working systems like Epinions, Flixster or any other social recommendation system. In this case the social and categorical information is already there and iAmélie can enhance the quality of recommendation.

When the components are ready the next question is: how do we provide the user with a recommendation? If we revisit the recommendation chapter and hybrid solutions, we note that the number of different recommendation systems allow us many possibilities to link recommenders together, such as: weighted recommenders, switching recommenders, mixed recommenders, feature-combining recommenders, cascading recommenders, feature-augmenting recommenders, and meta-level recommenders. For our purposes we adopted the mixed recommenders as we wanted recommendations from the four recommenders at the same time [11] in order to permit the user to choose the corresponding case. With this, iAmélie puts the user into the core of the system by giving him/her the different recommendation methods in a simple interactive way.

4.2.1. iAmélie recommender:

iAmélie builds the four recommendation methods then proposes its recommendation in the four components to users who complete actions in contexts, then both actions and context are integrated into the system to enhance the next recommendation. The actions are integrated, like any recommendation system, by adaptation of the recommendation matrix from time to time (each week) while the context affects the choice of neighbors in the components. Figure 35 shows the components of iAmélie.

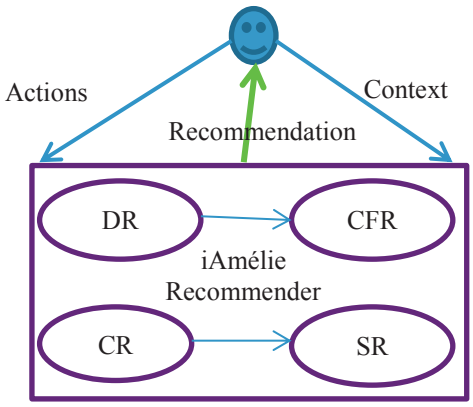


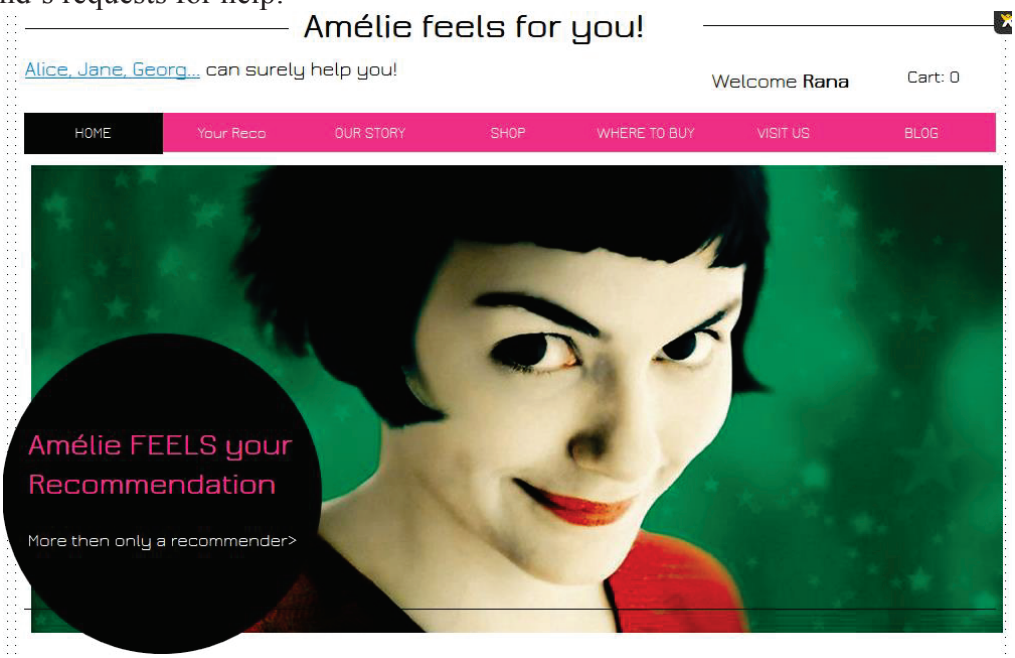
Figure 35 iAmélie recommender system it gathers the four recommenders in order to provide users with recommendations that are adaptable based on their context and actions.

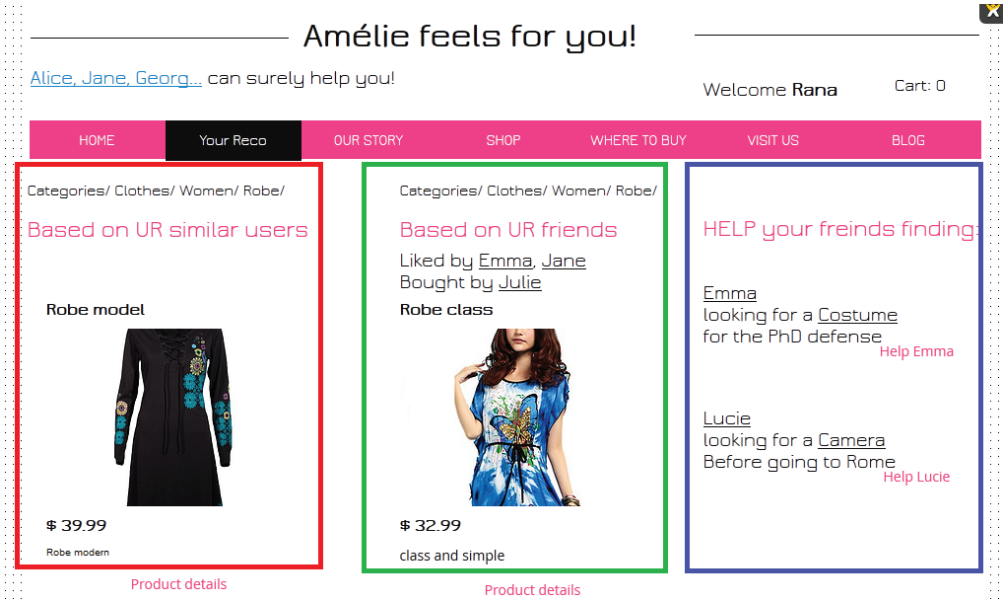
4.2.2. iAmélie in action:

For a user, as introduced before, we consider the DR as a special case of CFR on specific domains; as well, we consider the CR as a special case of SR on special domains. This vision helps us simplify to the proposed model to the user as seen in the screen shot in Figure 36. The user has two lists, one based on similar users over all categories or over one specific category, and the other based on his friends with whom he/she shares his interests, either over all the categories or over a specific category.

The help me friend also appears in Figure 36 as a way a user can ask for help while looking for a specific product (assumption 6).

In the second screen shot is an example of a user's recommendations as well as friend's requests for help.





(a) ***(b)*** ***(c)***

Figure 36 Screen shots from iAmélie system.

(a) Presents the CFR and the DR, (b) presents the SR and the CR, and (c) represents the help me friend, which responds to assumption 6.

4.3. iAmélie evaluations:

4.3.1. iAmélie Building:

We chose Epinions to evaluate iAmélie because we wanted a rich data set that permitted us to directly build the four components of iAmélie. In addition, we wanted to try a different scenario from using Facebook and Twitter as we did in iSoNTRE. The data set that we chose consisted of 127,711 users and 331,274 products, all the information about this data set is shown in Table 20.

Data Set information	
27	categories
331,274	products
127,711	users
1,199,632	ratings
582,613	trusts

Table 20 Epinions data set used to evaluate iAmélie.

The first step in evaluating iAmélie was to build the four components previously discussed; we compared, the User-User and SVD methods on the four components in order to better understand our data set. In order to build the four recommenders we used R and its recommender library. In To establish the DR, SR, and CR we changed the library to control the similarity function and the choice of similar users.

- **iAmélie CFR:** we built the recommender based on the entire amount of information, taking in consideration users with more than 20 ratings and products with more than 20 ratings; the results are in Table 21

	RMSE	MAE
UBCF	1.301	1.039
SVD	3.957	3.750
UBCF (normalized center)	1.310	1.050
SVD (normalized center)	1.183	0.953
UBCF (Z-score)	1.086	0.867
SVD (Z-score)	**0.99**	**0.808**

Table 21 RMSE and MAE over User-User, and SVD for users and products with more than 20 ratings over normalized and non-normalized data. SVD with Z score are the best choice to decrease the error.

We then applied this to users who had more than six ratings and products with more than three ratings. The results are also in Table 22. As the results did not change drastically, we excluded only those people with a low number of ratings from the system.

	RMSE	MAE
UBCF	1.456	1.125
SVD	4.139	3.953
UBCF(normalized center)	1.507	1.197
SVD (normalized center)	1.179	0.928
UBCF (Z-score)	1.189	0.981
SVD (Z-score)	**0.980**	**0.785**

Table 22 RMSE and MAE over User-User, and SVD for users with more than 6 ratings and products with more than 3 ratings, over normalized and non-normalized data. SVD with Z score are the best choice to decrease the error.

- **iAmélie DR:** We defined the matrix over different domains, from Table 23 we chose the following ones (4, 19, 10). The results are in Table 24.

Domain	**# Ratings**
3	167 263
4	**102 957**
5	85 419
19	**85 113**
10	**61 503**
18	55 087
11	48 112
2	45 730
7	45 459
8	45 113

Table 23 The number of ratings by domain

As shown in Table 24, the accuracy over domains is better than over the entire system. This matches results in the previous papers [12] and can be explained by the fact that over a smaller, more compact matrix the prediction is more accurate.

	RMSE			MAE		
	Cat 19	**Cat 4**	**Cat 10**	**Cat 19**	**Cat 4**	**Cat 10**
UBCF	1.189	1.01	1.243	0.887	0.759	0.944
SVD	4.285	4.341	4.273	4.111	4.220	4.139
UBCF (normalized center)	1.23	1.015	1.293	0.934	0.757	0.970
SVD (normalized center)	1.107	0.914	1.027	0.858	0.668	0.783
UBCF (Z-score)	1.264	1.078	1.099	1.055	0.865	0.808
SVD (Z-score)	**0.936**	**0.950**	**0.968**	**0.701**	**0.733**	**0.771**

Table 24 Over category: RMSE and MAE over User-User, and SVD over normalized and non-normalized data over cagtegories. SVD with Z score are the best choice to decrease the error.

- **iAmélie SR:** For each user a social graph was associated with the recommendation matrix (the general one), so that for each user the similarity when applied would be upon the groups of users who already exist and know him/her. This was done on the Epinions data set based on the trust relation defined explicitly by users. Results from this method are in Table 25.

	RMSE	**MAE**
UBCF	1.543	0.936
SVD	3.954	3.751
UBCF (normalized center)	1.502	1.085
SVD (normalized center)	1.142	0.891
SVD (Z-score)	**0.976**	**0.777**

Table 25 Using the social graph: RMSE and MAE over User-User, and SVD over normalized and non-normalized data. SVD with Z score are the best choice to decrease the error

- iAmélie CR: In this branch iAmélie found the related circles for each user; like in [5], the number of circles is produced for each user and recommendation was then performed on these circles.

4.4. iAmélie evaluation:

As iAmélie was built on an Epinions data set it was not possible to evaluate it directly on active users. Calculating the MAE and RMSE for such a recommender makes no sense because it is not a high accuracy that shows whether the system is a good recommender.

That is why we adopted the following scenario of evaluation: we asked a group of 25 students (18-22) to evaluate the recommendation results for users from the data set. We chose five users from the Epinions data set and showed his/her Epinions profile to 15 different users from the 25 students and asked them to evaluate his/her recommendation over iAmélie. Each instance used one recommender of the four following: CFR, DR, SR, and CR. The questions that we chose for the evaluation were based on the work in [13] where the authors compared recommendation systems based on real users feedback in the film domain.

We chose the questions that can be asked in our example where students are the judges about the recommendation for the five target users. Each time, we showed the students the two systems and asked them to answer the questions. The questions permitted us to compare the two systems, and each time we compared a system with iAmélie. The results in Table 26 show a shift in results to the benefit of the iAmélie recommender, which can be seen in the high choice of 4 and 5 options in the students' answers.

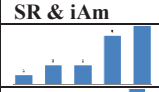
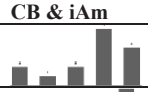
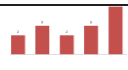
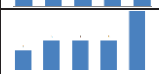
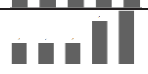
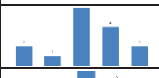
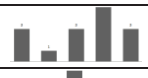
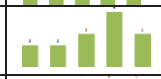
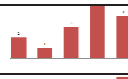
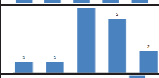
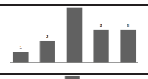
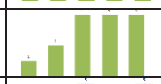
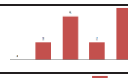
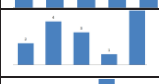
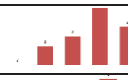
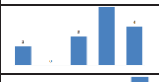
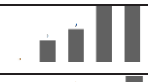
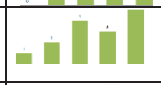
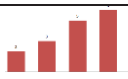
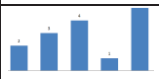
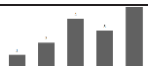
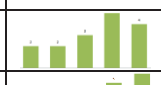
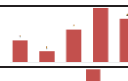
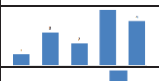
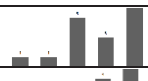
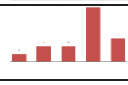
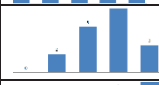
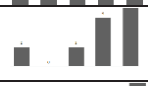
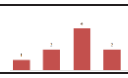
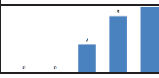
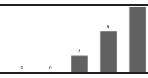
	CFR & iAm	DR & iAm	SR & iAm	CB & iAm
Accuracy w.r. has more items that user X might find appealing?				
Accuracy w.r. has more obviously bad item recommendations for user X?				
Diversity w.r. has more items that are similar to each other?				
Diversity w.r. has items that match a wider variety of moods?				
Understands w.r. better understands user A's taste in items?				
Satisfaction w.r. would better help you find items to buy?				
Satisfaction w.r. would you be more likely to recommend to your friends?				
Satisfaction w.r. of recommendations do you find more valuable?				
Satisfaction w.r. would you rather have as an app on your mobile phone?				
Satisfaction w.r. would better help to pick satisfactory items?				

Table 26 w.r. referes to which recommender. The shift in students' answers towards answers 4 and 5 means that the iAmélie system was well received by the students, better than the other systems

However, the evaluation from the students is only the first implementation of the iAmélie system and further work is needed to evaluate the system on large scale and over different users. Still, we highlight the high motivation among the students when evaluating the system.

5. Conclusion:

Different recommendation systems deal with over different recommendation assumptions; as discussed in this chapter, these assumptions are usually correct in some cases but not in all cases. We showed through a real-life scenario how different assumptions can be correct for the same user in different situations and contexts.

From another point of view, using a recommender based on the assumption of gathering similar users together might lead to the risk of building small, likeminded communities that are not open to new ideas.

iAmélie was built to address this. Being a hybrid recommender that gathers different recommenders into one system, it is able to respond to different user needs with its recommendations.

Results from an Epinions data set showed user interest in iAmélie, preferring it over the traditional collaborative filtering, domain-based recommenders, social recommenders and circle-based recommenders alone. Though iAmélie is not a complete solution, it is a starting point for future research in the recommendation domain.

In the social age, we are free to contact and communicate with any human on the earth. Studies have shown that we tend to communicate with the one in the next room over and those we already know. Of course, this kind of behavior comes with its own set of positive and negative outcomes.

When it comes to online communication and socialization, we believe that recommenders have a positive role to play. With the culmination of massive amounts of knowledge on the web, the right kind of recommender can play a tremendous role in introducing new content, positively shaping preferences, and fostering a more democratic environment where information is easily and enthusiastically shared. It is our hope that iAmélie is the first step in this direction.

6. References

[1] M. D. Ekstrand, J. T. Riedl, et J. A. Konstan, « Collaborative filtering recommender systems », Found. *Trends Hum.-Comput. Interact., vol. 4, no 2, p. 81–173,* 2011.

[2] T. DuBois, J. Golbeck, J. Kleint, et A. Srinivasan, « Improving recommendation accuracy by clustering social networks with trust », *Recomm. Syst. Soc. Web, vol. 532, p. 1–8,* 2009.

[3] R. Pan, G. Xu, et P. Dolog, « Improving Recommendations in Tag-based Systems with Tag Neighbor Expansion », in the Proceedings *of the 3rd International Conference on Computer Science and its Applications (CSA-11),* 2011, p. 12–15.

[4] O. Arazy, N. Kumar, et B. Shapira, « Improving Social Recommender Systems. », *IT Prof., vol. 11, no 4, p. 38–44,* 2009.

[5] X. Yang, H. Steck, et Y. Liu, « Circle-based recommendation in online social networks », in Proceedings *of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining,* 2012, p. 1267–1275.

[6] J. Noel, S. Sanner, K.-N. Tran, P. Christen, L. Xie, E. V. Bonilla, E. Abbasnejad, et N. Della Penna, « New objective functions for social collaborative filtering », in Proceedings *of the 21st international conference on World Wide Web,* 2012, p. 859–868.

[7] B. Adrian, L. Sauermann, et T. Roth-Berghofer, « Contag: A semantic tag recommendation system », *Proc. -Semant., vol. 7, p. 297–304,* 2007.

[8] R. Burke, « Hybrid web recommender systems », in *The adaptive web, Springer*, 2007, p. 377–408.

[9] G. Adomavicius et A. Tuzhilin, « Context-aware recommender systems », in *Recommender systems handbook, Springer,* 2011, p. 217–253.

[10] X. Yang, Y. Guo, Y. Liu, et H. Steck, « A survey of collaborative filtering based social recommender systems », *Comput. Commun., vol. 41, p. 1-10,* mars 2014.

[11] R. Burke, « Hybrid recommender systems: Survey and experiments », *User Model. User-Adapt. Interact., vol. 12, no 4, p. 331–370,* 2002.

[12] B. Xu, J. Bu, C. Chen, et D. Cai, « An exploration of improving collaborative recommender systems via user-item subgroups », in Proceedings *of the 21st international conference on World Wide Web,* 2012, p. 21–30.

[13] M. D. Ekstrand, F. M. Harper, M. C. Willemsen, et J. A. Konstan, « User perception of differences in recommender algorithms », in Proceedings *of the 8th ACM Conference on Recommender systems,* 2014, p. 161–168.

CHAPTER 7: CONCLUSION & DISCUSSION

For three decades, our domain of recommendation has been the subject of intense and rewarding study. Since the beginning of our work, we have agreed with a subject addressed by Martin from his keynote address at the 2009 ACM conference; namely, that algorithms are only a small part of the challenging issues facing the recommendation realm. Although there is certainly a need to refine current recommendation algorithms, a lot of work is yet to be achieved in areas once thought wrapped up, such as user experience, data collection, and other areas that make up the whole of the recommender experience.

That is why most of the work introduced can be classified in this category. We studied recommendation systems focusing on social and contextual recommendation (Chapters 1 and 2) based on our belief that they will be the future of the recommendation world based on their direct link with users' needs.

In the introduction we proposed four problems in the recommendation domain that we wanted to elaborate on during our work. The following summarizes how we approached and addressed each of these four problems, knowing that we did not propose new algorithms and instead combined existing.

First problem: ***C****an we predict from spontaneous actions on social networks information that was not explicitly provided by users? For example, can we predict shopping information about users from their actions on Facebook and Twitter?*

We conclude that we can. We carried out a survey of 63 Facebook users, where we found that none joined Facebook for commercial interests, yet we were able to use their implicit Facebook information to extract useful and correct commercial profiles. Then, a test on 2,000 Facebook users and 12,000 Twitter users affirmed our conclusion. The amount of commercial information was relatively high compared with that present in Epinions data sets.

Second problem: *Can we transform the implicit data in GPSNs into a Recommendation Engine, or a social machine of recommendation?*

We highlighted that if this was possible then we would be able to create a social machine fed by users' actions on GPSNs.

With this goal in mind we created iSoNTRE, the Social Network Transformer into Recommendation Engine. We defined iSoNTRE as a hybrid social recommender designed to transform the GPSN into a recommendation engine. To achieve its goal, iSoNTRE transforms the raw data (unary data, or implicit data) about users into a matrix of recommendation. iSoNTRE performed its goal by combining content-based techniques in which it tries to extract, for users and resources, the concepts that appear in their work in a specific domain, such as films or shopping. Then, using a method that is the inverse of matrix factorization, a whole matrix of user, resource, level of interest (predicted value) was built. Over this matrix any recommendation method can be applied. More than that, we showed that iSoNTRE can be applied over different domains.

***Third problem**: How can one evaluate such a social machine or recommendation engine aside from using traditional accuracy metrics like MAE and RMSE?*

The iSoNTRE evaluation was performed using the usual metrics (MAE, RMSE) as a first step, but then was also tested on Twitter using Sirens, which were automated Twitter accounts that made real contact with real users to receive genuine feedback. On Facebook, the tests were based on user feedback. Users were invited to evaluate their recommended items within a short window of time (two minutes). The results for both Twitter and Facebook were encouraging.

***Fourth problem:** Given the different assumptions in different recommendation systems, discuss these can one propose a solution that can respond to different correct assumptions at the same time?*

We took a step backward in the recommendation domain, analyzing the different assumptions of the different recommenders to show that while all the assumptions could correct, none alone could represent the whole reality of recommendation. In real life, cases for each user may contain different assumptions that are correct in different situations and contexts. That is why we proposed iAmélie. iAmélie considers the different assumption in one hybrid system. It utilizes users' social context to build its four components of CFR collaborative filtering recommendation, DR domain recommendation, SR social recommendation and CR circle recommendation. However, a simplified copy of iAmélie was built in which DR was treated as a special case of CFR as well as a CR as a special case of SR; this is why the final iAmélie system was rather simple. The results compared with each of the four recommenders showed a high preference of users to iAmélie. Tests on iAmélie were performed through an Epinions data set and the results were collected over users who had evaluated the recommended items.

1. Lessons Learned

GPSNs richness in users' information:

Social networks are a source of explicit and implicit user information. What surprised us, however, was the amount of commercial information to be found within social networks. An interesting point in this regard is that while users usually find it hard to provide information about themselves, they spend more and more time on social networks sharing valuable information about themselves and their relations.

Using this information will be of great help in any number of different domains. However a number of problems need to be solved. First, the raw data requires further attention, such as how one extracts words, how one moves from words to concepts, how to find the correct concepts according to the context, how to solve the ambiguity problem, how to choose the source of world knowledge to find the relevant information, how to filter it from non-relevant information, and so on.

High interest in the live tests:

Users showed a high motivation when trying out the Facebook application, as well as the iAmélie systems. They were excited about the idea of extracting their commercial information and wanted to use iAmélie in their daily lives.

We highlight the fact that the small number of students and friends is not a representative sample, but the similarity in users' reactions gives us a high indication of the problems still facing the recommendation domain.

In order to make them more practical and robust, we believe recommendation systems ought to transition from offline data sets towards live tests.

Social & contextual recommenders are the future:

As discussed throughout the study and specifically in Chapter 6, we noted that users change their needs according to different cases and contexts. Combining social and contextual recommenders meets people's consumer needs.

The availability of social information has made social recommenders possible. As we proposed in Chapter 4, contextual information can be extracted from social networks and then combined with contextual information, such as time and geographical information that can be extracted from other sources like users mobiles.

Combining contextual and social information enhances recommendation quality and makes for more satisfied users. We noticed an elevated interest in both domains in the latest works of recommendation.

2. Future Work:

Our work has its limits; in this section, the main areas for further consideration are:

1. Solve the disambiguation problem, move from words to concepts:

In our work we mishandled an important issue; specifically, how to move from words to concepts. In addition, the disambiguation of concepts requires further research. We can also further consider how each user understands the concept according to the context. Lastly, using other sources of world knowledge like word net or the ODP of Wikipedia may prove beneficial in future work, as well as trying different world knowledge over the different domains.

2. Linking iSoNTRE and iAmélie together:

Although the two models were created to work in concert, linking them was not possible during our work. In order to operate fully, iAmélie needed all kinds of recommendation information (users, items, ratings, and relations) in its system, which was not available at the level at which we left iSoNTRE. Additionally, we saw an advantage in trying iAmélie over a classic Epinions data set in order to show that the solution can work in traditional cases as well as in new cases like iSoNTRE. However, following our belief in the need to try recommendation systems by live users, iAmélie was built over an Epinions data set and tested through students to decide to what end iAmélie properly functioned.

3. Adaptive Learning of Sirens:

Creating the Sirens was a necessity due to the limits Twitter applies for sending tweets containing user mentions. When creating the Sirens, we randomly assigned them personalities, which remained the same during the trials. While the Sirens succeeded in their task they can be pushed to a new level of performance. Intelligence can be added to them so that they evolve with time. Based on the feedback of users, they can update their behavior (for example, their way of tweeting or their personalities) and self adjust to a personality that is more attractive. Genetic algorithms should be used to create new personality types in order to find the ones that attract the most attention and feedback from users. We note that Sirens can

be used for more than delivering offers to users and can be used over different domains. To conclude, we ask to what end can users determine whether Sirens are real users?

4. Integrating different contexts:

We dealt with the different assumptions behind using recommendation systems by users in iAmélie systems, which had previously not been dealt with at this level. iAmélie considered a part of context information when recommending, however context contains a high variety of information that can enhance recommendation, like geographical and temporal information. As discussed and demonstrated in chapter 4, contextual information exists in social networks and can be extracted from it, so integrating it in our models will enhance the recommendation.

5. Dynamic moving from one context to another:

For now, users choose to move from one context to another in iAmélie system. This works especially well since we designed the system in a simple manner. By adding intelligence and the dynamic of moving from one case to another we can help users gain time and save effort, as the system will learn from users' actions, context, and needs over time. This can be of great help when a user is in a hurry or reaching our system from a mobile telephone, where the least amount of clicks will lead to a better user experience.

6. Explain recommendations:

In iAmélie we provided explanations of the results at the level of a user's similar users; however, explaining results has been shown to enhance user satisfaction, even if changes in accuracy are not significant. As a first step, we can show the user the kind of the concepts that lie behind his recommendations. Going forward many future enhancements can be taken into consideration.

7. Integrating users' feedback

The way in which users evaluated their predicted shopping profile in Facebook can be generalized to enhance recommendation all over both the iSoNTRE and iAmélie systems. Users can refer to their profiles, which their recommendations were derived from, and can modify it by adding or deleting more concepts or items.

8. Using external knowledge from one social network into another.

iSoNTRE extracts information from different social networks. That extracted information, while coming from the same user, may change from one social network to another. What happens if we combine the information extracted from both social networks, or from multiple social networks? Can we use the commercial information extracted from Twitter for an Epinions recommendation? Or make a Facebook recommendation from Epinions data? And what added values will this offer the user and the whole system? Can we consider the information extracted in one domain as knowledge that can be used in the different recommendation systems related to the domain? Does this enhance or diminish the quality of recommendations?

We hope that continued work on the iSoNTRE and iAmélie systems will help us address these challenges and lead us to solutions that are applicable toward the research and day-to-day life. We believe that recommendation is a critical domain, where each recommended item that's consumed changes us incrementally but forever.

3. Publication

Conferences:

R. Chamsi Abu Quba, M. Rupert, S. Hassas "*Search by Role: A New Paradigm to Search by Similar Users using Social Information*", in the Proceedings of IDIAS 2012.

R. Chamsi, S. Hassas, U. Fayyad, and H. Chamsi "From a "*Cold" to a "Warm" Start in Recommender systems*", in the Proceeding of Enabling Technologies: Infrastructure for Collaborative Enterprises WETICE, 2014, Italie, Parme.

R. Chamsi, S. Hassas, U. Fayyad, M. Alshomary, "*iSoNTRE: the Social Network Transformer into Recommendation Engine*", AICCSA, in the Proceeding of AICCSA, 2014, Qatar, Doha

A chapter in a book :

R. Chamsi, S. Hassas, U. Fayyad, H. Chamsi and C. Gertosio, "*iSoNTRE : transformateur intelligent de réseaux sociaux en environnement de moteur de recommendation*", in the book « Les moteurs et des systèmes recommandation », ISTE, Septembre 2014

R. Chamsi, S. Hassas, U. Fayyad, H. Chamsi and C. Gertosio, *"iSoNTRE : The Intelligent Social Network Transformer into Recommendation Engine framework"* in the book, « Recommander systems », ISTE and Wily, Décembre 2014.

INDEX

LIST OF FIGURES

LIST OF TABLES