Università degli Studi di Cagliari

DOTTORATO DI RICERCA IN

INFORMATICA

Ciclo XXVI

Mining User Behavior in Social Environments

Settore/i scientifico disciplinari di afferenza

INF/01 – INFORMATICA

Presentata da: Matteo Manca

Coordinatore Dottorato Prof. Gian Michele Pinna

Tutor/Relatore Ing. Salvatore Carta, Dott. Ludovico Boratto

Esame finale anno accademico 2012 – 2013

Acknowledgements

First I want to thank my family, my friends, my colleagues and everybody who supported me with their constant love and encouragements during all these years. Moreover, I offer my gratitude to my supervisors Ludovico Boratto and Salvatore Carta that guide me throughout my PhD thesis.

Abstract

The growth of the Web 2.0 has brought to a widespread use of social media systems and to an increasing number of active users. This phenomenon implies that each user interacts with too many users and is overwhelmed by a huge amount of content, leading to the well know "social interaction overload" problem. In order to address this problem several research communities study Social Recommender Systems, which are information filtering systems that operate in the social media domain and aim at suggesting to the users items that are supposed to be interesting for them.
Social Recommender Systems usually filter content by exploiting the social graph or by mining the user content. Since the social domain is characterized by a continuous and quick growth of the the amount of content and users, both these approaches face some problems to produce accurate and up-to-date recommendations.

This PhD thesis proposes some social recommendation approaches based on the mining of the user behavior, i.e., on the exploitation of the activity of the users in social environments, in order to produce accurate and up-to-date recommendations.

Contents

Chapter 1

Introduction

> *I am a little tired this period, so I need to make a holiday in a relaxed resort, surrounded by the nature, where I can run away from the chaos of the city and from the stress of the work. During my holidays I would like to make some excursions, visit interesting natural places and, in the evening, just relax listening good jazz music. But...which is the best holiday to satisfy my needs? I would need someone or something to help me make these decisions.*

Everyday people have to make some choices and decisions, for instance which movie to rent, which smartphone to buy, etc. Until some years ago, in order to make right decisions, they asked for suggestions to other people, like friends or family. Nowadays, people usually spend several time reading up on the web before making a decision about something. To this purpose the World Wide Web (WWW) is a useful tool but, at the same time the coming of the web 2.0 applications, brought to a quick growth in the amount of available data. In some cases, this produced the opposite effect, i.e. confusing the user and leading to scarcity of attention and to the well know ”information overload“ problem (i.e. with a large amount of available choices, it becomes difficult for a user to identify the item that best fit her/his needs) [Ricci et al., 2011]. The scarcity of attention and the social interaction overload problem are strongly related to each other, given that with the social interaction overload each user has too many potential users and items to interact with and this does not allow the user to focus on users or items that might be interesting for her/him.

This phenomenon is addressed by Recommender Systems (RSs), which are information filtering systems that aim at suggesting to the users items that are supposed to be interesting for them. As proposed in [Burke, 2007], RSs may be classified into six different categories: *Content-based*, *Collaborative Filtering*, *Demographic*, *Knowledge-based*, *Social Recommender System*, *Hybrid Recommender System*. In this work particular attention is put on Social Recommender Systems, which target the social media domain. These systems, also known as “Community-based”, deal with the information overload over social media users, by recommending the most interesting and relevant content [Guy and Carmel, 2011]. Given an active user, they produce recommendations based on the preferences of the other users that are linked to the active one.

Social media systems are internet based applications, built on the ideological and technological foundation of Web 2.0, that allow the creation and exchange of user generated content [Kaplan and Haenlein, 2010].

This type of systems are characterized by a rapid and continuous content evolution and growth, so users are overwhelmed by content and by interactions with other users.

Moreover, these type of systems are often not specific for a given topic or for a limited number of users, but are systems opened to every topic and to everyone, so it becomes difficult to understand which content are rumors and which are trusted or to understand which users are trusted or which are malicious. ”Social Recommender Systems“ try to address users only to interesting and trusted content (or users) in the social media domain. Recommender systems and social media applications mutually benefit each other, because recommender systems have a relevant weight in the success of social media applications, helping users to find the items that best fit their needs, while social media applications introduce new data like tags, vote, likes, social relationship, etc which can be used by recommender systems [Guy and Carmel, 2011].

There are different areas that are covered by social recommender systems: content recommendation, tags recommendation, user recommendation, community recommendation, etc.

Social Recommender Systems can filter content in two main manners: by exploiting the social graph or by mining the user content. It is known that the mining of the graph suffers from scalability issues [Gupta et al., 2013], while the approaches that mine the user content in order to build a user profile, usually are based on complex algorithms (for example systems based on TF-IDF [Chen et al., 2009]). Since the social domain is characterized by an ever growing amount of content and users, these approaches might face some problems to update the user preferences in order to produce interesting recommendations.

In order to overcome the previously mentioned problems, this PhD thesis proposes approaches based on the mining of the user behavior, i.e., on the exploitation of the activity of the users in social environments, in order to:

- produce friend recommendations;
- produce tag recommendations;
- study how social media systems can be used as "persuasive technologies" (in other words the motivational aspect of social media systems is studied).

By mining *how* users interact with the content, instead of mining content itself or the social graph, this work aims at developing accurate approaches, designed to operate in social environments.

The rest of the chapter is organized as follows: paragraph 1.1 presents an introduction to the user recommendation topic; paragraph 1.2 introduces tag recommender systems; paragraph 1.3 makes an introduction of the social motivation aspect in the Human-Computer interaction domain, while in paragraph 1.4 the contributions of this work are discussed

1.1 User Recommendation

The main goal of user recommender systems in the social domain is to suggest friends (i.e, recommendations are produced for pairs of users that are supposed to be interested at each other's content) or people to follow (i.e., recommendations are produced for a user, in order to suggest users that might be interesting for him/her) [Guy et al., 2013]. The recommendation of a friend involves mutual interests, so the list of recommended friends and the list of recommended people to follow may be different. In fact, given two users u_i and u_j , u_i might be interested in u_j content but not vice versa. This means that u_j would be recommended to u_i as a user to follow, but not as a friend.

So the design of a friend recommender system is different from the design of a people to follow recommender system, since they involve different notions of users similarity.

User recommender systems can be classified into three main areas:

- Systems based on the exploration of social graphs, which analyze the set of users that interact with the considered user, in order to produce recommendations. These systems usually recommend either the closest users in the graph, like friends of friends and followees of followees (the most famous example of this type of systems is Facebook[1]), or can run a random walk algorithm on the social graph to produce recommendations based on the set of users that have the highest probability to be crossed [Gupta et al., 2013].

- Systems that analyze how the user interacts with the content of the system (tags, likes, shares, posts on news, bookmarks, pictures, etc.) to exploit the interests of the users. These systems usually use complex algorithms, for example, some approaches build a user profile using TF-IDF vectors based on each content the user interacts with [Chen et al., 2009]. Once a profile for each user is built, recommendations are produced by identifying users with similar profiles.

- Hybrid systems, which explore both the social graph and the interactions of the users with the content (an example is represented by [Hannon et al., 2010]). The use of different sources of data to build the recommendations usually leads to an improvement of the recommendation quality but at the same time it increases the complexity of these systems.

[1] http://blog.facebook.com/blog.php?post=15610312130

1.2 Tag Recommendation

In this paragraph tag recommender systems, which are a type of social recommender systems that operate in a tagging system, are introduced. Precisely, tagging systems are social media application that allows users to add keywords (so called tags) to classify resources [Zhou et al., 2012]. Real world examples of tagging systems are Del.icio.us[2], Flickr[3], Last.fm3[4], CitULike[5]. These systems are characterized by some well-known linguistic limitations. For example if a user classifies a resource with tag ”kiwi”, it may be ambiguous, because kiwi is a fruit, a software, a bird and a plant, so this can lead to incorrect relations between tags and resources. Moreover, people can use different keywords to classify the same resources, for example a picture about a pizza may be tagged with different tags like “pizza”, “italian food”, “delicious”, etc. So, tagging systems have the advantage to allow users to freely choose which tags to use to classify the resources of the system but, on the other hand, this freedom may complicate the search activity of the users within the tag space. In fact, given that users may use different tags for the same resource, a user might search a resource using a query that contains a set of tags different from the ones used to classify the resource, without finding it. So, in order to find a

[2]http://delicious.com/
[3]http://www.flickr.com
[4]http://last.fm
[5]http://www.citeulike.org

resource, it might be needed to search several times using different tags and people should evaluate the relevance of the retrieved resources. In order to support users in their search activity and in their tag activity and thus addressing people to interesting content, tag recommender systems have been developed. Many of them are based on approaches that group tags in order to help the identification of a context, which would avoid polysemy and synonymy thus making resources retrieval easier [Bielenberg, 2005].

1.3 Social Motivation

As already mentioned, in recent years the Social Web experienced an exponential growth; another area that grows proportionally with the Social Web is the Human-Computer interaction (HCI) area.

Human-Computer interaction is defined as "a discipline concerned with the design, evaluation, and implementation of interactive computing systems for human use and with the study of major phenomena surrounding them" [ACM SIGCHI, 1992]. The Social Web, instead, is defined as a set of relationships that link together people over the Web and a set of applications built on top of these relationships [Appelquist et al., 2010].

Nowadays, researches are conducting some studies about how the two disciplines mentioned above can work together. In fact, it is well known that HCI and the Social Web can benefit each other, for example, Human-Computer interaction application could be developed in the Social

Web scenario, in order to improve relationships among people. Moreover, some features still have not been explored in the Web 2.0; for example, in [Turetken and Olfman, 2013], authors state that, at now, the "any time, any place" nature of HCI has not been widely studied in the Social Web.

At the state of the art, several studies, that exploit the social interactions between users in order to motivate people to exercise more, have been presented [Consolvo et al., 2006, Virzi, 1992, Buttussi et al., 2006, de Oliveira and Oliver, 2008]. In order to study the social motivation aspect, in this work a web application, based on two Android applications that try to motivate people in their exercising activity, is presented. The first Android application, named *EveryWhere Run* [Mulas et al., 2011, Mulas et al., 2013a], allows users to get a workout plan from a personal trainer, while the latter, named *EveryWhere Race* [Mulas et al., 2012], is based on the concept of virtual competitions (*races*).

The interaction between users and personal trainers and the capability to interact in real-time with other users implemented in the Android applications highlighted great improvements in the motivation of users to exercise regularly.

The web application presented in this thesis includes some features (e.g., the creation, the subscription or the participation to a race), previously available only in the Android application and implements some new features creating an artificial cognitive system able to enhance the users experience and stimulating them to exercise more.

The social aspect added to the web application allows users to have an improved awareness of their performances, thanks to the feature that allows to share the user experience with their Facebook friends.

Moreover, the use of the Android applications allows users to stimulate the attention, motor, visual, and spatial processing capabilities or to know how well the other users in a race are doing.

At the state of the art, there are several applications that guide the users during their physical activity and that provide a web community (e.g., Endomondo, Runtastic, Nike+, etc.), but the web application presented in this work has been designed in order to favor also the social interactions *before* a user workout. The objective of this feature is to encourage people to partecipate in a race and not only allowing to share the results of a work out when the user has already completed her/his activity.

The interactions of the users with Facebook social network is a crucial aspect, because it leads to the phenomenon of the "social influence" (a widely known concept in sociology and viral marketing) in the social network domain [Cha et al., 2010]. The idea behind the "social influence" is that the enhancements of a user in her/his exercising activity can encourage and motivate other users to improve their performances exercising regularly.

The presented web application focuses mainly on the organization of races, since a race involves more than a user, so this is the ideal scenario to link Human-Computer interaction with Web 2.0 applications.

1.4 Contributions of the thesis

In the following some limitations of the state-of-the-art are pointed out, then the proposed systems are introduced.

Recommender systems usually suggest items that have a strong match with the user profile of the target user, consequently she/he always receives recommendations for items too similar to those that she/he already considered and never receives suggestions for unexpected, surprising and novel items. This approach is able to produce very accurate recommendations, but it does not means that the recommendations are also useful; this lack of diversity in the recommendations, known in literature as "serendipity problem" or "over-specialization problem", worsen the user experience and does not give the users the opportunity to explore new items and to discover items she/he might like [Abbassi et al., 2009]. To face the serendipity problem, the produced recommendations should be accurate but, at the same time, also novel (in some cases also serendipitous).

In relation to the user recommendation topic, in this PhD thesis a friend recommender system that operates in the social bookmarking domain is presented. In [Gupta et al., 2013], authors highlight that Twitter is an "interest graph", rather than a "social graph"; this definition of interest graph can also be extended to social bookmarking systems, since a user can add as a friend or follow another user, in order to receive her/his newly added bookmarks. In [Gupta et al., 2013], authors highlight that recommender systems that exploit the interest graphs suffer from scalability issues and, in order to reduce the complexity of the recommender system, it is better to avoid the use of the user profile information in the recommendation activity. Furthermore, in social media domain, applications are characterized by a huge amount of data that evolve rapidly and continuously, so it is essential to reduce the complexity of the system in order to infer the users interests changes and produce updated recommendations.

In relation to the tag recommendation topic, recently, several approaches have been proposed to cluster tags in order to face the problems related to tagging systems. One limitation of the state of the art works is that they do not monitor the users search activity performed into the system in order to use the behavior of the user like a source of information. These systems create associations between tag and resources only when a resource is loaded into the system but then they do not update in any way these associations, so if a resource has been associated to a misleading tag this ambiguity affect the performances of the systems. Another problem that may affect a tagging systems is the well known *Cold Start Problem*, i.e., if a user is new or a resource is not similar to any of the existing resources, no tag can be recommended to the user.

Regarding the social motivation area, none of the other HCI applications that operate in the sport environment focus on the social influence, i.e., on how the objectives of an application can be achieved and how performances can be improved thanks to the interaction with the social media.

For each area explored in this thesis, this work dealt with the following research questions:

Research questions:

- What information should be used in order to produce accurate, novel and serendipitous friend recommendations?
- How can the available information be exploited in order to update the user preferences and other information useful to produce tag recommendations?
- How can the use of the social media be exploited in order to encourage users to adopt specific behaviors?

In this thesis two different recommender systems and a study about the motivational aspect of the social medias are presented. The first system belongs to the user recommendation field (precisely friend recommendation), while the other one belongs to the tag recommendation field.

The former is a friend recommender system for a Social Bookmarking System, experimented on Delicious. This system exploits the user interests, in order to recommend other users interested to the same topics. Since in literature it is known that the methods that analyze graphs cannot exploit interests and are not scalable [Gupta et al., 2013], the proposed system makes a selective use of available information with the intent to use as less information as possible so it does not need to many computational resources.

Furthermore, the friend recommender system presented in this work does not suffer from the "serendipity problem", in fact, it is able to produce accurate recommendations that are also novel and serendipitous. Until now, research communities have developed some approaches of user recommendations [Zhou et al., 2010] but there are no approaches in literature that build friend recommendations in the Social Bookmarking Systems domain, so the presented system put the bases on a research area not yet explored in this application domain.

The presented system has been compared with state of the art algorithms, in order to evaluate the quality of the recommendation but not only in terms of accuracy. In fact, usually, recommender system are evaluated only in terms of accuracy of the recommendations with standard information retrieval metrics like MAE (Mean Absolute Error), Precision (fraction of retrieved instances that are relevant), Recall (fraction of relevant instances that are retrieved), etc. [Baeza-Yates and Ribeiro-Neto, 1999].

In [Ricci et al., 2011], authors highlight that in this domain there are other aspects to consider in order to evaluate a recommender system. Some examples are:

- *Trust*. May be trust about other users or trust about recommendations;
- *Explanation*. Each recommendation may be explained therefore a user can understand the reason of a given recommendation;
- *Persuasiveness*. People are likely to accept recommendations given by trusted and credible sources than recommendations given by anonymous sources;
- *Novelty*. How many recommended items were unknown for the target user that receives the recommendations;
- *Serendipity*. How surprising the successful recommendations are.

Given the serendipity problem above introduced and given that the accuracy of the recommendations is not enough to guarantee a good user experience, this work focuses about some aspects that allow to evaluate the quality of a recommender system from different perspectives.

Precisely, the precision of the system has been evaluated but *novelty* and *serendipity* have also been measured. In fact when the system build a friend recommendation, indirectly it is also recommending the content of the users; so, the content of the recommended users is analyzed, in order to find out which recommendations are novel, in other words, recommendations for unknown content for the user that receives the recommendation.

On the other hand serendipity measures how surprising the successful recommendations are [Shani and Gunawardana, 2011].

The other recommender system, named (RATC - *Robust Automated Tag Clustering*-), is based on tag clustering and monitors the users behavior to exploit implicit feedbacks left by users in order to improve the performances of the system. Monitoring the user behavior makes the system able to create and continuously update tag-resource associations and tag-tag associations, rewarding the real semantic relations among tags and penalizing the misleading ones. Moreover, RATC is able to produce recommendations without using neither the user profile nor the content of the resources, so it is not affect by the cold start problem and the complexity of the system is reduced. The proposed tag recommender system produces novel tag recommendations, since it does not consider the tags already used by the target user. On the other hand, the recommended tags are not serendipitous since they are in the same cluster of the tags used by the user. In this case, serendipitous recommendations would be a problem, because they could lead to misleading tag-resources associations or could complicate the search activity of the users.

In the study dealing with the motivational aspect in social media systems, two Android applications and a persuasive web application, which aims to help and motivate people to do more physical activity and to do it better, have been developed. The applications, based on the concept of *virtual personal trainer* and *virtual race*, allow users to interact with the Facebook social network. The conducted studies on these applications show how these interactions create a link between the Human-Computer Interaction (HCI) domain and the social web domain, improving the motivation of users to conduct a more active lifestyle.

So, the contribution of the thesis can be recapped as follows:

- In the field of friend recommendation, the presented work is the first approach able to recommend friends in a Social Bookmarking System; recommendations are produced, without using any graph, by exploiting users interests in a selective way in order to reduce the complexity of the system and to not have scalability problems. Since [Lops et al., 2011] highlights that there is no an universal definition of novelty and serendipity and in the literature there is no other work previous to this that recommends friends in the social bookmarking domain, a new definitions of novelty and serendipity in this context are proposed;

- In the field of tag recommendation, our approach is the first social recommender system that uses clustering to produce novel tag recommendations; in a social domain, where everything evolves quickly, a form of classification that does not require supervision like clustering is an extremely simple and strongly effective way to produce associations between similar tags that are used then to produce recommendations.

- About the study on the motivational aspect of social media systems, the proposed web application is the first that exploits virtual races to improve people motivation, and allows to challenge Facebook friends in real time encouraging, in this way, people to exercise more.

The rest of the thesis is organized as follows: Chapter 2 presents related work for the different domains discussed in this PhD thesis; Chapter 3 presents the friend recommender system and the related study; in Chapter 4 the tag recommender system based on tag clustering is presented and discussed; Chapter 5 presents a study of how the social aspect can be used as persuasive technology and in Chapter 6 the conclusions and future developments of this Phd thesis are discussed.

Chapter 2

Related Work

As already mentioned in the Introduction, different topics that belong to the Social Recommender Systems domain are studied. In particular, tag recommendation and user recommendation are the main aspects considered on this work. For this reason in the following some works about social recommender systems in general are first presented and then in paragraph 2.2 the state of the art on tag recommendation is discussed, while in paragraph 2.1 the state of the art in user recommendation domain is presented.

Also in the HCI field the social factors play an essential role; for example many mobile applications try to motivate people to do more physical activity using technique based on social influence theory. Others applications allow users to share their activity performances with their contacts, on different social network sites, often receiving feedbacks that could be seen as a kind of recommendations.

In 2.3 some works, that aim to encourage people to do more physical activity, or to do it best, by means of mobile applications that often include also social aspects, are presented.

2.1 User Recommender Systems

Accordingly to the classification of the user recommender system done in the Introduction, in this paragraph the main approaches developed at the state of the art are presented.

2.1.1 Systems Based on the Analysis of Social Graphs

In [Gupta et al., 2013] authors present Twitter's user recommendation service, which allows to daily create a huge amount of connections between users that share common interests, connections and other factors. The proposed system suggests people to follow based on shared interest and on the social graph that is fitted in memory on a single machine. In order to perform the recommendations authors build a Twitter graph in which vertices represent users and the directed edges represent the "follow" relationship. The graph is stored in a graph database called FlockDB, which is based on mysql and then data are processed with Cassovary, which is an open source in-memory graph processing engine. Finally, the system builds the recommendations by means of a user recommendation algorithm for directed graphs based on SALSA. The algorithm proposed in this work differs on several aspects; first of all, the proposed system produces friends recommendations, furthermore it uses just a restricted set of available information, without considering the social graph. In fact, as highlighted in [Gupta et al., 2013] the analysis of a social graph suffers from scalability issues and, in order to limit the complexity of the recommender system, no user profile information could be used to build the recommendations.

In [Liben-Nowell and Kleinberg, 2003] authors model the user recommendation problem as a link prediction problem. They develop several approaches, that analyze the proximity of nodes in the graph of a social network, in order to infer the probability of new connections among users. Experiments show that the network topology is a good tool to predict future interactions.

2.1.2 Systems based on the Interactions with the Content

Quercia et al. [Quercia and Capra, 2009] describe a user recommender system based on collocation. The proposed framework, called FriendSensing, recommends friends by analyzing collocation data. In particular, it uses short-range connections like Bluetooth, mobile phones "sense" and records which other mobile devices are in proximity. FriendSensing then processes those records and suggests to users people they may know. In order to produce the recommendations, it uses geographical proximity and link prediction theories. FriendSensing mainly operates in two steps: first, it uses short range radio technologies of modern mobile phones to build a log which contains information about how many times two devices have met and how much time they stayed in touch; then relevant encounters are inferred from the log records and arranged in a weighted social network. This network is used to produce personalized lists of people each user may know. The algorithm presented in this thesis cannot be compared with the one proposed in [Quercia and Capra, 2009] because it does not have such type of information deriving from mobile phones.

In [Brzozowski and Romero, 2011], researchers present a study that considers different features in a user profile, behavior and network in order to explore the effect of *homophily* on user recommendations. They use the Dice coefficient on two users sets of tags and they find that similar tags do not represent a useful source of information for link prediction, while mutual followers are more useful for this purpose. As previously highlighted, the presented friend recommender system focuses on producing friend recommendation based on users' content (tag, bookmarks, etc.).

2.1.3 Hybrid Systems

In [Zhou et al., 2010] authors propose a framework of user recommendation, based on users' interests and tested on Yahoo! Delicious. The proposed framework operates in two main steps: first, it models the users' interests by means of tag graph based community detection and represents them with a discrete topic distribution; then, it uses the Kullback-Leibler divergence function to compute the similarity between users' topic distribution and the similarity values are used to produce interest based user recommendation. Differently from this framework, the aim of the approach proposed in this thesis is to produce friend recommendations (i.e., bidirectional connections) and not user recommendations, which are unidirectional.

Chen et al. [Chen et al., 2009] present a people recommender system in an enterprise social network called Beehive, designed to help users to find known, offline contacts and discover new friends on social networking sites. Authors conducted two separate experiments, i.e., a personalized survey and a controlled field study. In the former, authors select 500 active users that were randomly chosen from all users satisfying several criteria and present them 12 recommendations (3 for each algorithm); then each user is asked to answer some questions related to their friending behavior, and to rate personalized recommendations created from each algorithm. In the latter experiment authors deployed the four different recommender algorithms as a feature of the site involving 3000 users randomly selected from all users that satisfy several criteria. The 3000 users were randomly divided into 5 groups; four were experimental groups, each one getting recommendations from a single algorithm only, while the remaining group was a control group that did not get any recommendations. Recommendations were presented in a widget, in which users could respond to the recommendation by choosing one of three actions: connect to the person, ask to be introduced, and decline by choosing "not good for me". With the proposed study, authors demonstrate that algorithms that use similarity of user-created content were stronger in discovering new friends, while algorithms based on social network information were able to produce better recommendations. The system presented in this thesis cannot be compared to those proposed in [Chen et al., 2009] since it is applied to a delimited enterprise social network domain.

In [Hannon et al., 2010], authors propose a user recommender system (called *Twittomender*) that, for each user, builds a user profile based on user's recent Twitter activity and user's social graph. The proposed system operates in two different manners; in the former mode the user puts a query and the system retrieves a ranking list of users, while in the latter mode the query is automatically generated by the system and it is mined by the user profile of the target user (the target user is the user that receives the recommendations). The work presented in this thesis does not use the social graph or any connection information between users and, furthermore, in building recommendations it considers the friendship relationship and not the "user to follow" relationship.

In [Guy et al., 2009] authors present a recommender system for the IBM Fringe social network, based on aggregated enterprise information (like org chart relationships, paper and patent co-authorship, project co-membership, etc.) retrieved using SONAR, which is a system that allows to collect and aggregate these kinds of information. The proposed system differs from other works in literature, because it does not use only the social network information but also information about other systems. Authors deployed the people recommender system as a feature of the social network site for a period of four months and the results showed a highly significant impact on the number of connections on the site, as well as on the number of users who invite others to connect. The proposed study is based on specific enterprise data, so for this reason it is hard to make a comparison with the friend recommender system presented in this work.

2.2 Tag Recommender Systems

As highlighted in the Introduction, the presented tag recommender system is based on tag clustering. Many works have been proposed in the literature, which aim to cluster tags or to recommend tags; for this reason related work about tag clustering is first presented and then related work on tag recommendation is presented.

2.2.1 Tag Clustering

In [Specia and Motta, 2007], authors present an approach that allows to infer the semantics behind a tag space in a social tagging system, so that this collaborative organization can emerge in the form of groups of concepts and partial ontologies. This approach is a combination of shallow pre-processing strategies, and statistical methods together with knowledge provided by ontologies available on the semantic web. The algorithm can be recapped in three main steps:

- *Pre-processing:* The pre-processing consisted on filtering out unusual tags, by following the rule that a tag must start with a letter followed by any number of letters, numbers, and symbols like dash, dot, underscore, etc. Tags morphologically similar are grouped, by using the Levenshtein similarity metric and filtering out infrequent and isolated tags.

- *Clustering:* the algorithm performs a statistical analysis of the tag space, in order to identify clusters related tags. Clustering is based on the similarity among tags given by their co-occurrence.

- *Concept and Relation Identification*: the algorithm uses knowledge provided by different sources, like Wikipedia and Google, to infer possible relationships between tags in each cluster and, if they exist, categorize them.

The presented approach differs from the one proposed in [Specia and Motta, 2007] because it does not pre-process the tag space. Furthermore, the presented approach is able to remove the noise by monitoring the user behavior.

In [Hamasaki et al., 2007], authors proposed an approach called HAMA, which tries to integrate a social network with collaborative tagging, in order to extract ontologies. They analyze a case study, using the model for emergent ontologies in academic conferences. HAMA is based on the Mika model [Mika, 2007], which is a state of the art model that describes the relation between social networks and ontologies by using actors, concepts and instances, and illustrating ontology emergence by actor-concept and concept-instance relation. The proposed approach adds a third dimension, i.e., the actor-actor relation, in order to face data sparsity and polysemy problems. The actors-concepts interactions are used to build groups of concepts and keyword associations. *RATC*, differently, does not create explicit users-resources associations but considers users interaction just to build tag-resources associations.

Wu et al. [Wu et al., 2006] propose a probabilistic approach to model the user tagging process, in order to automatically derive the emergent semantic of the tags. After a preliminary study, authors found that: (i) tags are usually semantically related to each other if they are used to tag the related resources for many times, (ii) users may have similar interests if their annotations share many semantically related tags and (iii) resources are usually semantically related if they are tagged by many users with similar interests. Starting from these points, they make some statistical studies about the co-occurences of tags, resources and users. The model represents each entity as a multidimensional vector $\overrightarrow{v} = \{v_1, v_2, ..., v_m\}$ in a multidimensional space called *conceptual space*, where each dimension is a category of knowledge. Hence, if one entity relates to the category of knowledge i, the corresponding dimension v_i of its vector has a high score. The proposed probabilistic model performs the following steps:

- *Dimensions:* it chooses a dimension to represent a category of knowledge;
- *User-Dimension relativity:* it measures the relativity between the interest of user u_i and a given dimension;
- *Resource-Dimension relativity:* it measures the relativity between the semantic of the resource r_i and the chosen dimension;
- *Tag-Dimension relativity:* it measures the relativity between the semantics of a tag t_k and the chosen dimension.

The presented approach differs from that proposed in [Wu et al., 2006], because it does not use any probabilistic model.

In [Giannakidou et al., 2008], authors present a co-clustering approach that considers both social and semantic aspects of the tags, in order to cluster items (tag and resources) of different datasets. In order to perform the clustering activity, the proposed approach represents each resource through the set of tags that have been used for its annotation and use a similarity metric based on tag co-occurences. Furthermore, in order to estimate the semantic similarity between tags, the approach proposed in [Wu and Palmer, 1994] is used; this one allows to use external resources like web ontologies, thesauri, etc., and to map tags and resource's concepts. *RATC* performs the clustering considering only tags, without inferring new knowledge from external sources.

Baeza-Yates [Baeza-Yates, 2005] analyzes query logs in order to create clusters of related queries, which are then used to recommend queries to search engine users. The proposed approach represents each query as an aggregation of term-weight vectors of the documents selected in the answers of the considered query. The weight of each term is computed according to the number of occurrences and the number of clicks of the documents in which the term appears. Once each query is represented as a vector, the clustering process is performed by an implementation of K-means algorithm. The advantage to represent queries by vectors based on selected documents is that they may be clustered and manipulated similarly to traditional document vectors, so it is possible to compute query-document similarity in order to perform query recommendations. *RATC* uses queries differently, since associations between tags are inferred by taking into account the resources that they classify and not by building clusters of queries.

In [Begelman et al., 2006], authors implement a clustering technique that aims at grouping strongly related tags, in order to improve the user experience of tagging services and to avoid the limitations of such types of systems. The presented technique performs two main steps: first, it finds strongly related tags and then it applies a clustering algorithm. In order to execute the first step, it builds a sparse matrix that represents tags, in which the value of each element is the similarity of the two tags. The similarity among tags is based on counting the number of co-occurrences (i.e., tags that are used for the same page) of any pair of tags and computing a cut-off point, which allows to decide when the co-occurrence count is significant enough. The clustering step is performed by means of an algorithm based on spectral bisection and by using the modularity function to evaluate the quality of the computed clusters. Clusters are then used to select the top N similar tags to a tag t_i that is frequent enough in the tag space.

2.2.2 Tag Recommendation

In [Symeonidis et al., 2008] Symeonidis et al. present a tag recommender system, whose main steps can be recapped as follows:

- The algorithm models the entities of the social tagging systems, users, items and tags by a 3-order tensor;
- a Higher Order Singular Value Decomposition algorithm is applied in 3-order tensors to reveal the latent semantic associations between users, resources and tags in order to perform the recommendations;
- a comparison with two state of the art algorithms is performed [Xu et al., 2006, Hotho et al., 2006b]

The experimental results show significant improvements with respect to the state of the art algorithm in terms of effectiveness measured through precision and recall. The proposed approach differs from the one just described because it does not consider which items the users interact with.

In [Rendle and Schmidt-Thieme, 2010], authors describe a tag recommendation system based on PITF model (Pairwise Interaction Tensor Factorization), which is a particular case of the Tucker Decomposition (TD) model with linear runtime, both for learning and prediction. The advantage of this model is that the complexity of the model equation is linear in the number of factorization dimensions, which makes it feasible for high dimensions. The proposed system operates in two steps: first, the system models interaction between users, items and tags, then it uses a Bayesian Personalized Ranking criterion to produce the recommendations. The approach of tag recommendation proposed in this thesis does not use any probabilistic model.

In [Carmel et al., 2010], authors present a framework for social bookmark weighting, which allows to estimate the effectiveness of each of the bookmarks individually for several Information Retrieval (IR) tasks. They consider each bookmark as an indivisible triplet (document, user, tag) and each bookmark is weighted by the framework, according to its predicted effectiveness in describing the content of the document it is associated with, given that it was annotated by a specific user with a specific tag. This framework is able to perform tag recommendations, user recommendations, and document recommendations. The tag recommendation process is done by computing the similarity between each tag and other tags previously used by the user and between each tag and the documents that have to be tagged. The study proposed in this thesis does not consider neither the tags previously used by the users, nor the similarity between tags and documents; in fact this system does not suffer from cold start problem.

Inspired by the PageRank algorithm [Brin and Page, 1998], Hotho et al. present *FolkRank* algorithm [Hotho et al., 2006a]. The basic idea of this algorithm is that a resource tagged by important users with important tags is important itself. In this work authors represent the system through a undirected graph (while, in the PageRank algorithm, the edges of the graph have a direction), where the nodes represent users, resources, and tags and the edges represent the connection between tags and users, users and resources or tags and resources. In order to assign a weight to each node, the algorithm executes a random walk algorithm on the graph and recommendations are built by choosing the top ranked tags associated to a given tag. RATC differs from this approach because it does not use a random walk algorithm to make associations between tags and resources and, moreover, the work proposed in this thesis updates these associations continuously and not only when new resources are added to the system.

In [Givon and Lavrenko, 2009], authors describe a system that recommends tags for full text books. They use a dataset composed only by books written in English, and that belong to the fiction/literature domains, which are split into a training set and a test set. Furthermore, they collect a set of social tags, which they pre-process by means of a stemming task, duplicate removing task. Then, each book is represented as a TF-IDF vector and each tag is associated to a given book by using a Relevance Model, which is a method adopted from Information Retrieval to match documents to a given query. Through Relevance Model, the system selects a set of tags to recommend for each book.

In [Sigurbjörnsson and van Zwol, 2008], authors present an approach to support the user during the tagging process of a photo in Flickr. Given a photo with user-defined tags, a set of candidate tags is derived for each of the user-defined tags, by using a "promotion function" based on tag co-occurrence. The lists of candidate tags are then used as input for tag aggregation and ranking, which ultimately produces the ranked list of n recommended tags. Tag co-occurrence identification is a crucial task in the presented tag recommendation approach. Co-occurrences between two tags is defined as the number of photos in the system for which both tags are used in the same annotation. It is common to normalize the co-occurrence count with the overall frequency of the tag. Once the lists of candidate tags for each of the user defined tags are built, a tag aggregation step is needed to merge the lists into a single ranking of tags to recommend. Even if co-occurrences of tags in resources are considered (like RATC does), the proposed system continuously and implicitly monitors the tagging behavior of users. Similarity between tags is not calculated using a promotion function, which is built with an observation of the tagging system at a certain time, but considering the use of the tags at the moment in which similarities are calculated.

2.3 Social Motivation

In recent years, many studies have been conducted in order to increase physical activity motivation. In [Toscos et al., 2006], authors propose a mobile application, called Chick clique, that tries to encourage teenage girls to adopt a correct lifestyle. Authors targeted teenager girls because other studies demonstrated that they are more likely to become less active throughout adolescence, with respect to their male counterparts. The software provides information about food calories and the necessary amount of steps needed to burn them; furthermore, users can invite their friends and share their achievements and their walking activity with them. The social factor is very important, in fact the enhancement of a user can inspire other users to do best or if some users see that one of their friends is bringing down, they can encourage that user.

In [Consolvo et al., 2006] a mobile application for Symbian, called Houston, is presented. The proposed system uses a pedometer to count the number of steps done by users and allows them to share daily statistics with a set of friends. Experiments, performed on data collected through the use of Houston, show that: (i) users expected to have detailed measures,

(ii) they prefer to use long term statistical report in order to have a detailed overview of the walking activity, (iii) the social aspect helps users to improve their performances and that (iv) users consider more comfortable to use an all-in-one device rather than to use external additional devices. Experiments have been conducted on a homogeneous sample composed by 13 participants (female friends aged 28-42) that were divided into three groups and the study ran for about three weeks. All participants wanted to increase their levels of physical activity and the results have been studied by means of questionaries and interviews.

Battussi et al. [Buttussi et al., 2006] present a PocketPC application, called MOPET, that aims at supporting the physical activity. MOPET uses a GPS devices to monitor user positions during their physical activity in an outdoor fitness trail situated in a public park. It provides navigation assistance by using a fitness trail map and giving speech directions, motivation support and exercise demonstrations by using an embodied virtual trainer, called Evita. Evita shows how to correctly perform the exercises along the trail with 3D animations and encourages the user. The proposed application has been tested with 12 users. In order to test the navigation support, the following variables have been measured: how many times the user followed paths that led away from the trail, how many meters have been run out of the trail, how many meters of the trail the user skipped, percentage of time the user spent on other paths. A questionnaire was administered to each user, in order to infer to the usefulness of the fitness trail maps. In order to test the motivation support, authors administered a questionnaire in which they asked the users how much MOPET motivated them and whether they would come more frequently to the fitness trail if they could use the proposed system. Finally, in order to test the training support, authors filmed users from a distance during exercises and, moreover, they asked users to rate the usefulness of the embodied virtual trainer. MOPET got good results for all the different types of supports.

In [de Oliveira and Oliver, 2008], authors present TripleBeat, a mobile phone application that uses ECG and an accelerometer, in order to support runners to reach their goals, particularly in terms of heart rate. The application assists the user by using musical feedbacks and persuasive techniques, like an interface and a virtual competition. In order to test the application, researches conducted a user study with 10 runners and compared TripleBeat with another previously implemented application, named MPTrain. The comparison has been made in terms of runners efficacy and enjoyment in achieving predefined workout goals. Results show that TripleBeat is more efficient and more enjoyable than the older application and that the virtual competition and the interface are two key factors to significantly improve the user experience.

Nike + Gps [1] is an mobile application that supports and encourages users during their physical activity, particularly during the walking and the

[1]Nike+ gps. http://nikerunning.nike.com.

running activity. The proposed application makes an intensive use of social networks, like Facebook and Twitter, to share results and to receive real time feedbacks from a user's contacts. Moreover it uses vocal feedbacks to inform the user about his performances. The project also include a web community, where users can organize trainings and share their own training experiences.

In [Jayant and Saponas, 2005] authors present MarioFit, which is an application that aims at taking advantage of the widespread use of video games among young people, to encourage them to do more physical activity. In fact authors propose a new way of gaming, in which users have an active role, by physically interacting with the game. MarioFit allows users to play the Nintendo game Super Mario Bros on a PDA, using the users body movements as inputs. Authors study some research accelerometer and compass data that led to the identification of six human movements to use as input to the game: jumping, ducking, turning, walking, running, and throwing. MarioFit also includes a social factor by means of a web site, where users can compare their Mario scores and performances with those of their friends.

In [Buttussi and Chittaro, 2010] authors present Monster & Gold, which is a context-aware and a user-adaptive game for mobile phones that considers several aspects like heart rate, fitness level, age, etc. to support and motivate users during their outdoor running activity, in order to obtain the best cardiovascular benefits. The proposed application uses Bluetooth pulse oximeter clipped on the user's ear to get pulse data and the mobile phone GPS to get the user's position. Authors finally make two evaluations, by means of some questioner, with two different groups of users, each composed by eight males and six females, in order to infer how the application improve the runners experience.

King et al. [King et al., 2013], starting from the consideration that often adult people conduct a sedentary lifestyle, propose three behavior change mobile apps to promote a regular physical activity and reduce sedentary behavior based on three distinct motivational frames. The first app applied an *analytic* motivational frame, based on social cognitive theory and self regulatory principles of behavior change. The second app considers a *social* frame that is based on social influence theory, while the third app is based on an *affective* motivational frame drawn from emotional transference to an avatar, whose movements and behaviors directly reflects the physical activity and sedentary levels of the user. Finally, another app has been implemented to provide real time feedback to users of all three behavior change apps using algorithms based on the national recommendations for physical activity. The apps have been tested for eight weeks; results show that the apps were useful to increase the average minutes of walking per week and the general levels of physical activity and to decrease the average time spent in front of the television.

In [Hamari and Koivisto, 2013], authors investigate on how social factors can influence the gamification field (which aims to develop services designed to provide game-like experiences to users, commonly with the end-goal of affecting user behavior). In particular the current study considers data from a gamification application for physical activity called Fitocracy. This work takes into account social influence, recognition, reciprocal benefit, network exposure, attitude and intentions to infer how social motivations can predict the use of services that belong to gamification field. The results of the proposed study show that social factors are strong predictors for how gamification is perceived and whether the user intends to continue using the service and/or recommending it to other users.

The work presented in this thesis, concerning with the social motivation in the Human-Computer interaction field, differs from the state-of-the-art works because it favors also the social interactions before a user workout, while the other existing applications allow to share the results of a workout only a the end.

Chapter 3

User Recommendation

3.1 Overview

Social media systems are "web-based services that allow users to (1) construct a public or semi-public profile within a bounded system, (2) articulate a list of other users with whom they share a connection, and (3) view and traverse their list of connections and those made by others within the system" [Boyd and Ellison, 2007]. Moreover, in their 2011 tutorial [Guy and Carmel, 2011], Guy et. al highlight that a social media system is characterized by: (1) a user-centered design, (2) user-generated content (e.g., tags), and (3) social networks and online communities.

Social bookmarking systems are a form of social media that allows users to use keywords (*tags*) to describe resources that are of interest for them, helping to organize and share these resources with other users in the network [Farooq et al., 2007]. The most widely-known example of social bookmarking system is Delicious[1].

These systems are characterized by an ever-growing amount of content and users, that leads to two problems that arise in cascade.

Social interaction overload problem. Social interaction overload [Guy et al., 2013, Simon, 1971] is a problem related to the huge amount of users and items that each user can interact with. This leads to the scarcity of attention, which does not allow to focus on users or items that might be interesting for a user.

In order to filter information and select only the interesting items, in the social media systems domain, in the last few years the research on recommendation has brought to the development of a new class of systems, named *social recommender systems* [Ricci et al., 2011]. These systems allow to face the social interaction overload problem, by suggesting users or items that users might be interested in.

Serendipity/Over-specialization problem. This problem arises from the approaches used by recommender systems which usually suggest items that have a strong match with the user profile; consequently the user always receives recommendations for items too similar to those that she/he already considered and never receives suggestions for unexpected, sur-

[1] http://www.delicious.com

prising and novel items. This limit of recommender systems, known in the literature as "serendipity problem" or "over-specialization problem", worsens the user experience and does not allow the users to explore new items and to improve her/his knowledge [Shani and Gunawardana, 2011]. The serendipity problem affects both the content-based recommender systems [Lops et al., 2011] and the collaborative filtering approaches [Ziegler et al., 2005]. In [Shani and Gunawardana, 2011] authors highlight that:

- in the evaluation of a recommender system, the accuracy is important but it is not enough. So, other metrics have to be considered to evaluate a system;
- users should be able to increase their knowledge and to improve their user experience by discovering new items; so, it is appreciate the introduction of the diversity among recommendations [Lops et al., 2011];
- the design of a recommender system is strongly related to the aspects that have to be evaluated;

Novelty and serendipity are two metrics that are gaining ever more attention in the evaluation of a recommender system. *Novelty* measures how many recommended items the user did not know about, while *serendipity* measures how surprising the successful recommendations are; serendipity can be seen as a way to diversify recommendations and to allow users to discover new items that they did not know they wanted. The main difference between a novel recommendation and a serendipitous recommendation is that a recommendation is novel when the user might have autonomously discovered the recommended item, while a recommendation is serendipitous when the user receives a recommendation that she/he might not have discovered [Lops et al., 2011]. Furthermore, authors assert that the definition of new metrics to evaluate those aspects "constitutes an interesting and important research topic".

Contributions. In this chapter a friend recommender system, which operates in the social bookmarking domain, designed and developed to face the social interaction and serendipity problems is presented.

In these systems, when a user adds another user as a "friend", she/he receives updates anytime a new resource is bookmarked by the friend. Those resources should be novel, i.e. diverse from those already in her/his user profile and should be interesting for the user. At the same time, the accuracy of the friend recommendations is a fundamental property.

Therefore, the proposed system should:

- recommend friends with a high *accuracy*, i.e. users that are proved to be interesting for each other;
- recommend friends whose bookmarks are *novel* and *serendipitous*, i.e. bookmarks related to resources that the target user has not already considered and that are diverse enough from those available in her/his user profile;

The proposed solution is based on *behavioral mining*, i.e., the system analyzes and exploits the user interaction with the content, in order to filter and recommend only the users with the same interests. The proposed form of mining takes into account only the tags and the resources shared by the users, in order to be able to accurately recommend friends, whose bookmarks can be novel and serendipitous for the target user.

The scientific contributions brought by this work are the following:

- for the first time ever a formal definition of a friend recommender system that operates in a social bookmarking system is proposed;
- the first algorithm in literature that recommends friends in the social bookmarking domain is proposed (other approaches in the literature recommend people to follow but, as previously highlighted, this is a different research topic);
- a study about how to mine content in this context, i.e., what information should be used to produce the recommendations and which importance should the different types of content have in the recommender system is presented. This is done by observing the behavior of users in their bookmarking activity.
- since in the literature it is known that there is no universal definition of novelty and serendipity [Lops et al., 2011] and there is no other works that recommend friends in the social bookmarking domain,

a new definition of novelty and serendipity in this context are proposed;

- a set of best practices and a critical discussion of the proposed system are presented, in order to support the research community in the development of a friend recommender system in the considered domain.

The rest of the Chapter is organized as follows: Paragraph 3.2 presents a formalization of a social bookmarking system and of the friend recommendation problem; Paragraph 3.3 describes the details of the friend recommender algorithm presented in this PhD thesis. Starting from an analysis of the user behavior in a social bookmarking system, the design of a friend recommender system is presented with the algorithms that compose it; in Paragraph 3.4 an analysis of novelty and serendipity in their classic definitions is presented and a definition of novel and serendipitous recommendation in the social bookmarking domain is proposed; Paragraph 3.5 illustrates the conducted experiments and outlines main results; Paragraph 3.6 presents a critical discussion of the proposed approach and presents a set of best practices to develop a friend recommender system in the social bookmarking domain; Paragraph 3.7 contains comments, conclusions and future work.

3.2 Friend Recommendation in a Social Bookmarking Systems

This Paragraph gives a formal definition of a social bookmarking system and of a friend recommender system in this domain.

Definition 1 *A social bookmarking system can be defined as a tuple* $Q = \{U, R, T, A, C\}$*, where:*

- U*,* R*, and* T *are sets of* users, resources*, and* tags*;*
- A *is a ternary relation between the sets of* users, resources*, and* tags*, i.e.,* $A \subseteq U \times R \times T$*, whose elements are the tag* assignments *of a user for a resource;*
- C *is a binary relation between the users, i.e.,* $C \subseteq U \times U$*, whose elements expresses the* connection *among two users. The user social relations of a user can be represented by means of a graph, in which each node represents a user* $u \in U$ *and each edge* $c \in C$ *represents a connection among two users; this graph will have an undirected edge if the users are connected as* friends *and a directed edge if one user* follows *the other.*

Definition 2 *A friend recommender system in a social bookmarking is a function* $f : U \times U \rightarrow C$*, which allows to define if, given two users* $u \in U$ *and* $m \in U$*, there is a undirected connection* $c \in C$ *among them.*

In this work an algorithms able learn the function f, which allows to produce recommendations among two users is presented.

3.3 Mining User Behavior to Produce Friend Recommendations

This section presents the friend recommender system developed in this PhD thesis. An analysis of the user behavior in a social bookmarking system (paragraph 3.3.1), which led to the design of the proposed recommender system (paragraph 3.3.2) is presented. In conclusion (paragraph 3.3.3), the algorithms that compose the recommender system, are presented.

3.3.1 User Behavior in a Social Bookmarking System

In the following, an analysis of the user behavior in a social bookmarking system, from a friend recommendation point of view, is presented. In particular, how the bookmarking activity of a user is related to that of the others has been studied by analyzing a Delicious dataset, distributed for the HetRec 2011 workshop [Cantador et al., 2011]. The dataset contains:

- 1867 users;
- 69226 URLs;

- 53388 tags;
- 7668 bi-directional user relations;
- 437593 tag assignments (i.e., tuples [user, tag, URL]);
- 104799 bookmarks (i.e., distinct pairs [user, URL]).

By analyzing user profiles, it emerges that users had an average of 123.697 tags used to bookmark the resources, and an average of 56.132 bookmarked resources.

In order to be able to infer the possible connections among users, which might lead to friend recommendations in this system, the number of common tags and resources between the users of the dataset have been computed, obtaining the following results: the average number of common tags among two users is 7.807, while the average number of common resources among two users is 0.042. In particular, considering only the users who have at least a common tag, the average number of common tags for a couple of users increases to 10.417; while considering only the users who have at least a common bookmarked resource, the average number of common resources for each couple of users increases to 1.673.

From the conducted analysis is possible to infer some properties related to the user behavior in a social bookmarking system, recapped below:

- the behavior of two users in a social bookmarking system is related both by the use of the tags and by the use of the resources;

- the use of tags represents a stronger form of connection (as also proved in literature), with respect to the amount of common resources between two users. This happens because the probability that two users use the same tags is higher than the one to bookmark the same resource, since a user classifies a resource with more tags (so in the system there are more tags than resources) ;
- by comparing the number of common tags and resources with respect to the number of all tags and resources, it emerges that the number of common tags and common resources is much smaller than the number of tags and resources used by each user (more precisely, 10.4 out of 123.7 tags, and 1.7 out of 56.1 resources).

This behavioral analysis has been one of the aspects that characterized the design of the system, which is presented next.

3.3.2 System Design

The goal of the proposed work is to build a friend recommender system in the social bookmarking domain. In its design, the following aspects were considered:

(a) In [Gupta et al., 2013], authors highlight that Twitter is an "interest graph", rather than a "social graph". Authors highlighted that the analysis of such a graph suffers from scalability issues and, in order to contain the complexity of the recommender system, no user profile information could be used to produce the recommendations. Also a social bookmarking systems can be seen as an interest graph, since a user can add as a friend or follow another user, in order to receive her/his newly added bookmarks.

(b) Social media systems grow rapidly. This means that the amount of content added to a social media system and the user population increase at a fast rate. A recommender system that operates in this context needs to build accurate profiles of the users, which have to be up-to-date with the constantly evolving preferences of the users.

(c) The analysis of the user behavior previously conducted showed that both the tags and the resources are a form of connection among two users. In particular, the number of common tags and resources between users is a small subset of all the tags and resources bookmarked by the users.

(d) As [Zhou et al., 2010] states, the tagging activity of the users reflects their interests. Therefore, the tags used by a user are an important source of information to exploit the interests of a user.

Considering the aspects mentioned above, a recommender system that operates in the following way has been designed.

Regarding point (a), in order to avoid the limitations due to the graph analysis in this domain, the proposed system analyzes only the content of the users (i.e., the tagged resources). So, the designed system belongs to the class of recommender systems which analyzes the interactions of the users with the content of the system.

Regarding point (b), in order to efficiently and quickly update the preferences of the users, the system uses algorithms and metrics quickly computable in order to keep the user profiles up-to-date. Therefore, a friend recommender system should mine the *user behavior* (i.e., the interaction of the users with the content), more than the content itself. In fact, the use of metrics like TF-IDF gives a structured form to the resources, but on the other hand they would significantly increase the complexity of the system. Since social bookmarking systems grow rapidly and continuously, content mining would lead to have outdated profiles, so this alternative is discarded in design and architecture of the proposed system;

Regarding point (c), the proposed work is based on the idea that the analysis of users with a similar behavior (i.e., users who have a large amount of common tags and common resources), is a good approach to produce accurate recommendations. Since from the analysis of the users behavior emerged that in a user profile there are many tags and many resources that have not been used by the other users, the produced recommendations lead to novel and serendipitous bookmarks.

Regarding point (d), the theory that user interest is reflected by the tagging activity was embraced; furthermore, this theory led to the intuition that users with similar interests make a similar use of tags and resources.

In the follows, the main steps performed by the system in order to produce recommendations, are presented.

3.3.3 Algorithms

Figure 3.1 illustrates the architecture of the proposed system. The main components of the proposed architecture are:

- Tag-based profile learner;
- Resource-based profile learner;
- Tag-based profile association computation;
- Resource-based profile association computation;
- Filtering component;

Given a target user $u_t \in U$, the system recommends the users with a high tag-based user similarity and a high percentage of common resources. Now the main five steps (each of them performed by a component of the architecture showed in Fig. 3.1) performed by the system are presented:

1. *Tag-based user profiling.* This step, performed by the *Tag-based profile learner component*, builds a user profile based on the tag assignments

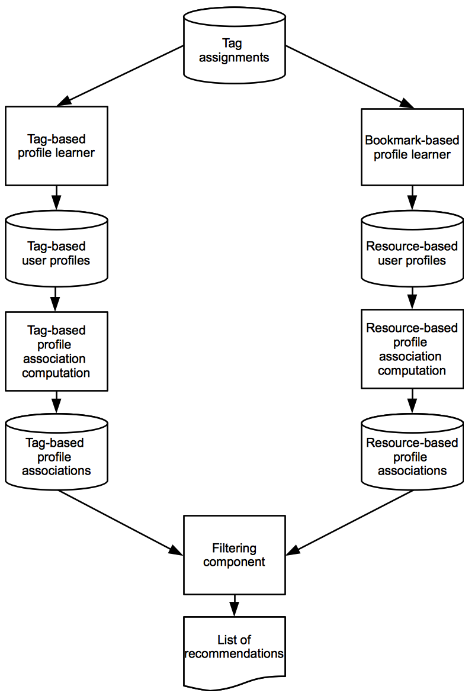

Figure 3.1: Architecture of the friend recommender system

of each user, i.e. by considering the frequencies of the tags used by a user.

2. *Resource-based user profiling.* Given the tag assignments of each user, this step, performed by the *Resource-based profile learner component*, builds a user profile, based on the resources bookmarked by a user.

3. *Tag-based similarity computation.* The first metric, calculated among a target user u_t and the other users, is based on the tag-based user profile. Pearson's correlation is used to derive the similarity.This step is performed by the *Tag-based profile association computation component*.

4. *User interest computation.* The *Resource-based profile association computation component* computes the second metric, i.e. the interest of a user towards another user. This metric is represented by the percentage of common resources among them.

5. *Recommendations selection.* This step, performed by the *filtering component*, recommends to u_t the users with both a tag-based and a user interest higher than a threshold value.

The steps previously presented are recapped in Algorithm 1.

In the following, a detailed description of each step is presented.

Algorithm 1 Friend recommender system

Let $Q = \{U, R, T, A, C\}$ be a social bookmarking system;
Let $U = \{u_i\}_{i=1}^{n}$ be the set of all users;
Let S be the candidate set of users to recommend;
for $i = 1 \dots n$ **do**
    $u = U[i]$ ▷ User 1
    for $j = 1 \dots n$ **do**
        if $U[i]! = U[j]$ **then**
            $m = U[j]$ ▷ User 2
            Let $\overrightarrow{u_1}$ be the Tag-based user profile for the user u;
            Let $\overrightarrow{m_1}$ be the Tag-based user profile for the user m;
            Let $\overrightarrow{u_2}$ be the Resource-based user profile for the user u;
            Let $\overrightarrow{m_2}$ be the Resource-based user profile for the user m;
            $sim = ts(\overrightarrow{u_1}, \overrightarrow{m_1})$ ▷ Eq. 3.3
            $user - interest1 = ui(\overrightarrow{u_2}, \overrightarrow{m_2})$ ▷ Eq. 3.5
            $user - interest2 = ui(\overrightarrow{m_2}, \overrightarrow{u_2})$ ▷ Eq. 3.6
            if $((sim > \alpha)$ && $((user - interest1 > \beta) || (user - interest2 > \beta)))$ **then**
                $S.add(u, m)$ ▷ Eq. 3.7
            end if
        end if
    end for
end for

Tag-based user profiling

This step builds a user profile, based on the tag assignments of a user, considering the frequency of each used tag. Given the sets defined in Section 3.2, the tag assignments of a user u can be considered as follows:

Definition 3 *Let $A(u) \subseteq A$, be the subset of A, whose elements are the triples that contain a user $u \in U$, i.e., $\forall r \in R \wedge \forall t \in T, (u, r, t) \in A \Rightarrow (u, r, t) \in A(u)$.*

Given a tag t, all the resources bookmarked by the user u with the tag t are considered:

Definition 4 *Let $A(u, t) \subseteq A(u)$, be the subset of $A(u)$, whose elements are all the triples that contain a tag $t \in T$ used by a user $u \in U$, i.e., $\forall r \in R, (u, r, t) \in A(u) \Rightarrow (u, r, t) \in A(u, t)$.*

A tag based user profile can be built, according to her/his use of the tags, by considering the relative frequency of each tag, as follows:

$$v_{uj} = \frac{\#A(u, t_j)}{\#A(u)} \tag{3.1}$$

Equation 3.1 estimates the importance of a tag $t_j \in T$ in the profile of a user $u \in U$, by defining the relative frequency as the number of times t_j was used, normalized with respect to the number of tag assignments of u.

A tag-based user profile can be implemented by representing each user $u \in U$ as a vector $\overrightarrow{v_u} = \{v_{u1}, v_{u2}, ..., v_{uk}\}$, where each element v_{uj} is the relative frequency previously defined and k is the number of tags in the system.

Resource-based user profiling

This step builds a resource based user profile, i.e. a profile based on the resources bookmarked by each user. A user can be profiled, according to her/his bookmarked resources, by considering the fact that she/he bookmarked a resource (i.e., she/he expressed interest in it).

Precisely, this profile can be implemented by representing each user $u \in U$ by means of a binary vector $\overrightarrow{v_u} = \{v_{u1}, v_{u2}, ..., v_{un}\}$, which represents the resources tagged by each user. Each element v_{uj} is defined as follows:

$$v_{uj} = \begin{cases} 1 \text{ if } \exists t \in T \mid (u, r_j, t) \in A(u) \\ 0 \text{ otherwise} \end{cases} \tag{3.2}$$

where n is the number of resources in the system. Equation 3.2 estimates the interest of a user u in a resource r_j with a binary value, equal to 1 in case r_j was bookmarked by u, and 0 otherwise.

Tag-based Similarity Computation

Since in [Zhou et al., 2010] authors highlight that the interests of the users are reflected in their tagging activities, the proposed system computes the similarity among two tag-based user profiles with the Pearson's correlation coefficient [Pearson, 1896]. As proved by Breese et al. [Breese et al., 1998], this metric is the most effective for the similarity assessment among users.

Let (u, m) be a pair of users represented respectively by vectors $\overrightarrow{v_u}$ and $\overrightarrow{v_m}$. The recommender algorithm computes the tag-based user similarity ts as defined in Equation 3.3:

$$ts(u, m) = \frac{\sum_{j \subset T_{um}} (v_{uj} - \overline{v}_u)(v_{mj} - \overline{v}_m)}{\sqrt{\sum_{j \subset T_{um}} (v_{uj} - \overline{v}_u)^2} \sqrt{\sum_{j \subset T_{um}} (v_{mj} - \overline{v}_m)^2}} \tag{3.3}$$

where T_{um} represents the set of tags used by both users u and m and values $\overline{v}_u$ and $\overline{v}_m$ represent, respectively, the mean of the frequencies of user u and user m. The tag-based similarity compares the frequencies of all the tags used by the considered users. The similarity values range from 1.0, that indicates complete similarity, to -1.0, that indicates complete dissimilarity. Herlocker et al. [Herlocker et al., 1999] demonstrated that negative similarities are not significant to evaluate the correlation among users, so in the proposed algorithm only positive values are considered.

User interest computation

Given a pair of users (u, m), in this step, two metrics based on the resources tagged by users are computed. The former, $ui(u, m)$, represents the interest of the user u towards user m, while the latter, $ui(m, u)$, represents the interest of the user m toward the user u.

The set of resources bookmarked by each user can be defined as follows:

Definition 5 *Let $R(u) \subseteq R$ be the subset of resources used by a user $u \in U$, i.e., $\forall r \in R, (u, r, t) \in A(u) \Rightarrow r \in R(u)$.*

While the resources in common among two users can be defined as follows:

Definition 6 *Let $D(u, m) = R(u) \cap R(m)$ be the subset of resources bookmarked by both user u and user m.*

Given the above definitions, the *user interest* of a user u in a user m can be estimated as:

$$ui(u, m) = \frac{\#D(u, m)}{\#R(u)} \tag{3.4}$$

As highlight in 3.5, the level of interest of a user u in a user m is estimated as the number of resources bookmarked by both the users, divided by the number of resources bookmarked by user u. This means that the interest of the user m in user u depends on the number of resources bookmarked by m (i.e., when calculating $ui(m, u)$, the denominator would be $\#R(m)$).

User interest ui previously defined, can be implemented, by using the two resource-based user profiles $\overrightarrow{v_u}$ and $\overrightarrow{v_m}$, as follows:

$$ui(u, m) = \frac{\sum_{j=1}^{n} v_{uj} v_{mj}}{\sum_{j=1}^{n} v_{uj}} * 100 \tag{3.5}$$

$$ui(m,u) = \frac{\sum_{j=1}^{n} v_{uj} v_{mj}}{\sum_{j=1}^{n} v_{mj}} * 100 \tag{3.6}$$

where n is the total number of resources of the system.

Recommendations selection

Once the tag-based similarities and the user interests have been computed for each pair of users, the recommender system choses the candidate set $S(u_t)$, i.e. a set of users to recommend to the target user u_t by selecting:

- the ones that have a tag-based user similarity higher than a threshold value α (i.e., $ts > \alpha$);
- the ones that have a user interest (at least one of the two computed) higher than a threshold value β (i.e., $ui > \beta$).

Definition 7 *Given a target user u_t, the candidate set of users to recommend $S(u_t)$ can be defined as*

$$S(u_t) = \{u_i \in U \,|\, ts(u_t, u_i) > \alpha \,\&\&\, (ui(u_t, u_i) > \beta) \,||\, (ui(u_i, u_t) > \beta)\} \tag{3.7}$$

3.4 Novelty and Serendipity in a Friend Recommender System

Novelty and serendipity are two metrics used to evaluate a recommender systems. Precisely, *novelty* measures how many recommendations include items that the user did not know about [Konstan et al., 2006] while *serendipity* measures "how surprising the successful recommendations are" [Shani and Gunawardana, 2011]. Serendipity can be seen as a way to introduce diversification in the recommendation, in order to allow users to discover new items that they did not know they were interested in and to improve their knowledge.

A serendipitous recommendation is, by definition, also novel, while the opposite is not true, i.e. a novel recommendation might not be serendipitous.

As mentioned in the Introduction, the development of new metrics to evaluate novelty and serendipity represents an interesting research topic [Lops et al., 2011]. The recommender system presented in Section 3.3 is the first that operates in the social bookmarking domain, consequently approaches developed in order to evaluate the novelty and the serendipity of a recommender system in such domain do not exist. Here, a definition of novelty and serendipity of the resources bookmarked by the recommended friends is given, and based on those definitions, the metrics that allow to compute the novelty and the serendipity of a friend recommender system are proposed in Section 3.5.

Definition 8 *A resource $r \in R$ can be considered* novel *for a user $u \in U$ iff $r \notin R(u)$. Let $N(u)$ be the set of novel resources for the user u.*

At the state of the art, several studies proved that the serendipity of a resource can be computed by measuring its distance from the items previously considered by the target user [Lops et al., 2011, Shani and Gunawardana, 2011, Iaquinta et al., 2008, Zhang and Hurley, 2008]. As already mentioned, the proposed system is able to produce recommendations by mining user behavior. When a user is recommended as a friend, it is possible to determine if a resource she/he bookmarked is serendipitous for the target user, by computing the distance between the resource and the user behavior. So, the distance between a recommended resource and the resources already bookmarked by the target user is based on the tags used to bookmark the resources.

In order to define the concept of serendipitous resource for a given user, first the set $T(r)$ of tags used for a specific resource r has to be defined:

Definition 9 *Let $T(r) = \{t \in T | \exists (u, r, t) \in A\}$ be the set of tags used for a given resource r.*

Given the above definition, the similarity $sim(r_i, r_j)$ between two resources r_i and r_j can be defined as follows:

$$sim(r_i, r_j) = \frac{\#(T(r_i) \cap T(r_j))}{\#((T(r_i) \cup T(r_j))} \tag{3.8}$$

Where:

- $T(r_i) \cap T(r_j)$ represents the set of common tags used to bookmark the resources r_i and r_j;
- $T(r_i) \cup T(r_j)$ represents the set that contains all tags used to bookmark the resources r_i and r_j;

To better understand the computation of the resource similarity $sim(r_i, r_j)$, each resource r can be represented as a k-dimensional binary vector $t = \{t_1, t_2, ..., t_k\}$, where k is the number of tags used in the system and each value t_i of the vector is computed as follows:

$$t_i = \begin{cases} 1 \text{ if } t_i \in T(r) \\ 0 \text{ otherwise} \end{cases} \tag{3.9}$$

Table 3.4 allows to present an example of how the resource similarity can be calculated. Resource r_i was bookmarked with tags t_1 and t_4, while resource r_j was bookmarked with tags t_2 and t_4. So, $T(r_i) \cap T(r_j) = \{t_4\}$ (the cardinality of the set is 1), $T(r_i) \cup T(r_j) = \{t_1, t_2, t_4\}$ (the cardinality of the set is 3), and $sim(r_i, r_j) = \frac{1}{3}$.

	t_1	t_2	t_3	t_4
$T(r_i)$	1	0	0	1
$T(r_j)$	0	1	0	1

Table 3.1: Example of the vectors used to calculate the resource similarity

Starting from Equation 3.8 a resource $r \in R$ can be defined as serendipitous for a user $u \in U$, as follows:

Definition 10 *A resource $r_i \in R$ can be considered serendipitous for a user $u \in U$ iff $r_i \notin R(u) \wedge \forall r_u \in R(u), sim(r_i, r_u) < 0.5$. Let $B(u)$ be the set of serendipitous resources for the user u.*

3.5 Experimental Framework

This paragraph presents the framework used to perform the experiments. The dataset used and the data preprocessing are first described. Then, the metrics used for the evaluation are presented. The last part of the paragraph presents the experimental setup and the obtained results.

3.5.1 Dataset and pre-processing

Experiments were conducted on a Delicious dataset, distributed for the HetRec 2011 workshop [Cantador et al., 2011], which was presented in the analysis of the user behavior (paragraph 3.3). In particular, now the content of the dataset can be associated to the sets previously defined:

- the set of users is the set U previously defined;
- the set of URLs is the set R previously defined;
- the set of tags is the set T previously defined;
- the set of user relations is the relation C previously defined;
- the set of tag assignments is the relation A previously defined;
- the set of bookmarks is the union of the subsets $R(u)$ previously defined.

The dataset has been pre-processed, in order to remove the users that were considered as "inactive", i.e., the ones that used less than 5 tags or less then 5 URLs.

3.5.2 Metrics

As highlighted throughout all the chapter, the goal of this work was to develop a system whose accuracy was not the only objective that had to be pursued. Now the other metrics used for the performance evaluation of the system are presented.

Precision

In order to measure the accuracy of the system, the usage of the recommendations (i.e., which recommended friends are actually friends with the target user) has been evaluated, by measuring its *precision*.

Definition 11 *Let W be the total amount of recommendations produced by the system, i.e., $W = \cup S(u_t), \forall u_t \in U$. This set represents the positive outcomes, i.e., the sum of the* true positive *and the* false positive *recommendations.*

Definition 12 *Let Z be the amount of correct recommendations produced by the system, i.e., $Z \subseteq W = \{(u,m) | (u,m) \in W \wedge (u,m) \in C\}$. So, Z represents the subset of recommendations for which there is a relation (i.e., a friend correlation) in the dataset. This subset represents the* true positive *recommendations.*

Starting from the previously defined sets W and Z, the *precision* of the recommender system can be computed as the number of correct recommendations, divided by the number of recommendations produced:

$$precision = \frac{true\ positive}{true\ positive\ +\ false\ positive} = \frac{\#Z}{\#W} \tag{3.10}$$

Percentage of Satisfied Users

This metric evaluates the system from a similar (but different) point of view with respect to the precision of the system. In fact, precision measures for how many couples of users a correct recommendation was produced, while the *percentage of satisfied users* measures for how many individual users a correct recommendation was produced.

Definition 13 *Let $X \subseteq U$ be the subset of users for which a recommendation was produced, i.e., $X = \{u \in U | \exists (u, m) \in W\}$*

Definition 14 *Let $Y \subseteq U$ be the subset of users for which a correct recommendation was produced, i.e., $Y = \{u \in U | \exists (u, m) \in Z\}$*

The percentage of users satisfied by the recommendations can be computed by dividing the set of users for which a correct recommendations was produced, i.e. Y, by the set of users for which a recommendation was produced, i.e. X, as follows:

$$\% \, satisfied \ users = \frac{\#Y}{\#X} * 100 \tag{3.11}$$

Novelty and Serendipity

The friend recommender systems presented is based on a mining of the user interests. When a friend recommendation is produced, at the same time also the content of the recommended users is recommended (i.e., their bookmarks).

The novelty for a set of recommendations can be computed as follows:

$$Novelty = \frac{\# \cup N(u)}{\# \cup R(u)}, \forall u \in Y \tag{3.12}$$

As shown in 3.12, the novelty is computed as the sum of novel resources recommended to each user, divided by the sum of resources recommended to each user. Novelty values range from 0 (all the recommended resources were already been considered by target users) to 1 (all the recommended resources were novel).

The serendipity for a set of recommendations, instead, is computed as shown in Equation 3.13.

$$Serendipity = \frac{\# \cup B(u)}{\# \cup R(u)}, \forall u \in Y \tag{3.13}$$

As Equation 3.13 shows, serendipity is computed as the sum of serendipitous recommended resources to each user, divided by the sum of recommended resources to each user. Also serendipity values range from 0 to 1.

To calculate novelty and serendipity only the bookmarks of the recommendations that belong to the set of true positives Z have been considered. In fact, if the novelty of the bookmarks for all the users were computed(no matter is the recommendation was correct of not), they might be new, but also worthless [Shani and Gunawardana, 2011].

3.5.3 Strategy

In the current study three different experiments were performed. The first aims to make an *evaluation of the accuracy*, by computing the precision of the system with different threshold values. The second experiment, makes an *evaluation of the satisfied users* in the produced recommendations, given a precision value. The third experiment is an *evaluation of novelty and serendipity* of the bookmarks shared by the friends.

In order to evaluate the accuracy, the state-of-the-art policy proposed in [Zhou et al., 2010] has been implemented and used as reference system. Zhou et al. [Zhou et al., 2010] implemented a tag-based user recommendation framework and proved that tags are the most effective source of information to produce recommendations. The performances of the presented system with respect to that of the reference one (which uses only tags i.e., $ui = 0$), in terms of precision were compared. Supported by the thesis that the use of only one source of data leads to better performances, a second reference system, which considers only the user interest (i.e., $ts = 0$), was considered.

During the analysis of the performances, all the values of the parameters α and β between 0 and 1, using a 0.1 interval, were evaluated.

3.5.4 Experiments

The details of each performed experiment and its results are now presented.

Evaluation of the Accuracy

Given a target user u_t, the system builds a candidate set, $S(u_t)$, of users to recommend. For each recommended user $u_i \in S(u_t)$, the bi-directional user relations in the dataset (i.e., if $(u_t, u_i) \in C$) have been analyzed, to check if there was a connection between the target user u_t and the recommended user u_i (i.e., if the users are friends). This experiment analyzes the performances of the system in terms of *precision*, given different values of α and β. The main goal of the current experiment is to analyze how the performances of the system vary as the similarity between users grows. The obtained results are illustrated in Fig. 3.2 and Fig. 3.3.

Fig. 3.2 shows the trend of the precision values with respect to the user interest *ui*. The figure contains a line for each possible value α of the tag-based user similarity *ts*. The plot shows that the precision values grow proportionally to the *ui* values. This means that the more similar are the users (both in terms of tag-based similarity and of user interest), the better the system performs. However, for *ui* values higher than 0.5 no user respects the constraints, so no recommendations can be produced. This characteristic confirms the analysis of the user behavior previously done, which highlighted that the amount of common resources among two users is low.

Fig. 3.3 shows the same results from the tag-based user similarity point of view. The figure presents the precision values, with respect to the tag-

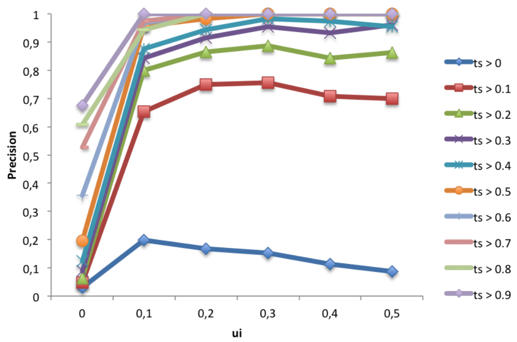

Figure 3.2: Precision of the system with respect to user interest *ui*, for each value of the *ts* user similarity

based user similarity *ts*; In this plot, each line shows the results for a given value β of the user interest *ui*. The obtained results are similar to those of the previously presented graph, i.e. also from this perspective, the precision grows proportionally to *ts*.

The blue lines in Fig. 3.2 and Fig. 3.3 show the results of the reference systems, where $ts = 0$ and $ui = 0$. From the plotted results, it is clear that the two metrics combined improve the quality of the recommendations with respect to the cases where only one is used. These results show that, even if the analysis of the user behavior previously presented highlighted that the average number of resource in common among two users is very small, to consider them through the metric *ui* is important, in order to compute accurate friend recommendations.

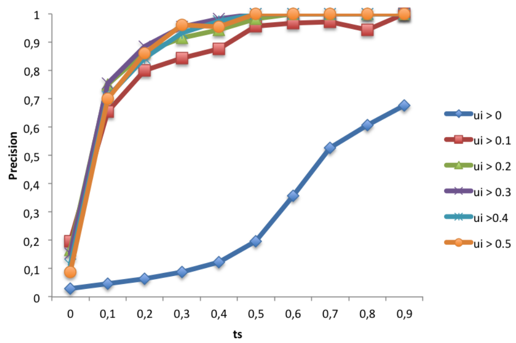

Figure 3.3: Precision of the system with respect to tag-based user similarity *ts*, for each value of the *ui* user similarity

Evaluation of the Satisfied Users

The second experiment aims at analyzing the trend of the satisfied users, with respect to the precision values. So, for each precision value obtained in the previous experiment, the percentage of satisfied users is computed as shown in Equation 3.11.

In order to present the results, Fig. 3.4 reports just a subset of precision values. These values have been selected by dividing the range [0 - 1] of possible precision values into intervals of 0.1 (i.e, [0 - 0.1), [0.1 - 0.2), ..., [0.9 - 1]) and assigning each previously computed value of precision to the right interval. From each interval, the record that corresponds to the precision value that led to the maximum percentage of satisfied users has been selected. The reason why there are no values for the intervals [0.2 -

0.3) and [0.4 - 0.5), is that in the previous experiments there are no values of α and β that led to precision values inside those intervals.

Fig. 3.4 shows that the percentage of satisfied users grows as the precision grows. Given that also the previous experiments highlight that the more similar the users were, the higher the precision was, it is possible to infer that more similar the users are (both in terms of tag-based similarity and of user interest), the higher is the likelihood that users are satisfied by the recommendations.

These results show an interesting property of the presented recommender system. In fact, even if the precision values are split into intervals that cover the same range (i.e., 0.1), there are two of them (i.e., $[0.6 - 0.7)$ and $[0.8 - 0.9)$) in which the percentage of individual users satisfied by the recommendations significantly increases. So, this experiment, by showing the impact of precision on individual users, is very useful when tuning the parameters of the system.

Evaluation of Novelty and Serendipity

This experiment aims to evaluate the novelty and serendipity of the proposed recommender system, by using the metrics previously presented. Also this experiment is conducted on a subset of cases and the evaluation has been done on the intervals previously considered.

Table 3.2 shows the Novelty and Serendipity computed values. Results

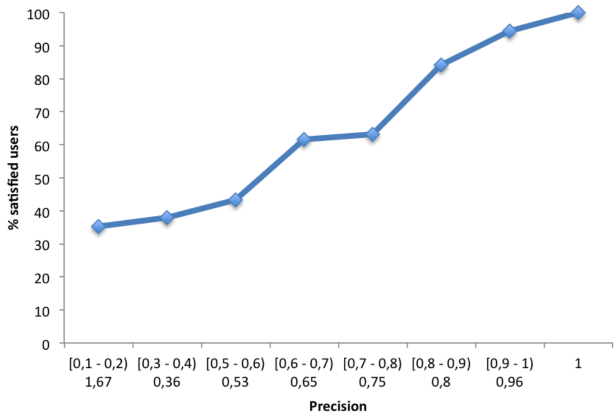

Figure 3.4: Percentage of satisfied users for different values of precision

highlight that both Novelty and Serendipity are inversely proportional to the precision. This means that the number of novel recommended bookmarks and the number of serendipitous recommended bookmarks decrease as the precision of the recommendations grows. However, results show that both novelty and serendipity decrease at a much lower rate, with respect to the increase of the prediction. So, the proposed system is able to produce novel and serendipitous recommendations even when its accuracy is high.

Table 3.2: Novelty and Serendipity

Interval	Precision	Novelty	Serendipity
[0.0 – 0.1)	0.03	0.96	0.92
[0.1 – 0.2)	0.12	0.93	0.81
[0.2 – 0.3)	-	-	-
[0.3 – 0.4)	0.36	0.90	0.65
[0.4 – 0.5)	-	-	-
[0.5 – 0.6)	0.53	0,89	0,54
[0.6 – 0.7)	0.65	0.83	0,69
[0.7 – 0.8)	0.75	0.74	0.59
[0.8 – 0.9)	0.88	0.79	0.61
[0.9 – 1.0)	0.97	0.79	0.53
[1.0]	1.00	0.67	0.47

3.6 Discussion

Now, a summary of the main results related to the proposed system is given, in the form of a set of best practices aimed at a researcher or a software designer involved in real world scenarios where friend recommendations have to be produced in the social bookmarking domain.

Some questions arise when approaching the design of the system:

- given a social bookmarking system, composed by tagged resources and a social network/interest graph that connects the users, which source(s) of information should be exploited when producing the recommendations?

- if content is exploited, what type of mining should be done on it?
- how can a system be designed to produce recommendations that are accurate but also novel and serendipitous?
- is there a source of information which is not useful when building the recommendations?

Some general answers, coming from the results of the experiments, are the following:

- The state-of-the-art highlighted that the mining of the interest graph leads to scalability issues (see Sections 2 and 3.3.2). Therefore, friend recommendations in the discussed domain should be built only by analyzing which resources each user bookmarked and with which tags. The experiment presented in 3.5.4 confirms that the mining of the resources and the tags leads to accurate friend recommendations;
- In the social bookmarking domain it is important to perform a mining of the *behavior*. It is known that social bookmarking systems grow continuously and at a fast rate. So, in order to quickly update user preferences (hence, the user profile) and follow the interests of the

users, a friend recommender system that operates in this domain has to analyze *how* the users bookmarks the resources (i.e., with which tags and with which frequency), instead of analyzing its content, which would strongly increase the complexity of the recommender system. The results reported in 3.5.4 and 3.5.4 report that this type of mining is strongly effective when producing friend recommendations in this context.

- In order to produce friend recommendations that lead to novel and serendipitous resources for the user, a system should be able to infer similarities among users, but also endorse, in the filtering, the users who have a subset of bookmarked resources and tags who are *diverse* from those considered by the user. Therefore, a friend recommender system that operates in this context should mine the common content among the users, without adopting notions of similarity among tags or resources. In other words, a mining that recommends a user if its contents or tags are *similar* to those of the target user should be avoided, in order to select only the users with a similar bookmarking behavior. Since in the behavioral analysis conduced in 3.3.1 has been highlighted that the amount of common tags and resources is relatively small, but the experiments confirm that accurate recommendation can be produced, the tags and resources not in common with the other users leave space for the recommendation of novel and serendipitous resources.

- In order to produce accurate recommendations, both a behavioral mining of the use of tags and resources is necessary. Though the proposed behavioral analysis (see 3.3.1) showed that the amount of resources in common is very small and the proposed system design (see 3.3.2) highlighted that in the literature only tags are considered, the reported experiments confirm that both tags and resources represent important sources of information when producing friend recommendations in this domain (see 3.5.4 and 3.5.4).

There would be some cases that worth considering:

- It is known that graph mining might lead to complexity and scalability issues in this context, but it might be useful when a user is new in the system so she/he has a little amount of resources and tags in her/his profile. Since the proposed system works with common resources and tags, it presents limitations in this case.

- If the resources and tags used by a user are different from those use by the others, she/he might not receive recommendations. In other words, the diversity exploited by the proposed system to produce novel and serendipitous recommendations might become a limitation, if it is not also associated to resources in common with the other users. Also in this context, graph mining might be useful to produce recommendations.

3.7 Contributions and Future Work

This chapter presented a system able to produce friend recommendations by performing a behavioral mining of the users in a social bookmarking system. Precisely, the proposed system considers the frequency of the tags and which resources each user bookmarked, in order to select only the users with similar profiles. The analysis of the user behavior highlighted that the amount of common tags and resources between two users is limited with respect to the amount of tags and resources bookmarked by each user. The characteristic that, given a user, a large amount of resources was not considered by the others, allows to design and implement a friend recommender system able to suggest friends with a high accuracy and that allowed users to come across novel and serendipitous bookmarks.

Furthermore, in the literature it is known that the definition of metrics to evaluate novelty and serendipity in a recommender system is an open research problem. In this chapter, new metrics that could be applied to considered application domain and to the behavioral mining used to build the recommendations, were proposed.

The reported experiments evaluated the accuracy in terms of *precision* and results highlighted the capability of the system to build recommendations with an increasing accuracy, as the similarity among users grows. Moreover, an evaluation of the capability of the system to suggest friends whose bookmarks are *novel* and *serendipitous* is presented and results highlight that even when a system achieves a high accuracy, it is still able to producing novel and serendipitous recommendations.

Future work will be focused on adding a graph mining component to the proposed system, in order to be able to produce recommendations also in the previously highlighted cases, in which users cannot receive recommendations.

Chapter 4

Tag Recommendation

4.1 Overview

The development of the Web 2.0 led to a quick growth in the amount of data available online and also changed the way people face the WWW. In fact users have become active, uploading and sharing content of any type. The huge amounts of data can create some difficulties to classical techniques to categorize and index data, so researchers realized that it may be useful to support classical systems with Collaborative Tagging Systems.

These systems are web based applications that allow a community of users to assign keywords (*tags*) to a given resource. Tagging does not require specific skills, so these systems had a rapid diffusion.

Nowadays, several social media systems are developed as tagging systems. Famous examples include Del.icio.us[1], Flickr[2], Last.fm3[3], CitULike[4].

Users use tags for different tasks: search, navigation, resource classification and serendipitous browsing obtaining a immediate benefit. As highlighted in the Chapter 1, Collaborative Tagging systems can be affected by some linguistic and semantic limitations like synonymy and polysemy (same term with different meanings). In order to limit these problems, most Collaborative Tagging System include a Tag Recommender System (TRS) [Guy and Carmel, 2011] that aims to help users finding appropriate tags, both during the search activity, in order to enhance the chances to find a given resource, and during the tagging activity in order to limit problems due to the freedom in the use of tags and consolidate the vocabulary across users.

The Tag Recommendation problem can be formally described as follows: let u be a user of the system and let r be a resource, the goal is to predict the set T_{ur} of tags that u will assign to resource r [Jäschke et al., 2012].

In order to simplify the tag space visualization, many TRSs build clusters of related tags. Recently, several approaches have been proposed to cluster tags. When a user puts a resource in a tagging system, an association between the resource and the tags used to classify that resource is created. If two tags are used to classify the same resource (tags co-occurrence), an

[1]http://delicious.com/
[2]http://www.flickr.com
[3]http://last.fm
[4]http://www.citeulike.org

association between those two tags is created. Tags associations are used to cluster together all the related tags in the tagging system. Other works in the literature do not exploit the potential source of information coming from monitoring users search activity performed inside the tagging system. Therefore, associations between tags and resources are static, since they are created only when resources are uploaded. Consequently, if a resource is associated to a misleading tag, this misleading association would affect the performances of the system.

This chapter presents RATC (Robust Automated Tag Clustering), an extension of the approach described in [Boratto et al., 2009].

Differently from the previous work, the contribution to the social recommender systems domain and the results of a new set of experiments, that aims at analyze the structure of each cluster, are presented.

RATC exploits the user behavior, by monitoring the user activity in the search engine of a tagging system, in order to infer implicit feedbacks provided by users. Every time that a user finds a relevant resource during his search activity, a feedback is collected and used to dynamically update associations between resources and tags. Then, from the tag-resources associations, the system is able to infer tag-tag associations, by means of a standard correlation measure, and those associations are used to build clusters of strongly related tags. This clusters may be used in order to recommend *novel* tags to the user in different phases: when a user put a new resource into a social web application to help her/him classifying that resource, or during the search activity to facilitate the retrieval of the searched resource. The results obtained have been compared with those of a classic tag clustering approach [Begelman et al., 2006] and show an improvement in the capability to cluster strongly related tags. The proposed tag recommender system, which operates in social environments, brings several contributions with respect to the state-of-the-art systems both to the tag clustering research area and to the tag recommender systems research area. As already mentioned it is able to update tag-resources associations, and then tag-tag associations, monitoring the user activity, promoting in this way the real semantic relations among tags and penalizing the misleading ones. In literature no work in the tag clustering area monitors the user behavior in order to update associations between tags. The ability to solve the misleading resource classification problem, make the proposed system "robust". In fact, as defined in [O'Mahony et al., 2004], Robustness is the capability of an algorithm to remain stable in presence of fake information, usually specifically added to influence its quality.

About the social recommender systems area, the existing systems, that operate with tags, do not use clustering to produce the recommendations and this is a limitation. In fact, in a social web scenario, where everything evolves very quickly and the amount of content grows continuously, a form of unsupervised clustering is a very simple and strongly effective technique to produce associations between similar items. Moreover, the proposed system produces recommendations without using neither the user profile nor the content of the resource, so it is not affected by the well known *Cold Start Problem*, i.e., if a user is new or a resource is not similar to any of the existing resources, no tag can be recommended to the user.

4.2 Method

As previously mentioned, the proposed approach performs a clustering of tags in order to produce recommendations. In the following, the methods adopted for the cluster task and then the methods adopted in the recommendation task are presented.

The four main steps performed by RATC can be recapped as follows:

- **Tag-Resource association creation:** When a new resource is uploaded in the system, an association between the resource and the tags used to classify it, is created.

- **Dynamic Tag-Resource association evaluation:** The activity of the user in the system is monitored and exploited, in order to update the existing *Tag-Resource* associations and to create the new ones.

- **Tag-Tag association creation and quantification:** *Tag-Resource* associations are used to build *tag-tag* associations. Then, a similarity between tags is computed by means of the cosine similarity and the result of this step is a weighted graph, named *tag similarity graph*, in which the nodes represent tags and the edges represent the similarity between tags.

- **Clustering:** The algorithm proposed by [van Dongen, 2000] is used to infer clusters of tags.

4.2.1 Tag-Resource association creation

A tagging system can be defined as a web application that allow users to classify resources by means of keywords called tags. A tagging system is represented as a graph composed by:

- a set T of tags t;
- a set R of resources r;
- a set $E : (T \times R)$ of weighted edges that represent the Tag-Resource associations. The weight of an edge is proportional to the number of times that a given tag has been used to classify a given resource.

When a user uploads a resource in the Tagging System, she/he classifies it with tags, so tag-resource associations are created.

4.2.2 Dynamic Tag-Resource association evaluation

When a user performs a search in a classical tagging system, this one usually retrieves a ranked list of resources, related to the search tags, based

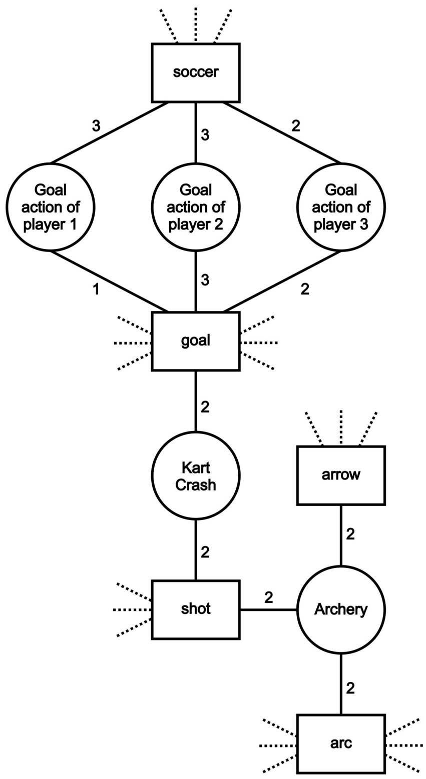

Figure 4.1: An example of a tagging system

on tag-resources associations. A limit of many state of the art tagging systems is that they create Tag-Resource associations only on upload time, i.e., when the resources are uploaded in the system, while updating this type of information can improve the performances of the system. In this work an algorithm that exploits the user feedbacks, in order to differentiate correct associations from noisy associations is adopted. Precisely, each Tag-Resource association has a counter (a weight) that indicates how many times a tag has been used to classify a resource; when a user, after a tag-search task, selects a given resource, the counter of the considered Tag-Resource association is incremented. Naturally, several resources may be related to the same tag; the strength of the associations between tags and resources depends on the behavior of the users. With this approaches all the noisy associations between misleading tags and resources will receive a poor attention. All Tags-Resources associations (i.e., the weights) can be represented in a matrix $W = \{w_{rt}\}$, where w_{rt} is the weight between the resource r and the tag t.

For example, given a tagging system like that represented in Figure 4.1, where rectangular nodes represent tags and round nodes represent resources and the connections between rectangular nodes and round nodes represent the weighted edges, the respective matrix W will be like that represented in Figure 4.2. Given the represented matrix, suppose that a user performs a tag based search operation using the tag “shot”; the system retrieves as search results the resources “Kart crash” and “Archery”. At

	Goal action of player 1	Goal action of player 2	Goal action of player 3	Kart Crash	Archery
soccer	3	3	2	0	0
goal	1	3	2	2	0
shot	0	0	0	2	2
arrow	0	0	0	0	2
arc	0	0	0	0	2

Figure 4.2: Tag-Resource Matrix W

this moment, if the user selected the "Archery" resource, the correspondent association would be updated to the value 3.

4.2.3 Tag-Tag association creation and quantification

Let v_i be the vector that represents the associations among a tag i and its related resources (represented by a row of the table in Figure 4.2) and v_j the vector of associations among a tag j and its related resources. The similarity S_{ij} between the tag i and the tag j can be computed by means of the cosine similarity:

$$s_{ij} = cos(v_i, v_j) = \frac{v_i \cdot v_j}{||v_i|| \times ||v_j||}$$

The similarities among tags are then represented with a directed weighted graph, called *tag similarity graph*. An example of this graph based on the previous tagging system and Tag-Resource Matrix examples (Figure 4.1 and Figure 4.2) is reported in Figure 4.3 (note that the values of the associations in the figure have been calculated considering the whole tagging system).

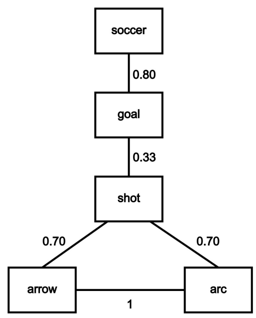

Figure 4.3: Tag similarities graph

4.2.4 Clustering

The tag clustering task adopts the MCL (Markow Clustering algorithm) algorithm [van Dongen, 2000], a community detection algorithm that considers the similarity between vertex in a simple graph, in order to build clusters. The basic idea of the algorithm is that the longest path between nodes that belong to the same cluster is relatively short, while the longest path between nodes that belong to different clusters is high. Consequently, using a random walk algorithm a movement intra-cluster is more difficult with respect to a movement inter cluster. The MCL algorithm runs a random walk through the graph so, if a hypothetical random walker is in a node i at time t, the algorithm selects the node j where the random walker will be at moment $t + 1$, by computing a probability proportional to the weight of the the edge between node i and node j. The random walk algorithm computes the probabilities using two operators, named *expansion*, which computes the power of the matrix, and *inflation*, which computes the Hadamard-Schur product of the matrix combined with a diagonal scaling. The two operators are subsequently applied by the algorithm, which converges quadratically in the neighborhood of doubly idempotent stochastic matrices, i.e., matrices that do not change under the action of the two operators. The result of the algorithm is a matrix that represents a disconnected graph, in which each component contains nodes that belong to the same cluster. The obtained clusters can be see to perform tag recommendation.

4.2.5 Novelty and Serendipity in Tag Recommendation

As already mentioned, the main goal of a tag recommender system is to recommend *keywords*, named tags, to users in order to help and support them both during the classification of the resources and during the search activity. Given the nature of the discussed domain, it is important to recommend novel tags, i.e. tags that the user has not already considered in the classification of a given resource. On the other hand, it is as much important to recommend tags that are strongly related to those already used by the users; in fact a not related tag recommended in the classification process could create misleading tag-resource associations, while a not related tag recommended during the search activity could drastically decrease the odds to find the desired resource. So, in this kind of systems it is essential to have novel recommendations but it is also necessary to avoid serendipitous recommendations, i.e. recommendations for tags unrelated to the ones used by the user.

The tag clustering allows to reach the goal mentioned above. In fact, the objective of the clustering is to maximize the Inter-cluster distances between items and to minimize the Intra-cluster distances between items [Tan et al., 2005]. So, when a user u uses a tag t, the system suggests to u some tags t_i that are in the same cluster of the tag t; in this way the system is able to recommend novel tags that are definitely related to those used by the user. In conclusion all tags recommended by the proposed system are novel, given that it does not consider tags already used by the target user, and are related to those used by the user so those tags are not serendipitous.

4.3 Experiments

In order to evaluate the performance of the algorithm, a tagging system, presented in [Carta et al., 2008] (that also includes a search engine), has been used to make a comparison with the classical tag clustering approach presented in [Begelman et al., 2006], named *ATC*. The main objectives were to evaluate the *quality of the clustering*, i.e. to evaluate the capability of RATC (ATC) to produce significant clusters of the tags, and to make an *analysis of the clusters*, i.e. to evaluate how correlated are the tags in each cluster.

4.3.1 Dataset collection and Pre-processing

10 volunteers whose main goals were to perform two main steps were recruited. The performed steps are the population of the tagging system (*resources acquisition step*) and the performing of search operations in the tagging system (*feedback collection step*).

Resources acquisition step

During the first step, volunteers were asked to choose some videos from YouTube [5], that belong to the "sport" concept domain, and put them into the tagging system. Each video had to be classified with at least four pertinent tags and with two unrelated (noisy) tags. Such noise is useful to simulate the classical noise of non-sperimental system, in order to study

[5] http://www.youtube.com

how the correlation between noisy tags and resource decreases during the system activity, to observe how the structure of the clusters changes and to evaluate the quality of the clusters.

At the end of this step the tagging system contained 406 videos, 1021 tags and 2597 video-tag correlations. The set of tags was pre-processed in order to remove all tags that express feelings and emotions (it does not make sense to cluster and/or recommend tags like "good", "beautiful", etc.); after the pre-processing the set of tags contained 964 tags.

Feedback collection step

In the second phase of the dataset collection, which started only when the *Resources acquisition step* was completed, the volunteers were asked to perform some search operations in the tagging system. This step allows *RACT* to monitor the user behavior and to improve its performances. When a user performs a search operation, the system retrieves a list of resources (videos) with their original description, so that the user can select the video. More in detail, each volunteer performed 300 search operations, by entering a list of tags as query, and then selecting the most related video from the retrieved list, providing, in this way, a feedback to the system. Each time a user selects a video from the retrieved list the system updates the tag-resources counters.

4.3.2 Strategy and Evaluation Metrics

As already mentioned, in order to evaluate the approach a comparison with a state of the art approach, named *ATC*, proposed by Begelman et al. [Begelman et al., 2006] was made. The main goal of ATC is to build clusters of tags to improve the user experience in the use of a tagging system and to minimize classical linguistic limitations. The work, which has been tested on Delicious [6], uses an algorithm that allows them to find strongly related tags by counting the number of tag co-occurrences (tags used for the same page) used for a page (a URL on Delicious) and defining a cut-off point, to establish when a counter makes sense. The co-occurences between tags are then represented in a sparse matrix, in which each element is a similarity between two tags. Then, a graph based on these similarities is built. Finally, tags are clustered by means of a graph clustering algorithm, based on the spectral bisection. Authors measure the quality of the clustering with the "modularity function" [Newman and Girvan, 2004], which measures the quality of a particular clustering of nodes in a graph. The main steps performed by ATC can be recapped as follows:

- Take as input the connected undirected graph of tag similarities.
- Use spectral bisection to split the similarities graph into two clusters.
- Compare the value Q_0 of the modularity function of the original

[6] https://delicious.com

graph with the value Q_1 of the modularity function of the partitioned graph. If $Q_1 > Q_0$ the partitioning is accepted otherwise it is rejected.

- Repeat recursively the previous described steps on each accepted partition.

Experiments were performed to study how RACT improve its performances by monitoring the activity of the users; for this reason the "state" of the tagging system (i.e., the current values of each tag-resource association) was saved every 50 feedbacks, obtaining 6 different sessions that allow to compare the RACT and the ATC tag clusterings. After the first session, a set of tags that could have been suggested to the user was already available, but it was decided to not do so because, as already mentioned, an objective of the proposed work was to evaluate how RATC would perform in presence of noise, without having the results biased by the fact that the tags collected were suggested by the system itself. The inflation parameter (introduced in 4.2.4), needed for the clustering algorithm, had been set to 3.0. In order to analyze the influence of the noise in the performances of the algorithms, the experiments were repeated also adding noisy tags to the tagging system.

In the following the two sets of experiments performed to evaluate the *quality of the clustering* and to make an *analysis of the clusters* are described.

Quality of the Clustering

The objective of the first set of experiments was to compare the capability of RATC and ATC to create significant clusters of tags. First of all, a domain engineer clustered the involved tags, by grouping together those that refer to the same topic. Then, the clusters created by the domain engineer were compared with those created by the RATC and ATC algorithms. Each cluster of RATC and ATC has been compared with the cluster created by the domain engineer that contained the highest number of corresponding tags. So, let T be the cluster of tags created with RATC or ATC and let D be the cluster of tags created by the domain engineer, the following sets were defined:

- *True positive tags (TP)* $= T \bigcap D = \{t_i | t_i \in T \bigwedge t_i \in D\}$ is the set of tags that belong both to the set T and to the set D;
- *True negative tags (TN)* $= \{t_i | t_i \notin T \bigwedge t_i \notin D\}$ is the set of tags that do not belong neither to the set T nor to the set D;
- *False positive tags (FP)* $= \{t_i | t_i \in T \bigwedge t_i \notin D\}$ is the set of tags that belong to the set T but do not belong to the set D;
- *False negative tags (FN)* $= \{t_i | t_i \notin T \bigwedge t_i \in D\}$ is the set of tags that do not belong to the set T but appear in the set D;

To evaluate the clustering algorithms, some classical information retrieval metrics were used: the *micro- and macro-averaging precision* and *micro- and macro-averaging recall* [Sebastiani, 2002]:

- *Microaveraging* precision and recall are obtained by summing over all individual values:

$$\pi^{\mu} = \frac{TP}{TP + FP} = \frac{\sum_{i=1}^{m} TP_i}{\sum_{i=1}^{m}(TP_i + FP_i)}; \quad \rho^{\mu} = \frac{TP}{TP + FN} = \frac{\sum_{i=1}^{m} TP_i}{\sum_{i=1}^{m}(TP_i + FN_i)} \tag{4.1}$$

  where the "μ" superscript stands for microaveraging.

- *Macroaveraging* precision and recall are first evaluated "locally" for each category, and then "globally", by averaging over the results of the different categories:

$$\pi^{M} = \frac{\sum_{i=1} m\pi_i}{m}; \quad \rho^{M} = \frac{\sum_{i=1} m\rho_i}{m} \tag{4.2}$$

  where the "M" superscript stands for macroaveraging.

Analysis of the clusters

In this set of experiments, the clusters created by the domain engineer were not considered, but a comparison between the clusters created by RACT and those created by ATC was made. Precisely, the engineer, identified a topic for each cluster and computed the percentage of meaningful tags belonging to the cluster itself. For example, let C be a cluster built by RATC or ATC for which the domain engineer identify the topic "basketball"; let $M \subseteq C$ the set of meaningful tags (that belong to the topic basketball), the percentage of meaningful tags MT was calculated as:

$$MT = \frac{\#M}{\#C} * 100 \tag{4.3}$$

Where $\#M$ and $\#C$ represent respectively the cardinality of the sets M and C.

4.4 Results

In this section the results of the previously presented experiments are presented.

4.4.1 Quality of the clustering

In order to evaluate the quality of the clustering, a comparison between between the clusters created by RATC, ATC and the domain engineering has been done. The clustering of RATC and ATC differs from the one performed by the domain engineer for the number of created clusters and

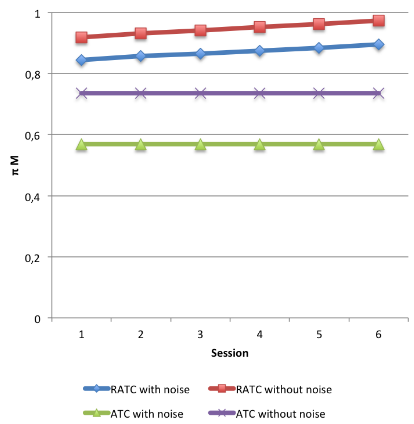

Figure 4.4: Macro-averaging precision

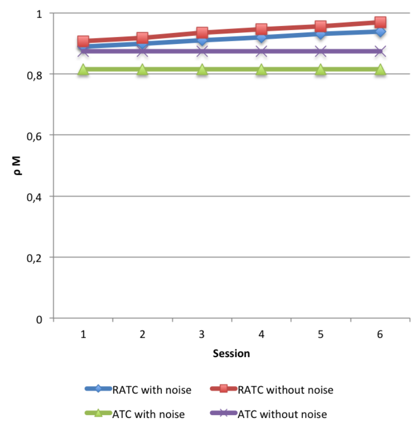

Figure 4.5: Macro-averaging recall

for the number of elements in each cluster. In fact, the number of clusters obtained with the two algorithms was higher than the number of clusters created by the domain engineer. For example, at the end of the last session, RATC without noise involved 266 clusters versus 148 clusters created by the domain engineer.

Fig. 4.4 and Fig. 4.5 present the comparison between the clustering of RATC and ATC algorithms in terms of macro-averaging precision and macro-averaging recall, while Fig. 4.6 and Fig. 4.7 show the results of the tested algorithms in terms of micro-averaging precision and recall.

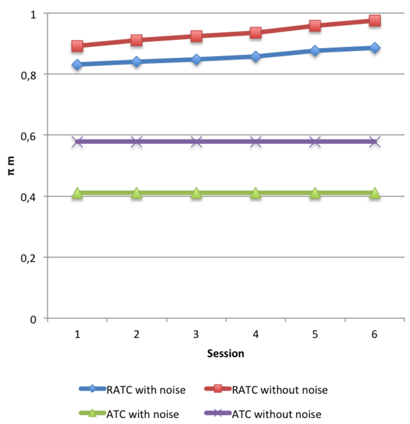

Figure 4.6: Micro-averaging precision

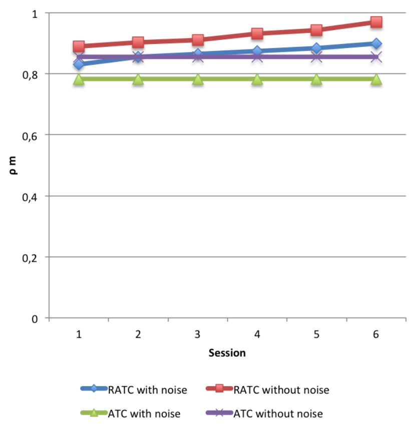

Figure 4.7: Micro-averaging recall

These plots consider both cases, with and without noise. The results show that RATC always perform better with respect to ATC. Furthermore, plots highlight how RATC improve its performances session by sessions; this is due to the capability of RATC to update tag-resources and tag-tag associations by learning from the users feedbacks.

Fig. 4.8 shows an example of clusters created by the two algorithms and by the domain engineer. In this case the topic of the clusters is the olympic games of the 1988 in which Mike Powell won the gold medal in the 100 meter race because Ben Johnson was disqualified doping reasons. These clusters highlight that in the set created by ATC there are some unrelated

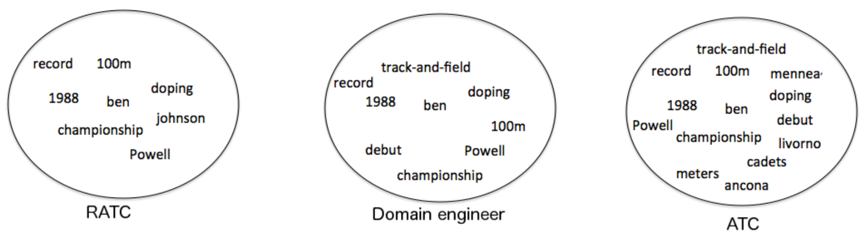

Figure 4.8: An example of cluster created respectively by the RATC, domain engineer and ATC

tags, like "mennea" for example, that lead ATC to create weak associations that not allow the clustering algorithm to create good partitions of tags.

4.4.2 Analysis of the clusters

In Fig. 4.9 a comparison between the percentage of meaningful tags achieved by the proposed algorithm with respect to those achieved by ATC algorithm, using the dataset without noise, is presented. As highlighted in the plot, RATC is able to reach a higher percentage of meaningful tags and, furthermore, it is able to improve its performances session by session by updating the tag-resource associations and tag-tag associations.

In the first session, the tag-resource associations have the same values

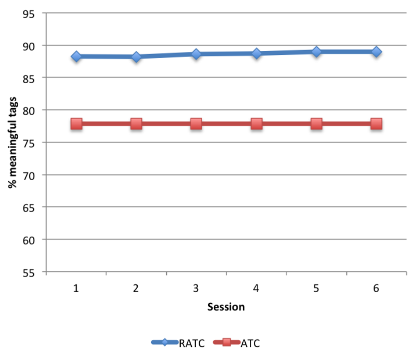

Figure 4.9: Meaningful tags in the dataset without noisy tags

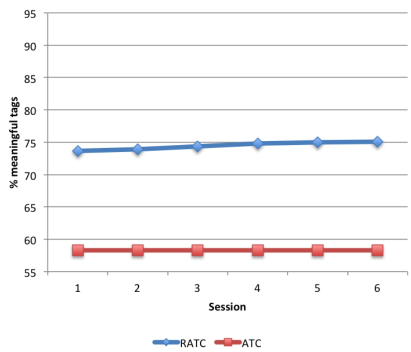

Figure 4.10: Meaningful tags in the dataset without noisy tags

for both the approaches, as no search activity was done in the system, but RATC get better results. So, it means that cosine similarity represents a better metric to measure associations between tags.

Also Fig. 4.10 shows the percentage of meaningful tags achieved by the two algorithms, but in this case the dataset with noise has been considered. The obtained results are similar to those previously presented; in fact also in this case the approach proposed in this work achieves better results and its performances improve session by session.

4.5 Contribution

This chapter presented RATC, an approach able to cluster tags in a tagging system, which can be used to produce tag recommendations that facilitate the exploration of a tagging system. Furthermore, RATC has the capability to dynamically improve its performances by monitoring users behavior and exploiting implicit feedbacks left by users during their search activity.

RATC brings several contributions both to the tag clustering research area and to the social recommender systems that recommend tags. In fact, the proposed approach is able to dynamically update tag-resource associations and tag-tag-associations in order to limit misleading semantic relations. None of the existing works in tag clustering is able to dynamically update associations between tags during the system activity and none of the existing social recommender systems that operate with tags uses clustering to produce the recommendations.

Moreover, thanks to the lack of super visioning during the classification step, this approach lends itself very well to operate in a real web scenario where everything evolves very quickly.

Chapter 5

Social Media Motivation

5.1 Overview

One of the several aspects that lead to a rapid growth of social media systems is that they are among the most persuasive technologies [Fogg and Iizawa, 2008, Weiksner et al., 2008]. In fact usually the behavior of a user in a social network can motivate other users to adopt the same or similar behavior. This phenomenon, known as "Social Influence", occurs when the opinions and behavior of a person influences those of the other ones.

So, starting from the main topic of this thesis, i.e. the social web, some interest was also put in the motivational domain, performing some studies, in order to infer how to motivate people to do something by using social media tools. In fact, motivating one person to do something can be seen as a form of suggestion. Therefore the motivation domain intersects with the recommendation domain presented in previous chapters.

In order to study the motivational aspect in the social environment, some Human-Computer Interaction (HCI) applications to support and motivate users to do more physical activity were developed. This work was followed by the design and implementation of a web application which allows users to interact with Facebook, in order to study how the social aspect can influence the motivational one. To be more precise, two Android applications, based on a persuasive technology, that aims at motivate people to practice running activity have been developed: *EverywhereRun* [Mulas et al., 2011, Mulas et al., 2013a], that allows users to get a workout plan from a personal trainer and *EverywhereRace* that allows users to create virtual competitions (*races*) with people anywhere in the world. These two application will be soon merged in a unique application.

The reason why this type of applications are gaining always more importance is that many people are facing health's problem due to a sedentary lifestyle. In fact, it is known that a sedentary lifestyle is the cause of several serious illnesses like obesity, diabetes, hypertension and so on. The main reason for which people do not practice any physical activity are motivational lack, time constrains, difficulties to start, gym membership fees, equipment costs and so on.

The idea behind this study is that HCI and Social web can benefit each other; for example, a Human-Computer Interaction application could be developed in the Social Web scenario, in order to study and improve relationships among people. Furthermore, many researches demonstrate that the inclusion of social interactions in HCI applications motivate people to practice more physical activity [Consolvo et al., 2006, Virzi, 1992, Buttussi et al., 2006, de Oliveira and Oliver, 2008].

Like other computer science domains, recently, Human-Computer Interaction (HCI) has had an exponential and rapid growth. In [ACM SIGCHI, 1992], authors define Human-Computer Interaction as "a discipline concerned with the design, evaluation and implementation of interactive computing systems for human use and with the study of major phenomena surrounding them".

In this chapter a web application, that implements some race management features (e.g., the creation, the subscription or the participation to a race), previously available only in the Android mobile application is presented. Particularly, in the web application some new innovative social features by means of the Facebook social network are implemented, in order to allow users to share their workout experience with their friends.

The social aspect, through the interaction with Facebook, allows to enhance the social engagement that, as previously mentioned, is a key aspect in a motivation scenario. In fact, the interaction with a social network leads to the phenomenon of the "social influence" (a widely known concept in sociology and viral marketing) [Cha et al., 2010], for which the enhancements of a user in her/his exercising activity can inspire and motivate other users to improve their performances. So, the availability of the features to manage the races on more devices and the capability to interact with a social network, should improve the motivation to exercise regularly. The choice to develop a web application that focuses on the organization of races, was made because a race involves more than a user, so this scenario lends itself well to link Human-Computer Interaction with the Social Web.

The contribution brought by the study presented in this chapter concern both to the Human-Computer Interaction and the Social Web research areas, precisely:

- exploit virtual races to motivate people;
- the use of a web application, which introduces some functionalities of the already existing Android application, allows to manage the races not only from small devices, like mobile phones and tablets. This simplifies the access to the functionalities and improves the user experience, in order to motivate people to organize more races and to exercise more;
- using the web application, users can create new races and challenge their friends in real time;
- by mining the behavior of the users with respect to the Facebook social network, it is possible to study how social media can act on the motivational aspect of the users.

In the following, the two developed Android applications (as already stated they soon will become one unique application that will include all the above characteristics) are presented, then the web application is described, by presenting its architecture and functionalities and finally conclusions and future work are discussed.

5.2 Improving motivation to practice physical activity

In current paragraph the two Android applications, developed in order to help and motivate people to practice more physical activity and to practice it in a better way, are presented.

5.2.1 Everywhere Run!

Everywhere Run![1] [Mulas et al., 2011, Mulas et al., 2013a] is an Android mobile application that aims to support people during their running routines. In fact, through this application, users can design their own regimes or get a tailored ones directly from a real personal trainer inside the application.

Figure 5.1 presents the screen used by users to create a workout, which allows users to plan relatively complex workouts, (for example, the figure shows a workout named "Monday"). A training is organized into "sessions", called "traits", which are described by a distance and a pace to keep. A user that wants to follow the "Trait 1" represented in Fig. 5.1, has

[1]http://www.everywhererun.com/

Figure 5.1: Workout creation menu

Figure 5.2: Personal trainer screen

to run 2km following a speed of 5 minutes per kilometer. While the "Trait 1" of the same figure indicates to the user that she/he has to run 10km at a pace of 4:20 minutes. The main aspect of the application is the definition of "Virtual personal trainer". In fact, this features, based on workout settings, support the runner during all workout, motivating him to speed up or to slow down, in order to reach the predefined goal. This characteristic is represented in Figure 5.2, which represents an ongoing workout where the runner has to speed up to reach the predefined objectives. Precisely Figure 5.2 is composed by 3 main parts: an horizontal bar, which contains an overview of the whole workout and information about the position of the user with respect to that of the virtual personal trainer, a dashboard

Figure 5.3: User workouts statistics

containing information about the speeds, times, etc, and the other part contains a graphic representation of the delay or the lead that the user has with respect to the virtual personal trainer.

Figure 5.3 represents a dashboard that recaps some statics about all user workouts. In particular, that figure shows that since the 1st november 2012 the considered user ran 350 km in 30 hours and, furthermore, information about the fastest workout, the slowest workout and the peace average are reported.

From the preliminary experiments conducted on *EverywhereRun* the application led to an improvement on the motivational aspect of the runners.

More details on the presented approach and on the experiments, have been described in [Mulas et al., 2011, Mulas et al., 2013a] and are presented in appendix A.

5.2.2 Motivation in Races

The other Android application that it is going to be described is *Everywhere Race!*[2] [Mulas et al., 2012]. This application introduces the concept of "virtual competition", based on social interaction. In fact, by means of the application it is possible to create and to perform a virtual competition with other users that are any where the world. Competition can concern running, cycling and any other sport that take into account the concept of "speed".

In Fig. 5.4, when the application starts, it presents a main menu, which allows the user to create a new race, to perform a search (based on some parameters like sport, distance, etc.) of an existing race or to perform a search based on the friends subscription to the races. If a user is enrolled for a given race, this screen shows always the remaining time to that race.

[2]http://www.everywhererace.com/

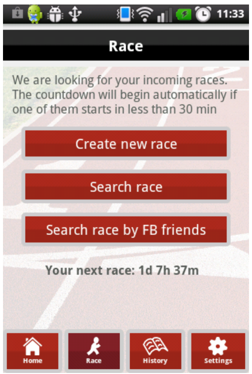

Figure 5.4: Main menu of Everywhere Race!

In the showed example, the next race of the considered user will be after one day and seven hour more or less.

Fig. 5.5 and Fig. 5.6 show how the application allows the user to monitor her/his position with respect to those of his/her opponents and other statistical data, during a race and at the end of the race.

The conducted experiments highlight the strengths of the social aspect to motivate people during races.

More details on this approach and the experiments are reported in [Mulas et al., 2012] and are presented in appendix A.

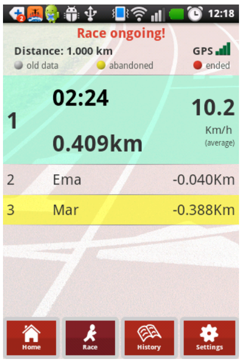

Figure 5.5: On-going race

Figure 5.6: Race results

5.3 A Web Application to Support Social Interaction

The web application discussed in this paragraph, at the moment works with *Everywhere Race!* but soon it will work with the unique application that will include both *Everywhere Run* and *Everywhere Race!*. This web application introduces several advantages, like all functionalities already available in the mobile version, plus the possibility to interact with the Facebook social network.

5.3.1 Architecture of the Web Application

Fig. 5.7 shows the architecture of the entire project, which includes both the web application and the Android application. The web application is

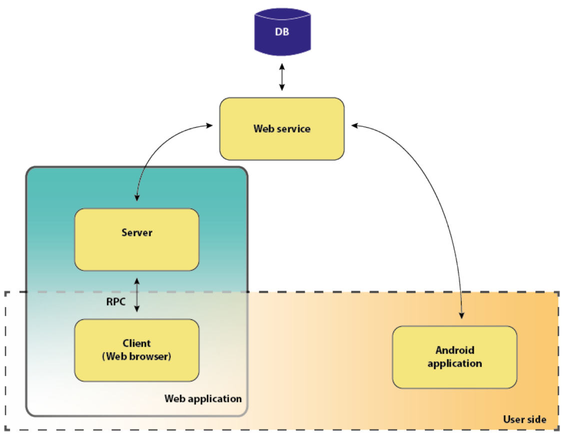

Figure 5.7: Architecture of the project

now described in detail.

As represented in the figure, the application is a client-server application. The user, at the client side (by means of a browser) makes some requests to the server through a remote procedure call (RPC). A request may have different goals like create a race, show existing races etc. The server elaborates the request and, depending on the request, uses a web service to store data in a database, to read data from it, etc.

5.3.2 Features Offered by the Web Application

The proposed web application is composed by three main sections; each of them appears each time that the "Races button", the "Friends button" or the “My Diary button” are clicked (see Fig. 5.8). In the following, the content of each screen is described:

- *Races screen*: the dashboard recaps data that belong to all the users of the application;
- *Friends screen*: the dashboard shows the data concerning the Facebook friends of the logged user;
- *My Diary screen*: the dashboard shows all data of the logged user, allowing her/him to have a summary of his racing activity.

Moreover, in the top right part of Fig. 5.8 there are several social buttons developed by AddThis[3], which is a free service that allows to interact with several social networks and a Facebook button to execute a Facebook login and logout.

In addition to the previously described dashboard, the proposed web application offers other functionalities represented in Fig. 5.9 and Fig. 5.10. Figure 5.9 shows a feature that allows users to search a race, by means of several parameters like sports, date, etc. As a result of a search task, a list of races and related status (i.e., ongoing, finished or future) is retrieved. The application, in addition, allows the users to publish a Facebook post regarding each race; this functionality is very important for the presented study and its details will be presented later. This screen allows also the

[3]http://support.addthis.com

Figure 5.8: Top of the web application (Dashboard and login features)

user to create a new race by filling a form. Each race is composed of several fields like date, start time, distance, sport name, maximum number of participants, race, place, description, etc.

Each race is associated to a specific URL, which allows to show the detail of a race in specific screen (see Fig. 5.10). The details of a race are composed by three main parts:

- *Details*, which contains all the data of a race (i.e., name, description, date, place and distance) and the list of Facebook friends that already joined the race;
- *Rank*, which contains data about ongoing and finished races. Precisely, it contains the list of participants to a race and its related data (i.e., some personal data, position, etc.).

Figure 5.9: The list of races shown by the web application

- *Graphic*, shows a plot which represents the evolution of the race.

5.3.3 Social Interaction

During the development of the web application particular focus was put on the social aspect, since the interaction with the social media domain (precisely with the Facebook social network in this work) leads to the phenomenon of the “social influence” (a widely known concept in sociology and viral marketing) in the social network domain [Cha et al., 2010]. Through these phenomena, the performances of a user in her/his exercising activity can inspire and motivate other users to improve their own. Furthermore, by sharing the results of a race, a user can receive suggestions or feedbacks from her/his friends through Facebook’s comments and

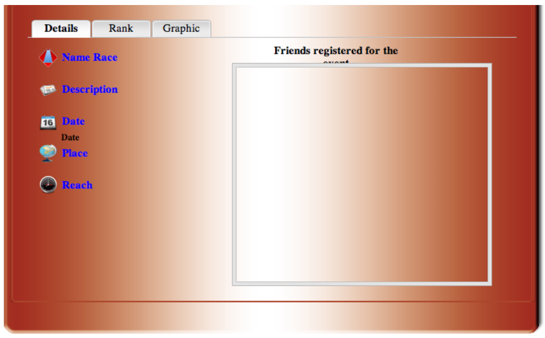

Figure 5.10: The details of a race shown by the web application

likes, which can motivate to do best and to exercise regularly. In order to implement what was just mentioned, the web application gives the user the possibility to publish posts on Facebook. The type of post that the user can share depends on the state of the race (ongoing, finished or future). Table 5.1 presents an example of the six possible Facebook posts, while Fig. 5.11 shows an example of the dialog window of a post.

5.4 Experimental Results

Some experiments about the usability and the social influence of the systems are now presented.

About the usability test, a standard System Usability Scale (SUS) questionnaire [Brooke, 1996] has been submitted in order to investigate how

Table 5.1: Facebook posts

	Participant	Not participant
Finished	I just finished this race: Final position: 1 Distance: 10km Date: 06/01/2014 Sport: Running Time: 1:30:00 Medium speed: 10km/h Medium pace:10km/h	This race is finished: Distance: 10km Date: 06/01/2014 Name: Charity Marathon Description: Charity Marathon
Ongoing	I'm participating to this race: Temporary position: 1 Distance: 10km Date: 13/08/2013 Sport: Running Time: 1:30:00 Medium speed: 10km/h Medium pace:10km/h	This race is ongoing: Distance: 10km Date: 13/08/2013 Name: Charity Marathon Description: Charity Marathon
Future	I'll participate this the race: Distance: 10km Date: 06/01/2014 Name: Charity Marathon Description: Charity Marathon	There will be this race: Distance: 10km Date: 06/01/2014 Name: Charity Marathon Description: Charity Marathon

Figure 5.11: Dialog window of a Facebook post

users evaluate the overall usability of the system. 20 volunteer users (6 males and 14 females with an average age of 24.5 years and standard deviation of 5.56 years), that used EWRace supported by the web platform for at least one month, were recruited.

The evaluation of the SUS questionnaires returns a mean value of 76.25 with standard deviation equal to 14.50. In order to obtain a more meaningful estimate for the mean value, a 95% t-confidence interval has been computed, obtaining the following estimates: 69.47 and 83.35 [Sampaio, 2013]. In this way, there is a confidence of 95% that the real score is between 69.47 and 83.35. Even if the value of estimate 69.47 is considered, the result is a percentile rank of 53%; this means that the proposed web platform is more usable than 53% of products in [Sauro, 2011].

In order to evaluate the social aspect of the proposed system, the average number of races created per week over a period of eight weeks has been monitored. During the first four weeks the users did not have the web application available and each user created an average of 1.4 races per week. The average number of races created per week growth to 1.8 (+28%) when allowing the users to use the web application. Moreover, users were asked to evaluate, with a scale that ranges from 0 to 5, how much their performances have been influenced by the social pressure provided by the real time competitions. The obtained final result has been of 3.7 out of 5. In conclusion, the users evaluation suggests that the preliminary design of the platform has an acceptable usability. Moreover, the results also highlight how social aspect is a key element to support users in their physical activity.

5.5 Conclusions

In the current chapter a web application that include the functionalities of *Everywhere Race!* and extends it, by allowing users to interact also with a Facebook social network is presented. In fact, in the last years the way people face the information drastically changed, due to the introduction of social medias with the web 2.0; moreover, is known that social networks are among the most persuasive technologies. These changes are the reasons that motivated to strongly consider the social aspect in the design of a novel web application. The co-operation of the social web domain and the HCI domain allowed to study how the social interaction can motivate the user and change his habits. In fact, preliminary studies highlight how the interaction of the users with Facebook led to a greater participation of the users to the system improving also the user motivation to exercise.

Chapter 6

Conclusions and future works

This PhD Thesis focuses on social media and social recommender systems. It has been discussed like the ever growing amount of data available in the web 2.0 applications might decrease the users attention leading to the well known "social interaction overload" problem. To face this problem several social recommender systems have been studied at the-state-of-the-art. In this work some limitations of the existing social recommender systems have been highlighted: the analysis of a social graph, in order to produce recommendations, suffers from scalability issues and, in order to limit the complexity of the recommender system, no user profile information could be used to build the recommendations [Gupta et al., 2013]; another limitation is that many social recommender systems focus on produce accurate recommendations without considering that it is not enough to guarantee a good user experience. In fact, often, social recommender systems suffer form "over specialization" (or "serendipity problem") problem, i.e. the recommended items are too similar to those already considered by the target user which never receives suggestions for unexpected, surprising and novel items [Ziegler et al., 2005]. The mentioned limitations highlight the importance to take into account also other metrics in the evaluation of a recommender system (and not only accuracy), like novelty and serendipity. In this work two social recommender systems have been presented: a friend recommender system and a tag recommender system (named RATC). In order to overcome the limitations mentioned above, these systems do not use the social graph but make a selective use of the available information in order to produce accurate, novel and serendipitous recommendations. This PhD thesis brings several contribution with respect to the state-of-the-art works. The presented tag recommender system is the first, in the tag clustering domain, that monitors the user activity in order to solve the misleading tags problem. Furthermore, this system uses a unsupervised clustering (a very strong technique to produce associations between similar items in a very dynamic domain) without using the user profile, to produce accurate and, at the same time, novel tag recommendations. RATC, since it does not use neither the user profile nor the content of the resource, is able to overwhelm also the cold start problem.

The friend recommender system, presented in Chapter 3 is the first friend recommender system designed in the social bookmarking domain.

It is able to produce accurate recommendations, just exploiting the users content without considering the social graph. Furthermore experimental results show that the system produces recommendations that are not only accurate but, at the same time novel and serendipitous, allowing users to improve their knowledge.

The tag system presented in Chapter 4 appears in:

- [Boratto et al., 2013] Ludovico Boratto, Salvatore Carta, Matteo Manca, Fabrizio Mulas, Paolo Pilloni, G Michele Pinna, Eloisa Vargiu **A Clustering Approach for Tag Recommendation in Social Environments** presented in International Journal of E-Business Development.

The friend recommender system described in Chapter 3 appears in:

- [Manca et al., 2013] Matteo Manca, Ludovico Boratto, Salvatore Carta **Producing Friend Recommendations in a Social Bookmarking System by Mining Users Content**, presented in IMMM 2013, The Third International Conference on Advances in Information Mining and Management (**best paper award**);
- [Manca et al., 2014b] Matteo Manca, Ludovico Boratto, Salvatore Carta **Design and Architecture of a Friend Recommender System in the Social Bookmarking Domain**, Science and Information Conference 2014 SAI2014;
- [Manca et al., 2014c] Matteo Manca, Ludovico Boratto, Salvatore Carta

Mining User Behavior in a Social Bookmarking System: a Delicious Friend Recommender System, submitted to 8th International Conference on Autonomous Infrastructure, Management and Security AIMS 2014;

- [Manca et al., 2014a] Matteo Manca, Ludovico Boratto, Salvatore Carta **Behavioral Mining to Produce Novel and Serendipitous Friend Recommendations in a Social Bookmarking System: a Delicious Case-Study**, submitted to Performance Evaluation Journal.

Another aspect that has been considered in this work is how the social aspect can be used as persuasive technology in order to encourage users to adopt specific behaviors. To this purpose two Android applications, that aim at support and encourage users to do more physical activity, have been developed: *EverywhereRun*, presented in [Mulas et al., 2011, Mulas et al., 2013a], and *Everywhere Race!*[1], presented in [Mulas et al., 2012]. Once these two mobile applications have been developed, a web application, that implements some race managements and some new innovative social features by means of the Facebook social network, has been designed. By means of this web application it has been studied as the social aspect through the "social influence" can inspire and motivate other users to improve their performances. The results of the conducted study highlighted that the social aspect is a very useful tool also in the Human-Computer

[1] http://www.everywhererace.com/

interaction domain to encourage and support people to adopt given behaviors. The study presented in Chapter 5 appears in:

- [Mulas et al., 2013c] Fabrizio Mulas, Paolo Pilloni, Matteo Manca, Ludovico Boratto, Salvatore Carta **Linking Human-Computer Interaction with the Social Web: a Web Application to Improve Motivation in the Exercising Activity of Users** presented in CogInfoCom 2013 - 4th IEEE International Conference on Cognitive Infocommunications;

- [Mulas et al., 2013b] Fabrizio Mulas, Paolo Pilloni, Matteo Manca, Ludovico Boratto, Salvatore Carta **Using New Communication Technologies and Social Media Interaction to Improve the Motivation of Users to Exercise** presented in FGCT 2013 - 2nd International Conference on Future Generation Communication Technologies.

Moreover, the contributions introduced by *Everywhere Run* mobile application (before the development of the web application) appear in:

- [Mulas et al., 2011] Mulas Fabrizio, Carta Salvatore, Pilloni Paolo and Manca Matteo **Everywhere run: a virtual personal trainer for supporting people in their running activity**. presented in ACE11 - 8th International Conference on Advances in Computer Entertainment Technology

Future works will focus on evaluating the proposed recommender systems also with other datasets and in different domains. Furthermore, also other mentioned metrics, like *Trust* and *Persuasiveness* will be tested in the evaluation of the recommender systems.

Moreover, regarding the presented friend recommender, future work will focus on the implementation of a graph mining component to use in those cases in which users cannot receive recommendations for the previously mentioned reasons (for example when the user has a small amount of used tags and resources).

Appendix A

Experiments on the Android Mobile applications

Before introducing the social features, some experiments that allowed to test the capability of the applications introduced in Chapter 5, i.e. *Everywhere Run* and *Everywhere Race*, to motivate users had to be conducted. This appendix presents the experiments and the results obtained.

A.1 Everywhere Run: Experimental Results

In order to evaluate the application capabilities, a survey to a group of ten runners that tested *Everywhere Run* was submitted. The group of runners was composed of five males and five females with an average age of about

28.3 and only four users practiced physical activity regularly. Users were asked to rate the application, with regard to some features, with a rate that ranged from 0, meaning "strongly disagree", to a maximum of 5, meaning "strongly agree". Figure A.1 shows the evaluated features and the obtained results. The average rating obtained by the application has been 3.8; in fact users stated that *Everywhere Run* had been a very useful tool to support their workouts. Particularly, users appreciated audio cues because they consider them more handier with respect to visual advise. The rate given by users for the motivational aspect reaches 3.8, they stated that performing a workout with predefined goals helps to reach them.

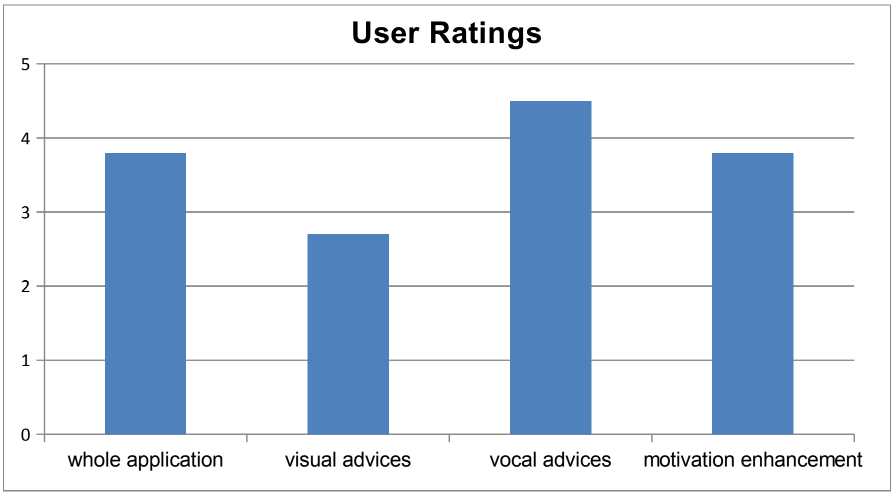

Figure A.1: User ratings

Usability Tests

Another considered aspect is the software usability. In order to test the application usability, five users are sufficient [Nielsen and Landauer, 1993] [Virzi, 1992]. So, in order to test the usability of the *Everywhere Run* five users, aged between 20 and 35, were enrolled. Two of these users were expert runners while others were just occasionally joggers and all of them had some experience in the daily use of smartphones.

The users were shown the application without giving them any explanation about its usage and then, users were asked to use some features in order to create a workout, modify it, etc.. The interactions of the users with the application allow to fix some trouble. The testers evaluate positively the application, in fact using the same scale adopted in the previous experiment the application scored an average rating of 3.8.

The inexperienced users face some difficulties in the use of the application, for example during the work out creation. This troubles were due to the unit of measurement, in fact, they did not know that runners usually indicate the speed as time to run one kilometer (or mile) so they create a incorrect workout. The two expert runners, instead, did not face any problem. In order to help inexperienced users to overcome the problems due to the unit of measurement the km/h or mi/h measurement has be set as the default one.

A.2 Everywhere Race: Experimental Results

During the development of the presented web application, Mulas et al. performed some experiments in order to evaluate the Android application proposed in [Mulas et al., 2012]. In the following the performed experiments and the results obtained by Mulas et al are reported.

In order to evaluate the effectiveness of the proposed Android applications, 35 volunteers were asked to use Everywhere Race! for 30 days and to perform a test. The group of volunteers users were composed of 25 male and 10 female aged between 19 and 40 and ten users regularly practiced sport.

Among the ten active users, four exercised about four times a week, while the others from two to one times a week and the average time of each session was 30 minutes. Twenty seven users had never used before any application during their physical activity, while the others already did. The goal of the evaluation test was to investigate about the influence of the proposed application on users motivation in order to both validate current application features and improve the application for future developments.

In order to evaluate *EveryWhere Race!* the Exercise Motivations Inventory - 2 (EMI-2), developed by Markland et al. [Markland and Ingledew, 1997] has been chosen. EMI-2 consists of 51 items that belong to14 scales. Users are asked to rate each item on a five-point scale ranging from 0 (“not at all true for me”) to 5 (“very true for me”). In order to obtain the scale scores the means for each item belonging to the appropriate scale are computed. Figure A.2 shows the obtained results.

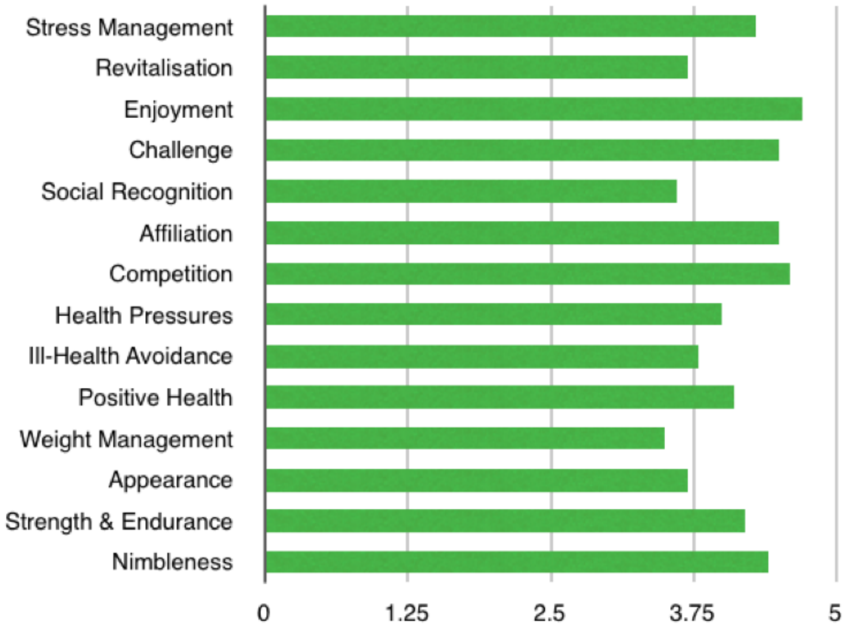

Figure A.2: EMI-2 Results

Figure A.2 shows all the scales received good scores so, the application is a valid tool to help people to start working out. Users were submitted also another questionnaire in order to better investigate the effects of the application on users sport habits. Also in this case, testers were asked to rate using the same scale (ranging from 0 to 5) used to rate the EMI-2 items. The proposed questions were:

1. "Did EWR help you to improve performances?"
2. "Were social features important to improve your performances?"
3. "Did EWR change your sport habits?"
4. "Will you continue to use EWR in the future?"

Preliminary results showed in Fig. A.3 highlight that users think that the proposed application is a valid tool to help users to reach predefined sport goals and to increase the motivation.

The average results put in evidence that the majority of users perceive the application as a valid tool that helps to achieve sport goals in a more enjoyable and regular manner. As it is possible to see, despite the limited sample of test users, encouraging preliminary results have been obtained. The positive trend emerging from the tests shows that the application may help to increase motivational factors through this new engaging and social way of active gaming.

A.3 Conclusions

The experiments previously presented allowed to validate the capability of both *Everywhere Run* and *Everywhere Race* to motivate users. Thanks to the results previously presented, the social features could be added, in order to analyze the impact of the interactions with a social network in the motivation of a user.

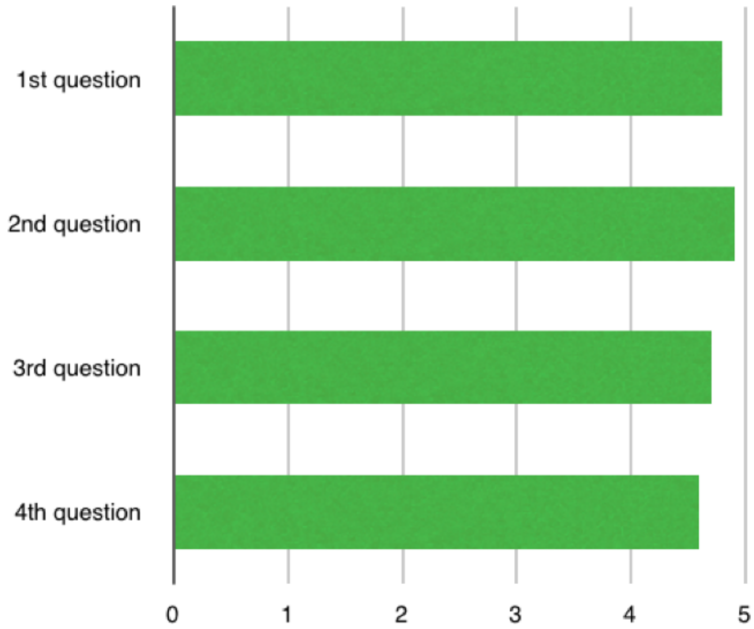

Figure A.3: Questionnaire Results

Bibliography

[Abbassi et al., 2009] Abbassi, Z., Amer-Yahia, S., Lakshmanan, L. V., Vassilvitskii, S., and Yu, C. (2009). Getting recommender systems to think outside the box. In *Proceedings of the Third ACM Conference on Recommender Systems*, RecSys '09, pages 285–288, New York, NY, USA. ACM.

[ACM SIGCHI, 1992] ACM SIGCHI (1992). Acm sigchi curricula for human-computer interaction.

[Appelquist et al., 2010] Appelquist, D., Brickley, D., Carvahlo, M., Iannella, R., Passant, A., Perey, C., and Story, H. (2010). A standards-based, open and privacy-aware social web. w3c incubator group report.

[Baeza-Yates, 2005] Baeza-Yates, R. (2005). Applications of web query mining. In *Proceedings of the 27th European conference on Advances in Information Retrieval Research*, ECIR'05, pages 7–22, Berlin, Heidelberg. Springer-Verlag.

[Baeza-Yates and Ribeiro-Neto, 1999] Baeza-Yates, R. A. and Ribeiro-Neto, B. (1999). *Modern Information Retrieval*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

[Begelman et al., 2006] Begelman, G., Keller, P., and Smadja, F. (2006). Automated tag clustering: Improving search and exploration in the tag space. In *Proceedings of the Collaborative Web Tagging Workshop at the WWW 2006*, Edinburgh, Scotland.

[Bielenberg, 2005] Bielenberg, K. (2005). Groups in social software: Utilizing tagging to integrate individual contexts for social navigation.

[Boratto et al., 2013] Boratto, L., Carta, S., Manca, M., Mulas, F., Pilloni, P., Pinna, G. M., and Vargiu, E. (2013). A clustering approach for tag recommendation in social environments. *International Journal of E-Business Development*, 3(3):126–136.

[Boratto et al., 2009] Boratto, L., Carta, S., and Vargiu, E. (2009). Ratc: A robust automated tag clustering technique. In Noia, T. D. and Buccafurri, F., editors, *EC-Web*, volume 5692 of *Lecture Notes in Computer Science*, pages 324–335. Springer.

[Boyd and Ellison, 2007] Boyd, D. M. and Ellison, N. B. (2007). Social network sites: Definition, history, and scholarship. *J. Computer-Mediated Communication*, 13(1):210–230.

[Breese et al., 1998] Breese, J. S., Heckerman, D., and Kadie, C. (1998). Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence*, UAI'98, pages 43–52, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

[Brin and Page, 1998] Brin, S. and Page, L. (1998). The anatomy of a large-scale hypertextual web search engine. *Comput. Netw. ISDN Syst.*, 30(1-7):107–117.

[Brooke, 1996] Brooke, J. (1996). Sus: A quick and dirty usability scale. In Jordan, P. W., Weerdmeester, B., Thomas, A., and Mclelland, I. L., editors, *Usability evaluation in industry*. Taylor and Francis, London.

[Brzozowski and Romero, 2011] Brzozowski, M. J. and Romero, D. M. (2011). Who should i follow? recommending people in directed social networks. In Adamic, L. A., Baeza-Yates, R. A., and Counts, S., editors, *ICWSM*. The AAAI Press.

[Burke, 2007] Burke, R. (2007). The adaptive web, methods and strategies of web personalization. In Brusilovsky, P., Kobsa, A., and Nejdl, W., editors, *The Adaptive Web*, chapter Hybrid web recommender systems, pages 377–408. Springer-Verlag, Berlin, Heidelberg.

[Buttussi and Chittaro, 2010] Buttussi, F. and Chittaro, L. (2010). Smarter phones for healthier lifestyles: An adaptive fitness game. *IEEE Pervasive Computing*, 9(4):51–57.

[Buttussi et al., 2006] Buttussi, F., Chittaro, L., and Nadalutti, D. (2006). Bringing mobile guides and fitness activities together: a solution based on an embodied virtual trainer. In *Proceedings of the 8th conference on Human-computer interaction with mobile devices and services*, MobileHCI '06, pages 29–36, New York, NY, USA. ACM.

[Cantador et al., 2011] Cantador, I., Brusilovsky, P., and Kuflik, T. (2011). Second workshop on information heterogeneity and fusion in recommender systems (hetrec2011). In Mobasher, B., Burke, R. D., Jannach, D., and Adomavicius, G., editors, *Proceedings of the 2011 ACM Conference on Recommender Systems, RecSys 2011, Chicago, IL, USA, October 23-27, 2011*, pages 387–388. ACM.

[Carmel et al., 2010] Carmel, D., Roitman, H., and Yom-Tov, E. (2010). Social bookmark weighting for search and recommendation. *The VLDB Journal*, 19(6):761–775.

[Carta et al., 2008] Carta, S., Alimonda, A., Clemente, M., and Agelli, M. (2008). Glue: improving tag-based contents retrieval exploiting implicit user feedback. In Hoenkamp, E., de Cock, M., and Hoste, V., editors, *Proceedings Of The 8th Dutch-Belgian Information Retrieval Workshop (DIR 2008)*, pages 29–35. isbn: 978-90-5681-282-9 idxproject: ?

[Cha et al., 2010] Cha, M., Haddadi, H., Benevenuto, F., and Gummadi, P. K. (2010). Measuring user influence in twitter: The million follower fallacy. In Cohen, W. W. and Gosling, S., editors, *Proceedings of the Fourth International Conference on Weblogs and Social Media, ICWSM 2010, Washington, DC, USA, May 23-26, 2010*. The AAAI Press.

[Chen et al., 2009] Chen, J., Geyer, W., Dugan, C., Muller, M., and Guy, I. (2009). Make new friends, but keep the old: recommending people on social networking sites. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '09, pages 201–210, New York, NY, USA. ACM.

[Consolvo et al., 2006] Consolvo, S., Everitt, K., Smith, I., and Landay, J. A. (2006). Design requirements for technologies that encourage physical activity. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '06, pages 457–466, New York, NY, USA. ACM.

[de Oliveira and Oliver, 2008] de Oliveira, R. and Oliver, N. (2008). Triplebeat: enhancing exercise performance with persuasion. In *Proceedings of the 10th international conference on Human computer interaction with mobile devices and services*, MobileHCI '08, pages 255–264, New York, NY, USA. ACM.

[Farooq et al., 2007] Farooq, U., Kannampallil, T. G., Song, Y., Ganoe, C. H., Carroll, J. M., and Giles, C. L. (2007). Evaluating tagging behavior in social bookmarking systems: metrics and design heuristics. In Gross, T. and Inkpen, K., editors, *Proceedings of the 2007 International ACM SIGGROUP Conference on Supporting Group Work, GROUP 2007, Sanibel Island, Florida, USA, November 4-7, 2007*, pages 351–360. ACM.

[Fogg and Iizawa, 2008] Fogg, B. J. and Iizawa, D. (2008). Online persuasion in facebook and mixi: A cross-cultural comparison. In Oinas-Kukkonen, H., Hasle, P. F. V., Harjumaa, M., Segerståhl, K., and Øhrstrøm, P., editors, *PERSUASIVE*, volume 5033 of *Lecture Notes in Computer Science*, pages 35–46. Springer.

[Giannakidou et al., 2008] Giannakidou, E., Koutsonikola, V., Vakali, A., and Kompatsiaris, Y. (2008). Co-clustering tags and social data sources. *Web-Age Information Management, International Conference on*, 0:317–324.

[Givon and Lavrenko, 2009] Givon, S. and Lavrenko, V. (2009). Large scale book annotation with social tags. In Adar, E., Hurst, M., Finin, T., Glance, N. S., Nicolov, N., and Tseng, B. L., editors, *ICWSM*. The AAAI Press.

[Gupta et al., 2013] Gupta, P., Goel, A., Lin, J., Sharma, A., Wang, D., and Zadeh, R. (2013). Wtf: the who to follow service at twitter. In Schwabe, D., Almeida, V. A. F., Glaser, H., Baeza-Yates, R. A., and Moon, S., editors, *22nd International World Wide Web Conference, WWW '13, Rio de*

Janeiro, Brazil, May 13-17, 2013, pages 505–514. International World Wide Web Conferences Steering Committee / ACM.

[Guy and Carmel, 2011] Guy, I. and Carmel, D. (2011). Social recommender systems. In Srinivasan, S., Ramamritham, K., Kumar, A., Ravindra, M. P., Bertino, E., and Kumar, R., editors, *Proceedings of the 20th International Conference on World Wide Web, WWW 2011, Hyderabad, India, March 28 - April 1, 2011 (Companion Volume)*, pages 283–284. ACM.

[Guy et al., 2013] Guy, I., Chen, L., and Zhou, M. X. (2013). Introduction to the special section on social recommender systems. *ACM TIST*, 4(1):7.

[Guy et al., 2009] Guy, I., Ronen, I., and Wilcox, E. (2009). Do you know?: recommending people to invite into your social network. In *Proceedings of the 14th international conference on Intelligent user interfaces*, IUI '09, pages 77–86, New York, NY, USA. ACM.

[Hamari and Koivisto, 2013] Hamari, J. and Koivisto, J. (2013). Social motivations to use gamification: An empirical study of gamifying exercise. In *ECIS*, page 105.

[Hamasaki et al., 2007] Hamasaki, M., Matsuo, Y., and Nisimura, T. (2007). Ontology extraction using social network.

[Hannon et al., 2010] Hannon, J., Bennett, M., and Smyth, B. (2010). Recommending twitter users to follow using content and collaborative filtering approaches. In *Proceedings of the fourth ACM conference on Recommender systems*, RecSys '10, pages 199–206, New York, NY, USA. ACM.

[Herlocker et al., 1999] Herlocker, J. L., Konstan, J. A., Borchers, A., and Riedl, J. (1999). An algorithmic framework for performing collaborative filtering. In *SIGIR '99: Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, August 15-19, 1999, Berkeley, CA, USA*, pages 230–237. ACM.

[Hotho et al., 2006a] Hotho, A., Jaschke, R., Schmitz, C., and Stumme, G. (2006a). Folkrank: A ranking algorithm for folksonomies. In *UNIVERSITY OF HILDESHEIM, INSTITUTE OF COMPUTER SCIENCE*, pages 111–114.

[Hotho et al., 2006b] Hotho, A., Jäschke, R., Schmitz, C., and Stumme, G. (2006b). Information retrieval in folksonomies: search and ranking. In *Proceedings of the 3rd European conference on The Semantic Web: research and applications*, ESWC'06, pages 411–426, Berlin, Heidelberg. Springer-Verlag.

[Iaquinta et al., 2008] Iaquinta, L., de Gemmis, M., Lops, P., Semeraro, G., Filannino, M., and Molino, P. (2008). Introducing serendipity in a content-based recommender system. In *HIS*, pages 168–173. IEEE Computer Society.

[Jayant and Saponas, 2005] Jayant, R. and Saponas, T. S. (2005). Mariofit: Exercise through mobile entertainment.

[Jäschke et al., 2012] Jäschke, R., Hotho, A., Mitzlaff, F., and Stumme, G. (2012). Challenges in tag recommendations for collaborative tagging systems. In Pazos Arias, J. J., Fernández Vilas, A., and Díaz Redondo, R. P., editors, *Recommender Systems for the Social Web*, volume 32 of *Intelligent Systems Reference Library*, pages 65–87. Springer, Berlin/Heidelberg.

[Kaplan and Haenlein, 2010] Kaplan, A. M. and Haenlein, M. (2010). Users of the world, unite! the challenges and opportunities of social media. *Business Horizons*, 53(1):59–68.

[King et al., 2013] King, A. C., Hekler, E. B., Grieco, L. A., Winter, S. J., Sheats, J. L., Buman, M. P., Banerjee, B., Robinson, T. N., and Cirimele, J. (2013). Harnessing different motivational frames via mobile phones to promote daily physical activity and reduce sedentary behavior in aging adults. *PLoS One*, 8(4):e62613.

[Konstan et al., 2006] Konstan, J. A., McNee, S. M., Ziegler, C.-N., Torres, R., Kapoor, N., and Riedl, J. (2006). Lessons on applying automated recommender systems to information-seeking tasks. In *Proceedings, The Twenty-First National Conference on Artificial Intelligence and the Eighteenth Innovative Applications of Artificial Intelligence Conference, July 16-20, 2006, Boston, Massachusetts, USA*, pages 1630–1633. AAAI Press.

[Liben-Nowell and Kleinberg, 2003] Liben-Nowell, D. and Kleinberg, J. M. (2003). The link prediction problem for social networks. In *Proceedings of the 2003 ACM CIKM International Conference on Information and Knowledge Management, New Orleans, Louisiana, USA, November 2-8, 2003*, pages 556–559. ACM.

[Lops et al., 2011] Lops, P., de Gemmis, M., and Semeraro, G. (2011). Content-based recommender systems: State of the art and trends. In Ricci, F., Rokach, L., Shapira, B., and Kantor, P. B., editors, *Recommender Systems Handbook*, pages 73–105. Springer.

[Manca et al., 2013] Manca, M., Boratto, L., and Carta, S. (2013). Producing friend recommendations in a social bookmarking system by mining users content. In *IMMM 2013, The Third International Conference on Advances in Information Mining and Management*, pages 59–64.

[Manca et al., 2014a] Manca, M., Boratto, L., and Carta, S. (2014a). Behavioral mining to produce novel and serendipitous friend recommendations in a social bookmarking system: a delicious case-study. In *Performance Evaluation Journal*.

[Manca et al., 2014b] Manca, M., Boratto, L., and Carta, S. (2014b). Design and architecture of a friend recommender system in the social bookmarking domain. In *Science and Information Conference 2014 - SAI-2014*.

[Manca et al., 2014c] Manca, M., Boratto, L., and Carta, S. (2014c). Mining user behavior in a social bookmarking system: a delicious friend recommender system. In *8th International Conference on Autonomous Infrastructure, Management and Security AIMS 2014*.

[Markland and Ingledew, 1997] Markland, D. and Ingledew, D. (1997). The measurement of exercise motives: Factorial validity and invariance across gender of a revised exercise motivations inventory. *British Journal of Health Psychology*, 2:361–376.

[Mika, 2007] Mika, P. (2007). Ontologies are us: A unified model of social networks and semantics. *Web Semant.*, 5(1):5–15.

[Mulas et al., 2013a] Mulas, F., Carta, S., Pilloni, P., and Boratto, L. (2013a). Everywhere run: a virtual personal trainer for supporting people in their running activity. *International Journal of Advanced Computer Science*, 3(2):75–79.

[Mulas et al., 2011] Mulas, F., Carta, S., Pilloni, P., and Manca, M. (2011). Everywhere run: a virtual personal trainer for supporting people in their running activity. In *Proceedings of the 8th International Conference on Advances in Computer Entertainment Technology*, ACE '11, pages 70:1–70:2, New York, NY, USA. ACM.

[Mulas et al., 2012] Mulas, F., Pilloni, P., and Carta, S. (2012). Everywhere race!: A social mobile platform for sport engagement and motivation. In

SOTICS 2012, The Second International Conference on Social Eco-Informatics, pages 63–69.

[Mulas et al., 2013b] Mulas, F., Pilloni, P., Manca, M., Boratto, L., and Carta, S. (2013b). Using new communication technologies and social media interaction to improve the motivation of users to exercise. In *In Proceedings of 2nd Int. Conf. on Future Generation Communication Technologies (FGCT 2013)*.

[Mulas et al., 2013c] Mulas, F., Pilloni, P., Manca, M., and Carta, S. (2013c). Linking human-computer interaction with the social web: a web application to improve motivation in the exercising activity of users. In *4th IEEE International Conference on Cognitive Infocommunications*, pages 351–356.

[Newman and Girvan, 2004] Newman, M. E. J. and Girvan, M. (2004). Finding and evaluating community structure in networks. *Physical Review*, E 69(026113).

[Nielsen and Landauer, 1993] Nielsen, J. and Landauer, T. (1993). A mathematical model of the finding of usability problems. *Proceedings of ACM INTERCHI'93 Conference on Human Factors in Computing Systems*, pages 206–213.

[O'Mahony et al., 2004] O'Mahony, M. P., Hurley, N. J., Kushmerick, N., and Silvestre, G. C. M. (2004). Collaborative recommendation: A robustness analysis. *ACM Trans. Internet Techn.*, 4(4):344–377.

[Pearson, 1896] Pearson, K. (1896). Mathematical contributions to the theory of evolution. iii. regression, heredity and panmixia. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Math. or Phys. Character (1896-1934)*, 187:253–318.

[Quercia and Capra, 2009] Quercia, D. and Capra, L. (2009). Friendsensing: recommending friends using mobile phones. In Bergman, L. D., Tuzhilin, A., Burke, R. D., Felfernig, A., and Schmidt-Thieme, L., editors, *RecSys*, pages 273–276. ACM.

[Rendle and Schmidt-Thieme, 2010] Rendle, S. and Schmidt-Thieme, L. (2010). Pairwise interaction tensor factorization for personalized tag recommendation. In *Proceedings of the third ACM international conference on Web search and data mining*, WSDM '10, pages 81–90, New York, NY, USA. ACM.

[Ricci et al., 2011] Ricci, F., Rokach, L., and Shapira, B. (2011). Introduction to recommender systems handbook. In Ricci, F., Rokach, L., Shapira, B., and Kantor, P. B., editors, *Recommender Systems Handbook*, pages 1–35. Springer.

[Sampaio, 2013] Sampaio, A. (2013). Quantifying the user experience: Practical statistics for user research by jeff sauro and james r. lewis. *SIGSOFT Softw. Eng. Notes*, 38(1):57–58.

[Sauro, 2011] Sauro, J. (2011). *A Practical Guide to the System Usability Scale: Background, Benchmarks and Best Practices*. CreateSpace.

[Sebastiani, 2002] Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM Comput. Surv.*, 34(1):1–47.

[Shani and Gunawardana, 2011] Shani, G. and Gunawardana, A. (2011). Evaluating recommendation systems. pages 257–297.

[Sigurbjörnsson and van Zwol, 2008] Sigurbjörnsson, B. and van Zwol, R. (2008). Flickr tag recommendation based on collective knowledge. In *Proceedings of the 17th international conference on World Wide Web*, WWW '08, pages 327–336, New York, NY, USA. ACM.

[Simon, 1971] Simon, H. A. (1971). Designing organizations for an information rich world. In Greenberger, M., editor, *Computers, communications, and the public interest*, pages 37–72. Johns Hopkins Press, Baltimore.

[Specia and Motta, 2007] Specia, L. and Motta, E. (2007). Integrating folksonomies with the semantic web. In *Proceedings of the 4th European conference on The Semantic Web: Research and Applications*, ESWC '07, pages 624–639, Berlin, Heidelberg. Springer-Verlag.

[Symeonidis et al., 2008] Symeonidis, P., Nanopoulos, A., and Manolopoulos, Y. (2008). Tag recommendations based on tensor dimensionality reduction. In *RecSys*, pages 43–50. ACM.

[Tan et al., 2005] Tan, P.-N., Steinbach, M., and Kumar, V. (2005). *Introduction to Data Mining, (First Edition)*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

[Toscos et al., 2006] Toscos, T., Faber, A., An, S., and Gandhi, M. P. (2006). Chick clique: persuasive technology to motivate teenage girls to exercise. In *CHI '06 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '06, pages 1873–1878, New York, NY, USA. ACM.

[Turetken and Olfman, 2013] Turetken, O. and Olfman, L. (2013). Introduction to the special issue on human-computer interaction in the web 2.0 era. *AIS Transactions on Human-Computer Interaction*, 5(1).

[van Dongen, 2000] van Dongen, S. (2000). *Graph Clustering by Flow Simulation*. PhD thesis, University of Utrecht.

[Virzi, 1992] Virzi, R. A. (1992). Refining the test phase of usability evaluation: how many subjects is enough? *Hum. Factors*, 34(4):457–468.

[Weiksner et al., 2008] Weiksner, G. M., Fogg, B. J., and Liu, X. (2008). Six patterns for persuasion in online social networks. In Oinas-Kukkonen, H., Hasle, P. F. V., Harjumaa, M., Segerståhl, K., and Øhrstrøm, P., editors, *PERSUASIVE*, volume 5033 of *Lecture Notes in Computer Science*, pages 151–163. Springer.

[Wu et al., 2006] Wu, X., Zhang, L., and Yu, Y. (2006). Exploring social annotations for the semantic web. In *Proceedings of the 15th international conference on World Wide Web*, WWW '06, pages 417–426, New York, NY, USA. ACM.

[Wu and Palmer, 1994] Wu, Z. and Palmer, M. (1994). Verbs semantics and lexical selection. In *Proceedings of the 32nd annual meeting on Association for Computational Linguistics*, ACL '94, pages 133–138, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Xu et al., 2006] Xu, Z., Fu, Y., Mao, J., and Su, D. (2006). Towards the semantic web: Collaborative tag suggestions. In *Proceedings of the Collaborative Web Tagging Workshop at the WWW 2006*, Edinburgh, Scotland.

[Zhang and Hurley, 2008] Zhang, M. and Hurley, N. (2008). Avoiding monotony: improving the diversity of recommendation lists. In *RecSys*, pages 123–130. ACM.

[Zhou et al., 2010] Zhou, T. C., Ma, H., Lyu, M. R., and King, I. (2010). Userrec: A user recommendation framework in social tagging systems. In Fox, M. and Poole, D., editors, *AAAI*. AAAI Press.

[Zhou et al., 2012] Zhou, X., Xu, Y., Li, Y., Jøsang, A., and Cox, C. (2012). The state-of-the-art in personalized recommender systems for social networking. *Artif. Intell. Rev.*, 37(2):119–132.

[Ziegler et al., 2005] Ziegler, C.-N., McNee, S. M., Konstan, J. A., and Lausen, G. (2005). Improving recommendation lists through topic diversification. In Ellis, A. and Hagino, T., editors, *Proceedings of the 14th international conference on World Wide Web, WWW 2005, Chiba, Japan, May 10-14, 2005*, pages 22–32. ACM.