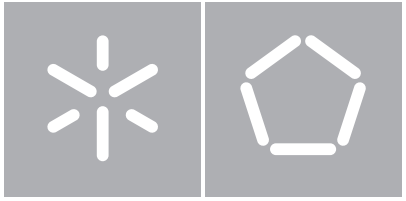
Universidade do Minho
Escola de Engenharia

Adão Carlos Fernandes Rodrigues

Analysis Of User Profile In Social Network

Outubro de 2013



Universidade do Minho
Escola de Engenharia
Departamento de Engenharia Informática

Adão Carlos Fernandes Rodrigues

Analysis Of User Profile In Social Network

Dissertação de Mestrado
Mestrado em Engenharia Informática

Trabalho realizado sob orientação de
Professor José Carlos Maia Neves

Outubro de 2013

"Knowledge Discovery is the nontrivial extraction of implicit, previously unknown, and potentially useful information from data."

William J. Frawley

Acknowledgements

First is a sincere gratitude to my adviser Professor Jose Carlos Maia Neves, who led and helped with her vast knowledge in the area, which made possible the success of the project. Likewise a thank you to my co-supervisor Professor Paulo Novais who helped me actively and intensively throughout the period of development of the dissertation.

I must thank the company that gave support and without it nothing would be possible, Brandit Portugal - Integrated Marketing Solutions and Communications, provided all infrastructure as well as full support for any problem that arose in development.

I also want to thank my family and friends who always encouraged me and motivated in the most difficult times. Thanks also to my friend in particular Joana Festa, who through their opinion, criticism and suggestions along this route, influenced this project.

Thank you all

Contents

List of Figures

Abbreviations

SNS	**S**ocial **N**etwork **S**ites
BN	**B**ayesian **N**etworks
CBR	**C**ase-**B**ased **R**easoning
AR	**A**ssociation **R**ules
NN	**N**eural **N**etworks
K-NN	**K**-**N**earest **N**eighbor
IPS	**I**nter-**P**rofile **S**imilarity
ML	**M**achine **L**earning

Chapter 1

Introduction

The Internet is part of every peoples life when worldwide population is taken into account. With the dependency that was created around this, new forms of communication have been created, where social networks sites (SNS) are also included. This structure exists since long before the thought of what Internet could be, appearing as a group of people who relate to each other. With the emergence of SNS, this concept remained, but the way people communicate and interact has changed. Nowadays there are numerous SNS that provide different types of services and diversified content where we can share information and interact with everyone.

Popular SNS such as MySpace and Facebook provide communication, storage and applications for hundreds of millions of users. Users join, establish links to friends, and leverage their links to share content, organize events, and search for specific users or shared resources. Provide platforms for organizing events, user-to-user communication, and are among the Internets most popular destinations [Wilson et al., 2009]. With different purposes, there are YouTube and Flickr that allow sharing videos and photos on the Internet, respectively. Also referred as a micro-blogging service, twitter are text-based posts of up to 140 characters displayed on the author's profile page and delivered to other users, known as followers. This service consists to send and read each others texts.

Today, SNS have been the subjects of several studies with the purpose of studying users interactions and behavior. To quantify the impact of the observations and to increase significantly the accuracy of the users' characterization, a great amount of data

has been collected in the context of many different studies. These studies are distinguished by the different approaches each follows, according to their expected results or aspect under analysis. There is a number of different Artificial Intelligence techniques that have been applied to these research area, such like Ad Hoc, Neural networks or Genetic Algorithm, Case-Based Reasoning, Decision Tree, Bayesian Networks and Association Rules. According to these studies, there is a need to find the best approach to follow, based on the profile structure, and get the most relevant information. Having this, a strong base of information is required to that it take effect immediately in the nearby future.

This work intends to create/identify user profiles through their actions on SNS. This identification aims to determine, in a specific way, which profile each user has, linking between some dimensions and their sets of variables: sociodemographic characteristics (gender, age, education) the specific type of practices conducted over the Internet (study, work, services, search for information, communication and entertainment), the context of use of SNS (home, school, workplace or other). In the scope of this master thesis, the study will be conducted on Facebook, the most popular SNS in the world, as it features a vast collection of data.

After a careful analysis, we are able to separate different types of users based on the association of their sets of variables. This analysis also deepens the knowledge about the various uses of SNS, and may also be useful to the market in that it provides substantive information concerning the forms of articulation between the social characteristics of users and their activities, schemes and contexts of use.

An overview to Facebook

Facebook was founded in February 2004 and nowadays have more than 425 million active users access Facebook through mobile devices across 200 mobile operators in 60 countries [Protalinski, 2012]. This SNS let users use the site to interact with people they already know or to meet new people. Before this kind of interactions, users need to create a profile with personal information that will identify them on the social network. After this step, they can accumulate 'friends' who can post comments on each others pages, and view each others profiles. Facebook members can also join virtual groups based on common interests, see what classes they have in common, and learn each others hobbies, interests, musical tastes, and romantic relationship status through the profiles [Ellison

et al., 2007]. The data collected for this study, was restricted to Facebook members 'likes'. Each user has associated a list of ids that has the necessary information from each like. Follows an example of a 'like' json request below on Figure 1.1.


```
$json (
    about = "Gareth Bale Welcome To Real Madrid"
    can_post = TRUE
    category = "Athlete"
    is_published = TRUE
    talking_about_count = 1284
    username = "Bale.Official.11"
    were_here_count = 0
    id = "174542652710626"
    name = "Gareth Bale."
    link = "https://www.facebook.com/Bale.Official.11"
    likes = 11784
    cover (
        cover_id = 209852542512970
        source = "https://fbcdn-sphotos-d-a.akamaihd.net/hphotos-ak-ash4/1175695_209852542512970_137128257_n.jpg"
        offset_y = 0
        offset_x = 0
    )
)
```


FIGURE 1.1: JSON Example.

All this information will be filtered to a more compact response, with only the relevant information. Each like will be connected to a major category presented in Figure 1.2.

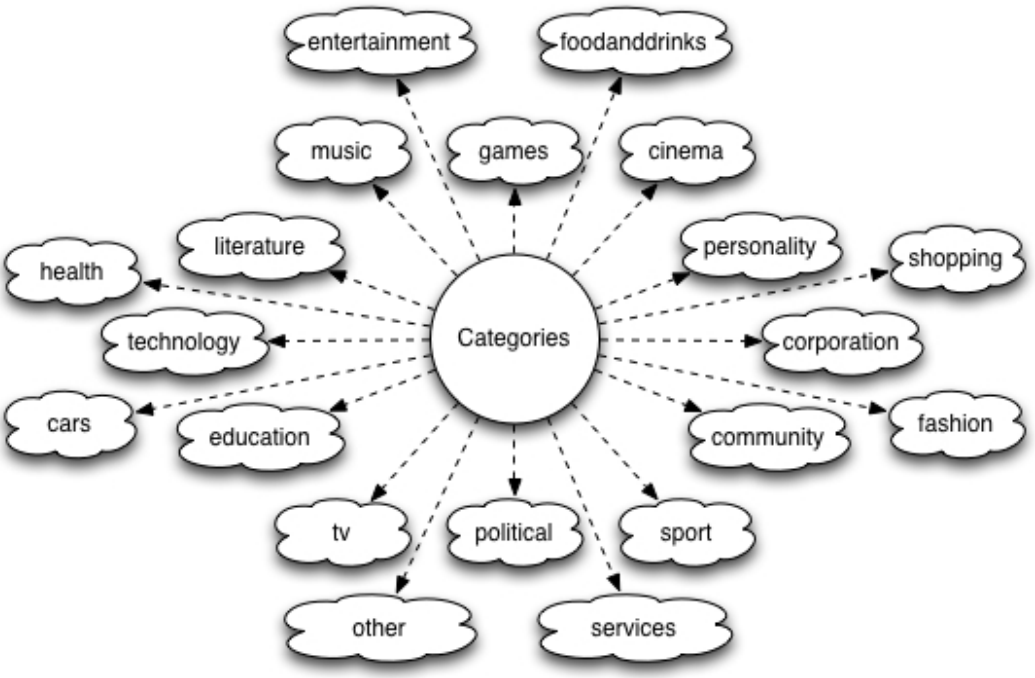


FIGURE 1.2: List Of Categories.

1.1 Objective

The information that exists on the Internet is increasing and is becoming a value resource. Many companies are trying to exploit this are with the goal of getting the users that can bring them success and money. To get those users, there is some information that needs to be collected and analyzed. Information that is individual from one user to another. This area is user profiling.

This work aims to identify user profiles within various SNS available today. All this identification goes through a very detailed analysis not only on the actions of users in such media as the characteristics that each user has [Bhattacharyya et al., 2011]. The importance of a theoretical scenario comes with the fact that it involves not only the growing and extending access of these services by citizens, business and public institutions, but also the expansion, diversification and intensification of its use in different contexts.

SNS have been growing very fast around the world, changing people's way of interaction with the internet and their relationships with others. This happens because SNS make it easier for people to have new friendships or rediscover long ones. Also in terms of communication and consumption habits, it is possible to see this effect, just by looking at the present day life routines. Looking to the present day, we see clearly that social networks have become part of the daily routine of people, changing their habits completely. Influenced not only interpersonal relations but also, for example, the habit of watching television and reading. The insertion of sharing your videos and blogs almost killed this media. SNS are being used as workplaces where companies can build teams. These teams help solve problems faster, since these networks act as a great service for sharing information, not only among employees but also between partners. These services enable companies to combine skills of people working around the world.

The work involved has as main objective the creation of user profiles. The data capture is crucial for the characterization of each user. Thus it is important to gather data sources to help us evaluate these users. An evaluation passes through a careful analysis of the characteristics of each user. The collected data was analyzed according to the following dimensions: sociodemographic characteristics (gender, age, education) the specific type of practices conducted over the Internet (study, work, services, search

for information, communication and entertainment), the context of use of SNS (home, school, workplace or other). After considering these dimensions, we have information to make the profiling. To achieve the desired results some steps are commonly followed:

- Identify how many data sources / social networks exist;
- Identify what information we can collect from each user;
- Identify how / who users relate;
- Identify ways of categorizing users not only using their personal data, but with the preferences that appears on theirs profile;
- Identify the tastes and interests of users.

After identifying all these points, we will develop an algorithm that:

- Analyze the information and create users according to their profile;
- Identify the issues that users are sharing or interacting;
- Through some parameters classify users by levels;
- Characterize the connection between users and their profiles.

Intends with the final solution not only help to study the profiles of users in the SNS but also present an algorithm which is able to categorize each user type considering the data that will be collected throughout the investigation.

One must take into account are the users with more data available that the degree of reliability is higher, when defining the profiles of each user. To obtain the data may also be used, if possible, Web Crawlers [Brandman et al., 2000] to keep our database as updated as possible. The collected data intends however that each user acts in a natural way in order to get reliable figures and not manipulated.

1.2 Research Methodology

To achieve the objectives present in the previous section a methodology was followed. First all the collected information was organized and evaluated with the purpose of give support and help to solve the problem in question. Second, that support will help to give the results expected. Finally, the results are analyzed and validated in order to trigger the final conclusions about the problem. To follow this methodology, some steps needs to be taken in account:

- Specify the problem discussed
- Update, whenever necessary, the related work
- Implementation of the solution
- Validation of the solution
- Analyze the results achieved

1.3 Structure

After presenting the problem in question and the objectives to resolve it, in section 2 is presented the process that refers to construction of a profile via the extraction from a set of data. This definition is essential to understand what data is important to collect to identify a user profile. In section 3, is presented a group of techniques that can be a solution to the problem found. In the Section 4, is presented some related works that helped to find the best approach to achieve the best results. Those works helped to observe the different methods used to acquire information about at specific user and finally build is own profile. In this studies are specified what each problem used to get the final results and what problems they exceeded. In the section 5, is presented the selected intelligent technique that will resolve the problem, followed by the steps that was needed to find the solution. In the section 6, is presented the tests made with the technique and the results that were achieved. Finally, in section 7, were made conclusions and possible directions that can be followed in the future. In the final Section, are presented some references that helped to substantiate the problem.

- Chapter 1: Introduction
- Chapter 2: User Profiling
- Chapter 3: Intelligent Techniques
- Chapter 4: Related Work
- Chapter 5: Implementation
- Chapter 6: Testing and Evaluation
- Chapter 7: Conclusion and future directions

1.4 Privacy Policy

A major ethical care will be the privacy of user data that will directly or indirectly participate in the study. One of the risks that the data collection can present the user is limited to information that may be collected. To make sure that people, who is presented in these study, understand the nature and extent of the requests, all gathered data were authorized by them and they needed to accept our request to access their private content.

Chapter 2

User Profiling

A profile contains the most important information about a user like name, age, location, etc. When looking inside the context of users of software applications, profiling can be much more than just personal information. Every user differs on their preferences and find what kind of information can determinate who you are is essential.

Silvia Schiaffino and Anala Amandi [Schiaffino and Amandi, 2009] discuss how the user profile is represented and how that information is acquired and build. The content of a user varies from one place to another because the context changed too. Getting an example: considering an online newspaper domain and a calendar management domain. On the first one, the user profile contains news about what he likes and dislikes reading. Taking the calendar management, there are information about time and date. The content of a user profile has to be learned using some techniques. Each individual user represents a different kind of information sometimes but there is a set of the most common contents between the users: the knowledge, background and skills, the goal and behavior, individual characteristics and users context.

Schiaffino refers that user interests are one of the most important part of the user profile in information retrieval, filtering systems and adaptive systems that are information- driven. The users behavior comes also as an important part of user profile. Depends on the domain and can be represented a pattern if it is repetitive, or has a routine. Users context appears as a quite new feature in user profiling. The information collected may be explicitly input by the user or implicitly gathered by a software agent. Explicitly input comes as the last option because is more intrusive, saving exceptions.

It may be collected on the user's client machine or gathered by the application server itself. Depending on how the information is taken, different data about the users may be extracted. In Personalized Services general, systems that collect implicit information place little or no burden on the user are more likely to be used and, in practice, perform as well or better than those that require specific software to be installed and/or explicit feedback to be collected. Getting all the demographic data actually turns out to be more accurate than surveys to customers themselves. Usually, all that is required to get full demographic data is a credit card number or the combination of name and zip code, information that is often collected during purchase or registration. The most reliable approach is software agents that are incorporated inside user's computer. However, it requires user-participation in order to install the desktop software. User profiles may be based on heterogeneous information associated with an individual user or a group of users who showed similar interests [Gauch et al., 2007].

With all the information available on the Internet, getting only the essential part is crucial to successfully build a profile. The system may acquire explicit information using questionnaires or explicitly by watching users' actions and behaviors. To learn a user profile from a user's actions, some conditions need to be achieved. The user behavior must be a pattern otherwise there is no conditions to build an individual user profile. According to this conditions the user behaviour has to be repetitive and perform similar actions under different situations.

Types of information in a user profile [Schiaffino and Amandi, 2009]:

- **Personal information** is one type of information to gather. Information as name, age, country, etc.
- **Interests** of a user are the most important information to gather. This information contains activities, works and much more that the user are interested.
- **Behavior** is one kind of information that is gathered implicitly. With the user behaviour there is a possibility to represent a pattern.
- **Goals** of a user are important to detect user's objective. Find what the user wants is not trivial and can be very important.

Explicit user information collection: The data collected may contain demographic information such as birthday, marriage status, job, or personal interests. In addition to simple checkboxes and text fields, a common feedback technique is the one that allows users to express their opinions by selecting a value from a range. All these methodologies have the drawback that they cost the user's time and require the users willingness to participate. If users do not voluntarily provide personal information, no profile can be built for them. With this method, the information is gathered through direct user interaction. With this kind of gathering information, comes some problems: first, users are not prepared to give information by filling long forms, second they can give wrong or false information about the question, and third they can not tell or write what they really want, feel or means to. Normally the information gathered with this way is demographic, like the user's age, name and hobbies. In some cases this kind of information constitutes the factual profile, as Adomavicius and Alexander [Adomavicius and Tuzhilin, 2001] reported.

Implicit user information collection: As Gauch [Gauch et al., 2007] said, user profiles are often constructed based on implicitly collected information, often called implicit user feedback. The main advantage of this technique is that it does not require any additional intervention by the user during the process of constructing profiles. On this method, there are agents that monitor user activities. Kelly [Kelly and Teevan, 2003] shows an overview of the most popular techniques used to collect implicit feedback and the information about the user based on the user's behaviour. This technique has its advantages when comparing to explicit, that removes the cost to the user of providing feedback. However, both techniques can be combined to achieve a better result.

Comparing Explicit and Implicit user information collection: Quiroga [Quiroga and Mostafa, 1999] compared the results obtained between profiles that were built using explicit, implicit and both ways together using a collection of 6000 health records, classified into 15 health areas referred to as classes. Each user used the system for 15 sessions and the profiles built with the combination feedback obtained the highest precision followed by explicit feedback alone and then implicit feedback alone. The differences presented on these results were found to be statistically relevant, telling that systems that implements a explicit or a combination of explicit and implicit feedback, gives better results than an alone implicit feedback. Contradicting Quiroga, White [White et al., 2001] consider that profiles using implicit feedback or explicit feedback

does not have significant differences. To find out, White performed experiments some users to answer specific questions on the web. The results told that users with implicit feedback were able to complete 61 in a total of 64 questions, against 57 of explicit feedback. Since the differences were not statistically significant, the author concluded that were identical. In 2005, Teevan [Teevan et al., 2005] performed better results with the user profiles constructed with implicit feedback than the users with the explicit feedback. According to these authors, the experiences can change, depending on the information collected. Once the using of implict feedback is growing, this also means that the information gathered for this kind of profile is better too.

2.1 Information Filtering

There are some typical characteristics/features that is commonly used when trying to define information filtering [Belkin and Croft, 1992]:

- **An information filtering system** is an information system designed for unstructured or semistructured data. Email messages are an example of semistructured data in that they have well-defined header fields and an unstructured text body;
- **Information filtering systems** deal primarily with textual information where, in fact, unstructured data is often used as a synonym for textual data;
- **Filtering systems** involve large amounts of data;
- **Filtering applications** typically involve streams of incoming data;
- **Filtering** is based on descriptions of individual or group information preferences, often called profiles;
- **Filtering** is often meant to imply the removal of data from an incoming stream, rather than finding data in that stream;

Have been proposed different architectures to build an efficient filtering system. Moukas [Moukas, 1997*b*] said that information filtering systems can be categorized along several different axes based on the technology/architecture they use for filtering the data. They can all be classified under two broad categories:

- **Content-based filtering** try to recommend content/items to the users. As Lops [Lops et al., 2011] described, the basic process performed by a content-based consists in matching up the attributes of a user profile in which preferences and interests are stored, with the attributes of a content object (item), in order to recommend to the user new interesting items. This kind of system needs some techniques for representing and producing the user profile: content analyser, profile learner and filtering component. Some advantages and drawbacks have been found about this technique. This technique has it advantages and disadvantages. Systems that implements a content-based approach learn from the content of the text documents or a set of documents. The so-called vector representation is the most frequently used document representation in information retrieval and text learning [Mladenic, 1999];

- **Social (or collaborative) and Economic-based filtering** has a different approach when comparing to content-based filtering. The objective is to use the feedback and rating given from all different users and filter out irrelevant information. This index is not global, but is computed for each user on the fly by using other users with similar interests: documents that are liked by many people will have a priority over documents that are disliked. It takes into consideration parameters like the price of the document and its cost of transmission from the source to the user (in the case of company intranets) when making decisions on whether to filter it out or not [Moukas, 1997*b*];

2.2 Information Retrieval

According to Belkin [Belkin and Croft, 1992], information retrieval has some different characteristics when comparing to information filtering:

- **Information Retrieval** is normally used with static databases of information;
- **Information Retrieval** is typically concerned with single uses of the system;
- **Information Retrieval systems** is normally query based;

As information filtering, information retrieval can be splited into different categories: boolean-based systems, vector-space based system and probabilistic systems [Salton and Buckley, 1988]:

- **Boolean-based systems** use boolean operators (like AND,OR,NOT) to find an exact match;
- **Vector-space based system** is used for representing text documents with a multi-dimensional vector of keywords and weights. One of the advantages of this method is that allows ranking documents according to their possible relevance;
- **Probabilistic systems** identify relevant and non-relevant in the database items using inference network models;

2.3 Keywords Profiles

One of the most famous representation for user profiles are sets of keywords. Those keywords can be represented in many different ways. One keyword can represent a topic of interest or can be grouped into categories. Following this, each profile is represented in a form of keyword-vector where each keyword have a weight associated. Follows and example of a weight keyword-based user profile on Figure 2.1:

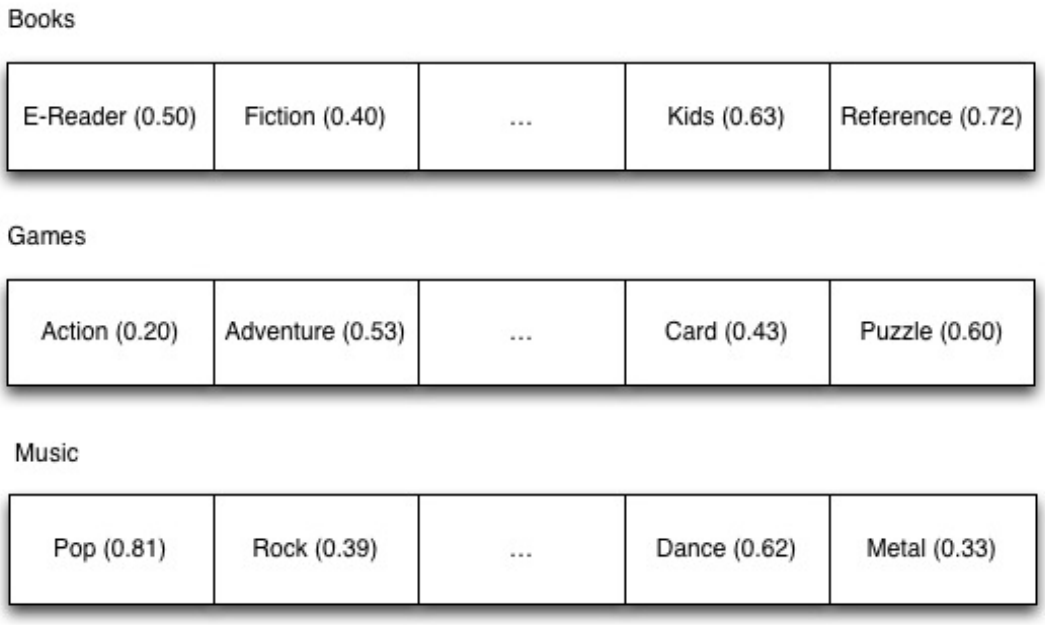


FIGURE 2.1: Weight Keyword-Based.

Gauch [Gauch et al., 2007] said that profiles represented in this way were among the first to be explored. These kind of contents are gathered from documents visited by the user or saved by the user during is experiment, or else the keywords were explicitly provided by the user. Each keyword has associated a weight that represents its importance in the profile. Amalthaea [Moukas, 1997*a*] is a system that creates keywords profiles. Is an evolving, multiagent ecosystem for personalized filtering, discovery, and monitoring of information sites. Amalthaea's primary application domain is the World-Wide-Web and its main purpose is to assist users in finding interesting information.

2.4 Semantic Network Profiles

InfoWeb [Gentili et al., 2003] builded a semantic network that represent long-term user interests where each user profile is represented as a semantic network of concepts. Each network contains an amount of nodes unlinked where each node represents a concept with a specific weight. As more information is gathered from the user, more enriched will be with additional weight keywords. It uses a stereotype-based mechanism for the construction of the initial user model. The ability of InfoWeb to expand the query on the basis of the semantic network that makes up the user model has been appreciated by

users because of the importance of inputting the right query to the system. According to InfoWeb [Gentili et al., 2003] their tests demonstrated that, after a certain number of queries, the system is sufficiently fast in reaching the stability of the model for a users domain of interest, thus obtaining satisfactory performance. The system has also shown its ability to adapt to sudden changes in user interests.

The goal of Sieg [Sieg et al., 2007] is to utilize the user context to personalize search results for a given query. The personalization is achieved by reranking the results returned from a search engine. An ontological approach to user profiling has proven to be successful in addressing the cold-start problem in recommender systems where no initial information is available early on upon which to base recommendations [Middleton et al., 2003]. The purpose of Sieg of using an ontology is to identify topics that can be interesting to the user. Every time the user interacts with the system, the ontological user profile is updated. Accurate the information about the user is very important. Too many factors are taking into account, as the time spent in each page, how many times the page is visited and which pages are bookmarked [Dumais et al., 2003].

2.5 Concept Profiles

Concept-based profiles and semantic network profile are related and both are represented by nodes and connections between (Figure 2.2). In concept nodes, each node is not represented as set of words or some specific word, these nodes contains more abstract topics that is considered relevant to the user. Determinate how much important some topic is to an user is not easy, and to reach that importance, each topic has a weight associated. Bloedorn [Bloedorn et al., 1996] has demonstrated that a relevant generalization hierarchy together with a hybrid feature representation is effective for accurate profile learning.

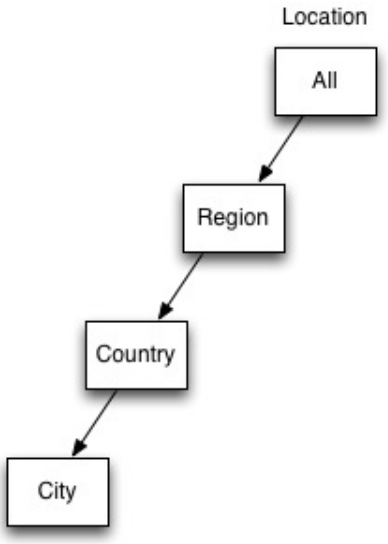


FIGURE 2.2: Concept Hierarchies.

Concept hierarchies were initially used to represent the content of Web pages but have more recently been used to represent user profiles. Most systems are based on a reference concept hierarchy, or taxonomy, from which a subset of the concepts and relationships are extracted and weighted to form a user profile. Because creating a broad and deep concept hierarchy is an expensive, mostly manual process, profiles are typically based on subsets of existing concept hierarchies [Gauch et al., 2007].

2.6 User Representations

Gauch [Gauch et al., 2007] said that user profiles are generally represented as sets of weighted keywords, semantic networks, or weighted concepts, or association rules. Keyword profiles are the simplest to build, but they require a large amount of user feedback in order to learn the terminology by which a topic might be discussed. According to the user interest, the system should reflect the user interest based on his/her activity. The information of a user in these cases is dynamically, since a static profile maintains the same information indefinitely. The content inside a dynamic profile may change constantly. Short-terms indicates interests that remains static to the user unlike long-terms that can changes over and over again. Profiles that can change over the time are called dyanmic profile and those that maintain the same information over time are called static profiles [Hoashi et al., 2000, Widyantoro et al., 2000]. Short-terms may be more difficult

to find and manage than long-terms interests. The purpose of user profiling is to collect information about a user interests, and determinate how longer will take those interests, aiming to improve the quality of that information. The user profiling process generally consists in three main phases [Gauch et al., 2007] as we showed below in Figure 2.3.

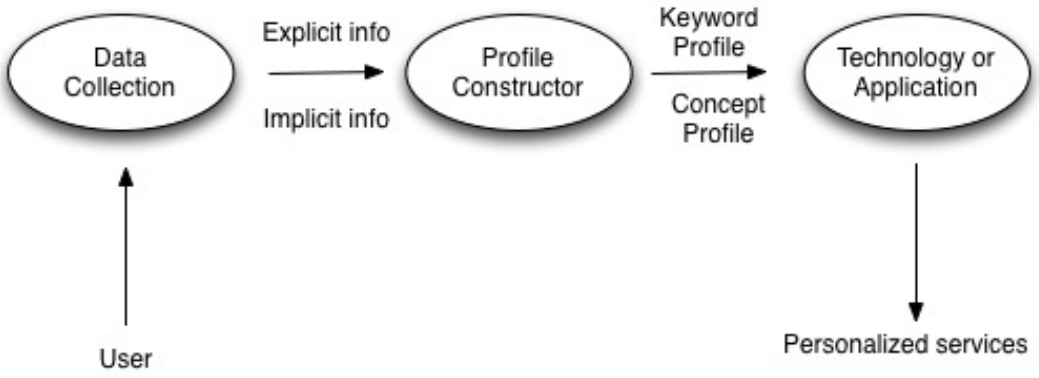


FIGURE 2.3: User Profiling Process.

First, is gathered information about the user in question through a process. From user to user, the information collection can change, and each one will define which data can be extracted. After this phase, its time to center the attention to the user profile construction based on the data collected that may be represented in a variety of ways, depending of each profile. After this process, the user is exposed. There are different patterns to represent a user profile:

- **Static user model** are the most basic kinds of user models. Once the main data is gathered they are normally not changed again, they are static. Shifts in users' preferences are not registered and no learning algorithms are used to alter the model.

- **Dynamic user models** allow a more up to date representation of users. Changes in their interests, their learning progress or interactions with the system are noticed and influence the user models. The models can thus be updated and take the current needs and goals of the users into account.

- **Stereotype based user models** are based on demographic statistics. Based on the gathered information users are classified into common stereotypes. The system then adapts to this stereotype. The application therefore can make assumptions

about a user even though there might be no data about that specific area, because demographic studies have shown that other users in this stereotype have the same characteristics. Thus, stereotype based user models mainly rely on statistics and do not take into account that personal attributes might not match the stereotype. However, they allow predictions about a user even if there is rather little information about him or her.

- **Highly adaptive user models** try to represent one particular user and therefore allow a very high adaptivity of the system. In contrast to stereotype based user models they do not rely on demographic statistics but aim to find a specific solution for each user. Although users can take great benefit from this high adaptivity, this kind of model needs to gather a lot of information first.

2.7 User Construction

Every user is represented by the information he has and the actions he make. To create a user profile is preferred to use a less intrusive method where we can extract from the user all the information that matters to identify that user. There are too many techniques that can be used, based on machine learning or information retrieval, depending on the user profile representation that is desired. Techniques usually used to contruct profiles are keywords profiles, semantic network profiles and concept profiles. Updating a user profile can be done automatically or manually. Normally, people use automatic methods beacuse are less intrusive to the user. On the first step, the system should gather the information of a single user. That information can be obtained in two ways: explicitly or implicitly.

Breu [Breu et al., 2008] suggest that a user profile can be represented as a probabilistic network. A probabilistic network provides a formal foundation for probabilistic inference. More importantly queries involving any subset of terms (attributes) may be posed to the network. Once the probabilist network is constructed, the document can be ranked according to the computed conditional probabilities. Such a network is learned from a sample of documents that are judged by the user to be relevant or non relevant.

Chapter 3

Intelligent Techniques

One of the difficult part of user profiling lies how to get all the information that matters from data. On this section, we will discuss some intelligent techniques for automatically creating user profiles coming from areas such as machine learning, data mining or information retrieval [Schiaffino and Amandi, 2009]. The purpose goes through discover patterns and behaviors and apply obtained knowledge to make decisions. There are three types of knowledge management systems: enterprise-wide knowledge management systems, knowledge work systems and intelligent techniques such as data mining, expert systems, neural networks, fuzzy logic, genetic algorithms and intelligent agents [Laudon and Laudon, 2012]. Some techniques will be presented for extracting and encoding knowledge from data. For representation data analysis there are rule bases, decision trees, and artificial neural networks and there are many techniques for data analysis such as density analysis, classification, regression and clustering [Heckerman, 2008]. The intelligent techniques discussed in this chapter are: bayesian networks, decision trees, case-based reasoning, association rules, neural networks and k-nearest neighbor algorithm. Each technique has one purpose: discovering knowledge, distilling knowledge in form of rules or discovering optimal solutions [Laudon and Laudon, 2012]. After this discussion, we will make a comparison between those techniques and see what sets them apart.

3.1 Bayesian Networks

Bayesian networks belong to the family of probabilistic graphical models that encode probabilistic relationships among variables of interest. Represents a probability distribution where nodes represents random variables, attribute or feature, and arcs represent probabilistic correlation between variables [Schiaffino and Amandi, 2009]. These graphical structures are used to represent knowledge about an uncertain domain [Networks et al., 2007]. Graphical models when with undirected edges are normally called Markov networks, that are popular in statistical physics and computer vision. According to Heckerman [Heckerman, 2008], this technique has, at least, four advantages for graphical model when used in conjuntion with statistical techniques:

- Encodes dependencies among all variables, even if some entries are missing.
- Can be used to learn causal relationships and predict consequences of intervention.
- Has both a causal and probabilistic semantics.
- Bayesian statistical methods in conjunction with bayesian networks avoids the overfitting of data.

Getting an example from Schiaffino, "A BN is built gradually as a given user queries the database. When a user submits a query, the query is stored in the form of a case and a node is added to the network for each attribute involved in the query. Arcs are drawn between the correspondent nodes, considering the relationships established for the particular domain. Probability values are updated as attributes frequencies in queries are modified with each new query. Each variable can have only two values: true, representing that the attribute is present in the query, and false, indicating that the attribute is absent.".

Naive Bayes

The simplest Bayesian classifier is Naive Bayes where it assumes that all attributes of the examples are independent of each other given the context of the class [McCallum and Nigam, 1998]. It's one of the most efficient and effective inductive learning algorithms for machine learning and data mining. Its competitive performance in classification is surprising, because the conditional independence assumption on which it is based, is

rarely true in real- world applications [Zhang, 2004]. According to Fleuren [Fleuren, 2012], a Bayesian network is something that cannot be done automatically, because the domain can have specific knowledge. This author has presented some advantages and disadvantages:

Disadvantages:

- It is difficult to Bayesian network make a decision if there is a lack of relevant information and the classification will be damaged;
- The domain should be specific to actually classify users into meaningful classes;
- Is something that needs to be done manually, particulary when variables are dynamic.

Advantages:

- Users can often be classified based on just a few variables;
- Use information that is easily gathered;
- It only needs a small amount of data to estimate the parameters, means and variances of the variables, necessary for classification because independent variables are assumed.

Some comprehensive comparison with other classification algorithms in 2006 showed that Bayes classification is outperformed by other approaches, such as boosted trees and random forests. Caruana [Caruana and Niculescu-Mizil, 2006] evaluate the performance of SVMs, neural nets, logistic regression, naive bayes, memory-based learning, random forests, decision trees, bagged trees, boosted trees and boosted stumps on eleven binary classification problems using a variety of performance metrics: accuracy, F-score, Lift, ROC arena, average precision, precision/recall break-even point, squared error and cross-entropy. With excellent performance on all eight metrics, calibrated boosted trees were the best learning algorithm overall. According to Caruana, random forests are close second, followed by uncalibrated bagged trees, calibrated SVMs, and uncalibrated neural nets. The models that performed poorest were naive bayes, logistic regression, decision trees, and boosted stumps. Although some methods clearly perform better or worse

than other methods on average, there is significant variability across the problems and metrics. Even the best models sometimes perform poorly, and models with poor average performance occasionally perform exceptionally well.

3.2 Decision Trees

A decision tree is a technique that can help making a decision as how to classify a new user. Getting a user variables from a dataset, a user can be classified by comparing to those in the tree. A great aspect of decision trees is that they, unlike Bayesian networks, can contain a diverse set of variables that are not way related. However, when there is a lack of information, the user will be classified by the majority of that variable. As Fleureu said [Fleuren, 2012], this is also known as a greedy algorithm, meaning that the best path down the tree will not necessarily be found due to a wrong turn based on a lack of information about a certain variable. This technique is commonly used in data mining. The goal is to create a model that predicts the value of a target variable based on several input variables. This model uses a set of binary rules to calculate the target result that can be used for classification (categorical variables) or regression (continuous variables). Two advantages of these methods is that offers simplicity of results and the tree methods are nonparametric and nonlinear. After this selection, different algorithms are used to find the best/indicate split method. Decision trees used in data mining are of two main types as described below:

Classification tree analysis is when the predicted outcome is the class to which the data belongs. They are used to predict cases or objects based on their dependent variables. Classification trees can have thousands of nodes and these need to be reduced to simplify the tree. When the model becomes too many complex, like having too many relative parameters to the number of observations, occurs overfitting. This happens because the criterion used for training the model is not the same as the criterion used to judge the efficiency of a model. Overfitting occurs when a model begins to memorize training data rather than learning to generalize from trend. As an extreme example, if the number of parameters is the same as or greater than the number of observations, a simple model or learning process can perfectly predict the training data simply by memorizing the training data in its entirety, but such a model will typically fail drastically when making predictions about new or unseen data, since the simple model has

not learned to generalize at all. This type of model, can handle problems with more than two classes and provide a probabilistic output [Criminisi, 2011].

Regression tree analysis is when the predicted outcome can be considered a real number. The main difference between regression and classification is that the output label to be associated with an input data is continuous, so the training labels are continuous. In terms of efficiency and flexibility, are both similar [Criminisi, 2011]. The terminal nodes, unlike classification tree, are predicted funtion values.

Some techniques, often called ensemble methods, construct more than one decision tree:

- **Bagging** decision trees, an early ensemble method, builds multiple decision trees by repeatedly resampling training data with replacement, and voting the trees for a consensus prediction. It is a relatively easy way to improve an existing method. One gains of this method is an increased of accuracy [Breiman, 1996].
- **Random Forest** classifier uses a number of decision trees, in order to improve the classification rate. Each Decision Tree is made by randomly selecting the data from the available data. According to Breiman [Breiman, 2001], the features are randomly selected in each decision split which reduces de correlation between trees and improves the prediction power and results in higher efficiency. Some random forests reported lower generalization error when comparing to other methods. For instance, random split selection [Dietterich, 2000] does better than bagging. There are some desirable characteristics presented by Breiman:
  - "Its accuracy is as good as Adaboost" [Freund et al., 1996] (the most common implementation of boosting) " and sometimes better";
  - "Its relatively robust to outliers and noise";
  - "Its faster than bagging or boosting";
  - "It gives useful internal estimates of error, strength, correlation and variable importance";
  - "Its simple and easily parallelized".
- **Boosted Trees** can be used for regression-type and classification-type problems [Hastie et al., 2001]. As Yang [Yang et al., 2005] concluded, the major advantages of boosted decision trees include their stability, their ability to handle large number

of input variables, and their use of boosted weights for misclassified events to give these events a better chance to be correctly classified in succeeding trees.

- **Rotation Forest** in which every decision tree is trained by first applying principal component analysis (PCA) on a random subset of the input features [Rodríguez et al., 2006].

3.3 Case-Based Reasoning

Case-based reasoning uses old cases to solve newer cases. It tries to remember similar situations and understand them with the objective to meet new demands. The solution pass not even for understand the similarities between two cases, but also what are the differences between those cases and create new solutions. But the question that needs to be made is: what happen when there are two different cases with the same result, but with different information? To anwser this problem, this method follows some steps to complete each case as showned on the next figure 3.1 [Fleuren, 2012].

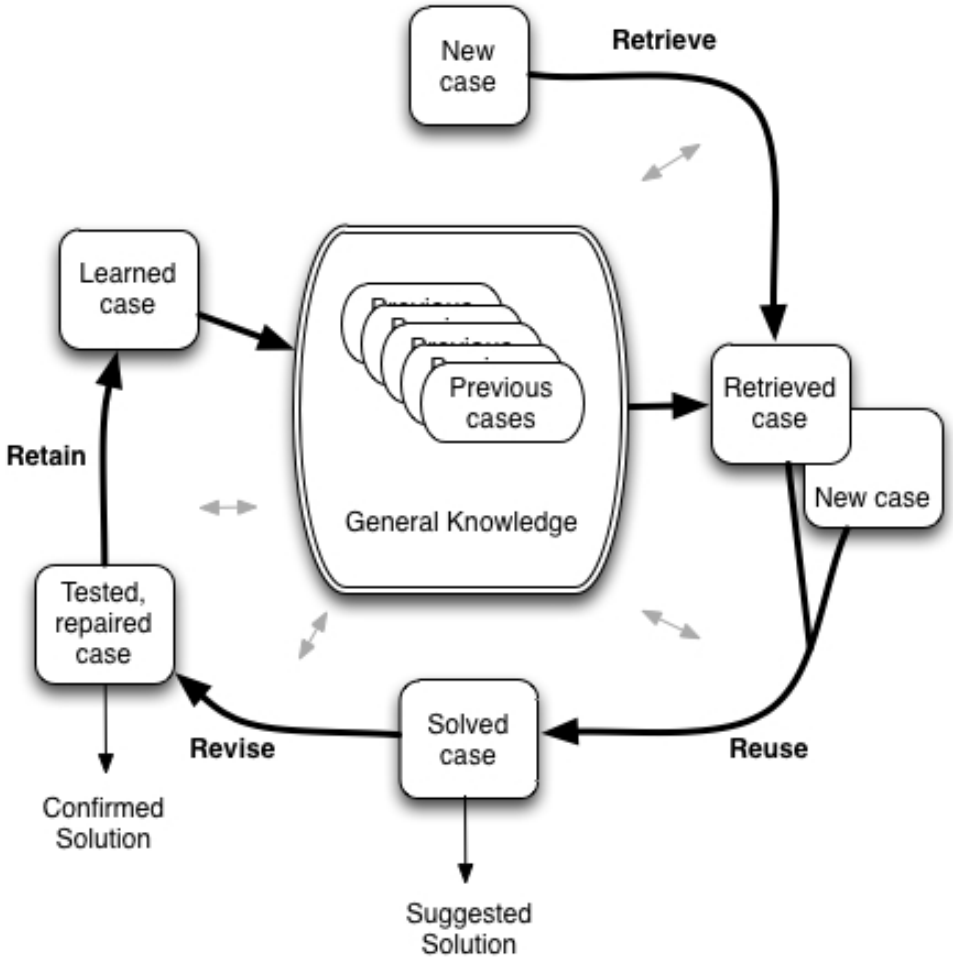


FIGURE 3.1: The Case-Based Reasoning Cycle.

The first step in CBR cycle is to RETRIEVE a similiar previous case which would be another existing customer with similar information. After this, the new case and the retrieve case are combined through REUSE into a solved case to find a new solution to the problem. The next step in the cycle is to apply this solution and to test its success through REVISE. The solution must be adjusted and repaired. When in Tested and Repaired Case step, the solution has been found. The final step in the cycle is to RETAIN the case and save it in the database of previous cases as a Learned Case [Fleuren, 2012]. Case-Based Reasoning is a technique that solves new problems by remembering older experiences [Schiaffino and Amandi, 2009]. Fleuren has an example where this technique can be used: "when a doctor has a patient with a certain combination of symptoms, he might remember another patient in the past that had the same kind of symptoms, and propose the same diagnosis. This type of reasoning can be applied for building user

profiles: when a new customer has a certain combination of interests, the CBR could look up what products customers with similar interests bought and propose these to the new customer". Though Bayesian networks and decision trees are good tools, they have some weaknesses such as use generalization to approach a new user. But when applying case-based reasoning it find a best fit solution, evaluation each case separately. Due to the incremental learning the system is directly able to apply newly learned cases to new cases [Fleuren, 2012].

3.4 Association Rules

Association rule learning is a data mining method to find relations between data [Tan and Steinbach, 2006]. For user profiling, this technique can be applied in some many ways. The task of this method is to find pairs of data that are, in some how, similar or complementary. Let's take an example: when a customer goes to the market buying some products, sometimes there are some connections between the products. Every year, usually at september, mothers take the children to go buy school supplies: notebooks, pencils, erasers, pens and a lot of stuffs. According to this routine, the system can create relations between the products to advise other customers about what they should buy and when they need to buy this kind of products based on customers behaviour. Having a large data sets, this method can suggest to the customers that when people buys notebooks, they usually take pencils, erasers and pens. The system not only suggest similar products but also complementary products. That's one great advantages of association rules. And the only think the user needs to do is to accept that advice or not. Taking this example, there is another funcionallity that the method can take from this kind of behavior. It can notify the customer, every september, that this is what he really needs at this time. So, the system make suggestion not only based on products, but also based on the season and time that some products is needed. This kind of system also have disadvantages. When they buy school supplies and because of something the customer buys a fridge, the system will create a relation between these things that have no relation. So the disadvantage of this method pass through the system may find association rules between products that are only related by coincidence. It is difficult to filter out these rules that seem to be nonsense. There are some different techniques for analyzing user data and producing information about a user from this

data. Some techniques rely a lot on data collected from previous users of a website and others rely more on the collection of information about a certain user [Fleuren, 2012].

3.5 Neural Networks

Neural networks was used to refer a network of biological neurons. They can be used to model complex relationships between inputs and outputs or to find patterns in data. A biological neural network consists of neurons that communicate with each by electrical signals. In an artificial neural network, the modern usage of the term, these are represented by nodes and connections between these nodes that is represented like the Figure Ėach connection has a weight that can be changed based on the outcome result [Fleuren, 2012]. This method is good in recognizing patterns. Getting on Fleuren example, "it can learn to distinguish between the letters A and B, by putting more weight on the horizontal bottom row pixels and the vertical left column pixels, and less weight on the middle horizontal row pixels. These are the pixels where the letters A and B are more or less distinguishable". Neural networks can be used to classify an user, using the gathered information of the user as input, into stereotypes based in certain assumptions [Chen et al., 2000]. Another example, "suppose there is a user of whom it is known that he has an expensive car and lives in an expensive neighborhood and that this person is classified in the stereotype rich. From this, one may infer that this person may also like to play golf, since this is also part of that stereotype. Of course, these assumptions are not always accurate" [Fleuren, 2012].

Neural networks can be really useful due to the fact that they can guess missing information in a user profile with quite a high level of detail. By applying stereotypes to a user, assumptions about a user will be made until the contrary has been proven. According to Fleuren, the best way of implementing this technique is when the website is able to identify the user with some certainty. This technique relies on the ability of building a profile over a longer period of time. There is the risk of users that share accounts or a single internet connection, taking the author example within a corporate environment, leading to mixed, inaccurate results. Another point of concern is that a neural network relies on feedback. As refered above, the weight of nodes are adjusted constantly based on the user outcome. The system only "receives positive feedback when a user spends a lot of time on a product's page or when he decides to buy a

certain product and negative feedback when a user repeatedly ignores a suggestion or spends very little time on a product's page" [Fleuren, 2012]. The information will be all distorted.

3.6 K-Nearest Neighbor algorithm

The K-Nearest Neighbor, or K-NN, algorithm is a method well suited for generating personalized query results. Is a non-parametric method for classifying objects based on closest training examples. This method preforms really great when in presence of a large amount of training set. Search results are personalized by comparing the current users profile to other user profiles and selecting the most similar one. Gemmell [Gemmell et al., 2009] describe how they use K-NN to suggest tags for a music piece a user wants to classify, based on the profile of the user. K-Nearest Neighbor algorithm is a method for classifying objects based on closest training examples, by a majority vote of its neighbors. Hall [Hall et al., 2008] said that the knn-nearest neighbor rule is arguably the simplest and most intuitively appealing nonparametric classification procedure. However, application of this method is inhibited by lack of knowledge about its properties, in particular, about the manner in which it is influenced by the value of k; and by the absence of techniques for empirical choice of k. K-NN is a very simple method to understant and implements. Delany [Delany, 2007] presented some advantages and disadvantages of this method:

Advantages:

- It's easy to implement and debug;
- Can be very effective if the output of the classifier is useful;
- There are some noise reduction techniques that work only for K-NN, improving classifier's accuracy [Delany and Cunningham, 2004];
- Can greatly improve run-time performance on large case-bases.

Disadvantages:

- Can have poor run-time performance if the training set is large;

- Is very sensitive to irrelevant or redundant features;
- May be outperformed by techniques like Support Vector Machines or Neural Networks on very difficult classification tasks.

3.7 Comparasion

In order to compare these techniques a model has been devised in which different aspects of the different techniques are compared, see Figure 3.2 [Fleuren, 2012].

	Bayesian Networks	Decision Trees	Case-Based Reasoning	Association Rules	Neural Networks	K-Nearest Neighbor
User Classification	x	x	x	x		x
User Profile Building					x	
Based on Learning			x		x	
Based on Statistics	x	x		x		x
Learn from Single Users	-	-	-	-	x	-
Learn from All Users	-	-	x	-	-	-

FIGURE 3.2: A comparison between different user profiling techniques.

3.7.1 User Classification and User Profile Building

The purpose of most techniques is for classifying a user into a group based on the choices each user make. Neural networks is a technique where the purpose is to build a profile around a single user, trying to learn as much about a user and improve the profile as it goas along. A reason why building a profile around a certain user is difficult is that it can be difficult to confirm a website users identity. Therefore profile building techniques are less suitable for website environments [Fleuren, 2012].

3.7.2 Based on Learning and Based on Statistics

There are a difference between those techniques that learn based on learning and those based on statistics. There is only one intelligent technique described here based on learning that is case-based reasoning, the others are base on statistics. Those techniques

based on statistics simply use past experiences and old data to apply on new data. As Fleuren [Fleuren, 2012] said, "a system can create and revise a decision tree from the data it is not considered learning. The system does not learn to make a better tree by choices it made in the past", but there are some techniques that depends on that decisions such as neural network. Association rules are based on statistics. Learning from the behavior of an user and the products selected by, the method can suggest new items to other user according to is behavior. Decision trees and Bayesian networks are techniques where the purpose is to calculate the chance of an user being from a certain class.

3.7.3 Learn from Single Users and Learn from All Users

There are only two techniques are based on learning. Case-based reasoning uses all the information gathered from the users. The system learns from old cases and tries to apply that knowledge to new cases. On the other hand, neural network only base it's work on a single user, trying to fill it with the right information. As Fleuren said "the system relies on stereotypes defined by other users, the network does not define these stereotypes itself" [Fleuren, 2012].

3.7.4 Differences between bayesian networks, decision trees and association rules

Having studied these three techniques, can be concluded that each one of them should be used not only because is the best method, but taking into account the problem that needs to be solved.

- **Bayesian networks** are commonly used for classifying users that handle incomplete data sets of information. Heckerman [Heckerman, 2008] refered some points that can give several advantages for data analysis. First when some data is missing, the model encodes dependencies among all variables. Second, can be used to learn causal relationships. Third, once having both a causal and probabilistic semantics, it's one of the best possible representation for combining knowledge and data. And finally, combining Bayesian statistical with Bayesian networks offer an efficient for avoiding the overfitting;

- **Decision trees** are very simple and have higher quality at classifying users on data sets with multiple variables. This method is specifically good in decision analysis, to help identify a strategy most likely to reach a goal. According to Fleuren, "also decision trees can make stereotypes visible, allowing the technique to be suitable as a complement to neural networks";

- **Association rules** "do utilize classification but to a different extent. Association rules are useful for finding products that the user is likely to respond to based on the products he is buying or viewing" [Fleuren, 2012]. It's a very popular and well researched method for discovering interesting relations between variables in a large data sets, identifying also strong rules between some information.

3.7.5 Comparing some Random Forest Decision Tree implementations

Some benchmarks have been made comparing different implementations of Random Forest. WiseRF^{TM} comes as version of the popular machine learning algorithm, Random Forest, and appears to resolve problems of scability. This implementation is fast, scalable, memory efficient and one of the most beloved machine-learning algorithms, Random Forests. Here some benchmarks between this implementation and other competitors.

WiseRF^{TM} vs scikit-learn

Richards [Joseph W. Richards, 2012] made in 2012 a benchmark between these two implementations and found that WiseRF^{TM} was a best solution. Richards presented a "data set that consists of 70,000 pixelated images of handwritten digits, from 0 through 9, each image measuring 28-by-28 pixels". To perform the comparison, he used 63,000 images as training data and a random 7,000 as testing data. Results can be seen below in Figure 3.3.

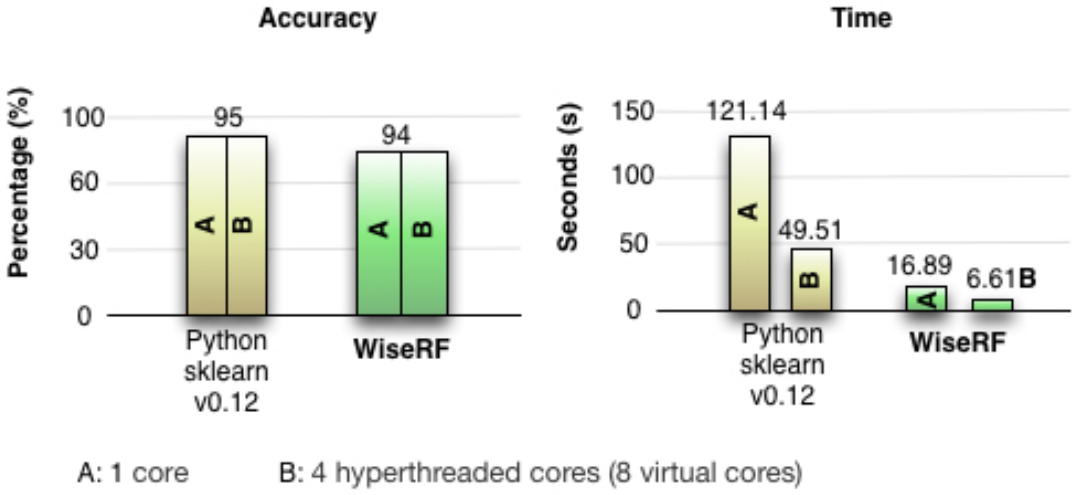


FIGURE 3.3: Benchmark between WiseRF^{TM} and scikit-learn.

On these results, can be seen that with a single core WiseRF^{TM} enjoys a factor of 7 boost in speed over scikit-learn with a comparable accuracy and with 4 hyperthreaded cores WiseRF^{TM} performs a 7.5x advantage in speed over scikit-learn. With these results, Richards [Joseph W. Richards, 2012] concluded that "wiseRF is at least 5x faster and sometimes as much as 100x faster than scikit-learns random forest, with the factor improvement depending on the number of trees and number of cores used for training".

WiseRF^{TM} vs weka vs R vs scikit-learn

The benchmark presented by the official WiseRF^{TM} website [*WiseRF BENCHMARKS*, 2013], shows that with a dataset of 60.000 instances, 784 feature dimensions and 10 classes the accuracy from WiseRF is as good or better than the other implementations. Results can be seen below in Figure 3.3.

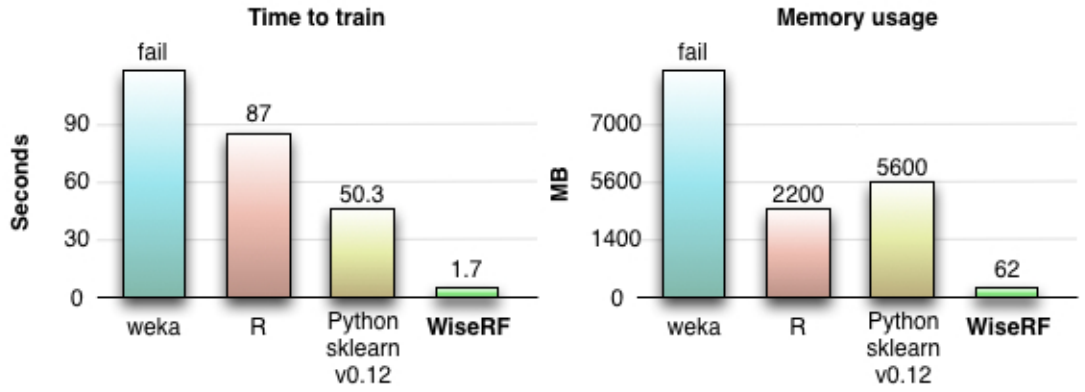


FIGURE 3.4: Benchmark between WiseRF^{TM} and other implementations.

Seeing these results can be detected that WiseRF^{TM} performs better than the others implementations. Comparing time train chart implementations, WiseRF^{TM} performs 29 times faster than Python sklearn, the second best result. Looking to the memory usage, WiseRF^{TM} uses, approximately, 97% less memory than R implementation, the second best result.

Chapter 4

Related Work

With this work, we intend to categorize users around the social networks. The work involved has as main objective the creation of user profiles. The data capture is crucial for the characterization of each user. Thus it is important to gather data sources to help us evaluate these users. An evaluation passes through a careful analysis of the characteristics of each user. The collected data are analyzed according the dimensions referred previously.

To categorize these users, the studies referred in this chapter followed some different approaches based on what they considered to be the approach that would give them the best results, based on some set of approaches that exists like Ad Hoc, Neural, Genetic Algorithm, among others. According to these cases of studies, there is a need to find the best approach to follow, based on the profile structure, and get the most relevant information. Having a way to get all the necessary information, there is a need to find a model that can separate all the keywords semantically unlinked and connected those that are semantically linked. Is required to have a strong base of information to that it takes effect immediately in the nearby future and determinate the profile of each user with the higher degree as we can get.

To achieve the best results, studies of different cases were done with the goal of understand what problems were found and how they resolved them. With these articles, some doubts needs to be answer. How do we get the information from the users? How we can separate all the keywords semantically unlinked and connected those that are semantically linked? How do we get the keyword weight? How we can create each user

profile based on their information? Accordingly, we analyzed a series of articles that follow in models that may be useful and have a purpose similar to the results pretended.

4.1 Analysis of User Keyword Similarity in Online Social Networks

The question that this article is trying to answer is: How do two people become friends? What role does homophile play in bringing two people closer to help them forge friendship? Is the similarity between two friends different from the similarity between any two people? How does the similarity between friends of a friend compare to similarity between direct friends? The goal of this study is to answer these questions, characterizing users profile entries and trying to find a similarity between a pair of users. On-line social networks (OSNs) helped them to study such problems using the set of rich data present about the users. A typical user profile in an on-line social network is characterized by its profile entries like location, hometown, activities, interests, favorite music, professional associations [Bhattacharyya et al., 2011]. The first topic to get their attention was about Keyword usage patterns. To measure the similarity between keywords and understand the usage scope of keywords as entered by different users in their on-line social network profiles; they analyzed Facebook profiles, considering only keywords that exist in the English dictionary.

After the capture, the questions raised were: How do we relate two keywords? How do we keep two keywords separated when they can not be related? so the real goal pass by clearly distinguish between related and unrelated keywords. Keywords can say to be related when they are semantically linked. Otherwise, they are unrelated and kept separated. To build a forest, they adopted a more ad hoc approach, allowing each keyword of a keyword pair to build its own tree. The next step passed through get the similarity between the users using all the Trees generated. To get the users similarity, there are three definitions to have in consideration. The first one is given to get the distance between two words, the second one to calculate the weak similarity between two users and the third one to determinate the strong similarity between two users. Then the keywords of user pairs were compared according to each of the heuristics defined above. This study allowed to say that this observation were significant because it shows

that users become more divergent in their interests to form new friendships, resulting in a decrease of similarity activities.

Briefly first they present results from the analysis on the number of keyword pairs the forest model was successful in matching. Second, showed results describing the variations in number of matches between keyword pairs and the variations in weak similarity and strong similarity for different number of keyword pairs between two users. Finally, the results are showing the variation in weak similarity and strong similarity based on different node degree of users and their individual number of keywords.

4.2 Intelligent User Profiling

This article aims to examine what information each user contains to create the exact profile. According to this, they face some issues: how the user profile is represented; how the user profile is acquired and built; and how the profile information is used [Schiaffino and Amandi, 2009]. A profile can be create getting the user information based on known qualities, based on what we consider to be the most important information or interesting facts about him or her. Having this, each user profile varies depending on the content that can be obtained. The user can explicitly provide all the information or it has to be learned using some intelligence approaches. This study indicates a variety of Artificial Intelligence techniques that have been used for user profiling such as case-based reasoning [Lenz et al., 1998], Bayesian networks [Horvitz et al., 1998], association rules [Adomavicius and Tuzhilin, 2001], genetic algorithms ([Moukas, 1997*b*], neural networks [Yasdi, 1999].

Determining user profiling is always a hard work and chooses the best approach to follow is decisive to obtain the real success. Commonly, user profile interests and information are keyword-based models. All the obtained information is represented by weight vectors of keywords that determine the importance of that word in comparison with other words. These representations are commonly used in the Information Filtering and Information Retrieval areas. Having a way to determine the importance of a keyword, getting the information becomes the next step. To respond to this, there are two alternatives, or the information is obtained in the implicit way, that is provided directly by the user, or implicitly, through the observation of the users actions.

Intelligent user profiling implies the application of intelligent techniques, coming from the areas of Machine Learning, Data Mining or Information Retrieval, for example, to build user profiles. The data these techniques use to automatically build user profiles are obtained mainly from the observation of a users actions, as described in the previous section [Schiaffino and Amandi, 2009]. In this article they present three techniques: Bayesian Networks, that represents a set of random variables and their conditional dependencies via directed acyclic graph, Association Rules that is a popular and well researched method for discovering interesting relations between variables in large databases and Case-Based Reasoning that is the process of solving new problems based on the solutions of similar past problems. Getting user profile content has been increasing interest in modeling users emotions in areas such as social computing and intelligent agents.

The challenges in this area pass through combine individual preferences into a group profile, determinate how to help users to reach some kind of consensus, and how to make group recommendations trying to maximize average satisfaction, minimize misery and/or ensure some degree of fairness among participants [Schiaffino and Amandi, 2009].

4.3 Inter-Profile Similarity (IPS): A Method For Semantic Analysis Of Online Social Networks

The method for semantic analysis of online social networks that this article said to be simple and efficient is called Inter-Profile Similarity (IPS). This method allows comparison of short text phrases even if they share no common terms. There is a short list of techniques for comparing users and this case of study devised a simple novel method that extends to compare short-text snippets using Natural Language Processing (NLP). They pointing two benefits for this usage: different words that possess the same meaning will be correctly identified and the number of terms in common decrease as the size of the vocabulary increases. They show that IPS yields both a larger range for the similarity values and obtains higher values than the intersection-based approaches. They present a set of benefits and current limitations of the IPS system [Spear et al., 2009].

The benefits are:

- Identifying similar concepts despite being expressed with different words
- Provides a total ordering over any set of users with regard to queries
- Handles phrases of varying length by ignoring words that do not match

The limitations are:

- Ignores negation
- The left-over words for phrases of varying length may be important

This method was applied to evaluate two popular social networks: Facebook and Orkut. They showed how similarity correlated with topological distance with various sub- grouping; part of this is validating the results from using NLP instead of the intersection based approach they utilized and part is extending said work with flow inside affiliations and across genders. On Facebook they concentrated only on the following categories: (1) activities, (2) interests, (3) gender, and (4) networks (affiliations). In Orkut they concentrated on the following categories: (1) activities, (2) passions, (3) sex, and (4) communities [Spear et al., 2009]. They showed that IPS is an option; a simple and novel extension to WordNet can be used to evaluate similarity of words, phrases and profiles.

4.4 You Are Who You Know: Inferring User Profiles In Online Social Networks

In this paper, they asked the question: given attributes for some fraction of the users in an online social network, can we infer the attributes of the remaining users? In other words, can the attributes of users, in combination with the social network graph, be used to predict the attributes of another user in the network? [Mislove et al., 2010]. To answer this question they gather an amount of data from two social networks and try to infer user profile attributes. They have found that two people with more attributes in common are more likely to be friends than the others.

A problem that still exists in getting users information is that users are allowed to mark their profiles as private. With this there is no longer possible to gather information

from that users. To get users information, they used Facebook crawls but this only works if users havent changed the default Facebook privacy settings. They evaluate their algorithm along with the algorithms of Bagrow [Bagrow, 2008] and Clauset [Clauset, 2005].

With this work and with the decisions they have made, they found that with as few 20 percent of users with known attributes, the remaining users attributes could be inferred with over 80 percent of accuracy. In their collected networks, they found that this algorithm is able to infer multiple attributes with high accuracy when given a few users with a common attribute. After the analysis of these cases, a set of models and approaches were found that can be a way to obtain the results that is pretended to achieve.

4.5 Not Every Friend On A Social Network Can Be Trusted: Classifying Imposters Using Decision Trees

In this paper, the authors are trying to anwser the question: how many accounts are fake? The goal of this paper is classifying users as imposter or not using a decision tree implementation. People usually creates a Facebook account to share photos between friends, sharing his thoughts, talking with known and unknown people, making friends and many other things.

First they tried to find the motives that make these imposters creating these fake profiles. Fong [Fong et al., 2012] presented some reasons like purely for fun or prank but often the ultimate purpose behind bogus accounts is malicious. Based on CNET, Fong indicated that Facebook has 8.7% fake users and this percentage estimates to 83.09 million accounts. These numbers represents a serious security problem. Identifying the relevant features for the training data is crucial. The attributes considered by the authors are: age, gender, college degree, avatar photo, personal information in the profile, authentic pictures, advertisement, profile completeness, number of friends, length of membership, gender of majority of mutual friends, comments on other posts between others. Each attribute is described in the article and are presented the reasons for each selection. For conducting this problem, they introduced five decision tree algorithms: J48, REPTree, RandomTree, ADTree and FT [Fong et al., 2012]. The dataset includes

both the specified real and fake users. Having this they performed some tests to each algorithm and the results presented are: J48 with 87.88% of accuracy, REPTree with 70.30% of accuracy, RandomTree with 75.15% of accuracy, ADTree with 90.30% of accuracy and FT with 92.12% of accuracy.

Classifying imposters on OSN has always been difficult and the efficacy has been validated by using empirical data collected longitudinally from the authors Facebook account, as a case study [Fong et al., 2012]. Their accuracies range varies from 70.3% to 92.1% depending of algorithms.

4.6 ISLab Project

This project aims to dynamically improve Collective Environments through Mood Induction procedures based on the user profile. All ambient certainly affects users condition on aspects such as stress, mood or fatigue, without affection indicators like productivity, quality of work, quality of life, personal/group performance or even health. With this work, it is pretended that based on behavior analysis of users, adapt its conditions to improve particular indicators. And the questions putted in here are: how can we get that users information? How can we determinate each user profile without being intrusive? This thesis aims to help this project on determine the user profiling just based on the data collected from the user, with less explicit user feedback. In a first place, ISLab project must be able to categorize user profile and change the ambient according to its profile and affect the mood of the users in order to improve their state.

Chapter 5

Implementation

This chapter will present the machine learning technique used and all the process until get the final results. It introduces the different steps passed until arrive the solution and discusses certain security and privacy considerations having regarded the privacy of users. This process has three main phases that will be described below.

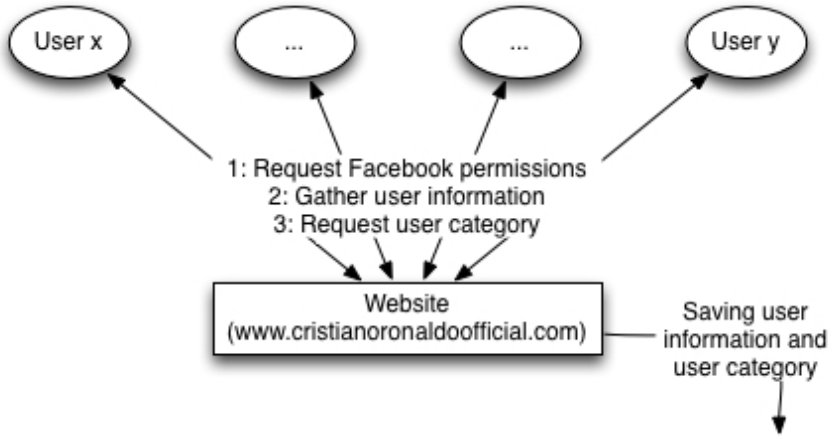


FIGURE 5.1: Getting Authorization And User Information.

Before start to determine the profiles, there were a set of tasks to be done. The first task passed through understand what information can be gathered from each user without compromising their privacy and any legal effect. Because of this, all data gathered in this study were taken by authorized users and they needed to accept our request to access their private content. There must be a decision about what kind of permissions will be requested, and what information will be relevant. Following the Figure 5.1 represented

above, first were requested permissions to the user, gathered his information and then were requested a category, selected by himself, that characterizes him. The information gathered from each user were the users' 'likes', and that was the requested information for each one. All the 4535 users came from `http://www.cristianoronaldoofficial.com/`. After they granted access to the private information, they selected one category in a list of 20 categories (Figure 1.2). Only 12 categories were selected on that list. The result is presented in Figure 5.2.

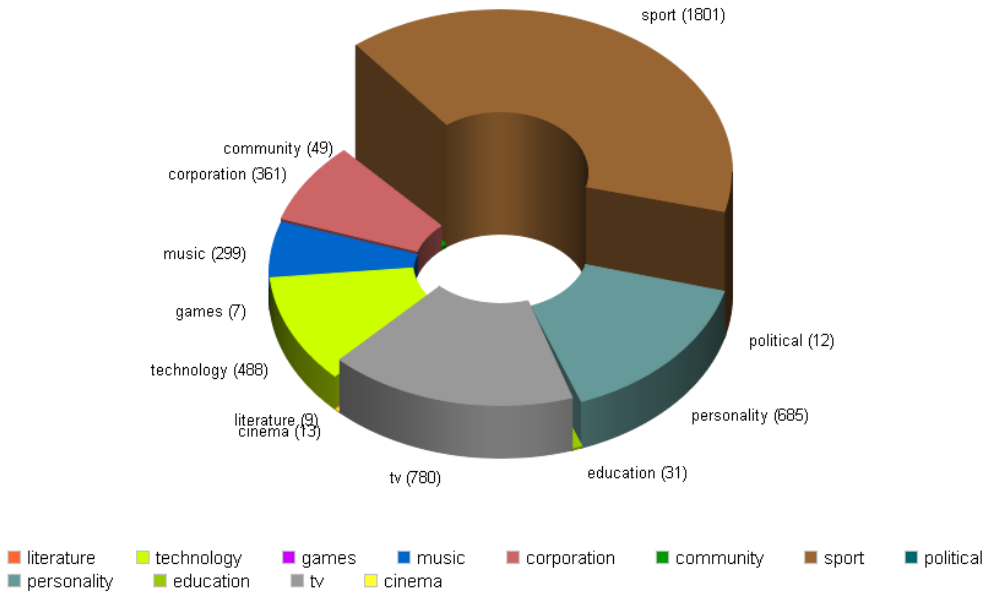


FIGURE 5.2: Users' Trainning.

The reason a user has chosen a specific category may be related to what users like, and it is on this type of relationship that will work. Each 'like' has one Facebook category associated as "Musician/band", "Artist", "Video game", "Tv show", "Public figure", "University" and much more (e.g. on Figure 1.1, Gareth Bale has "Athlete" as category). But the information gathered from 'likes' is not enough. When the user provided it's personal information about what he really liked, the information gathered was insuficient, containing only four fields for each like: "category", "name", "created time" and "id". There are other information that is considered relevant for this case like "how much likes this category has" or "if he really talks about this kind of content".

This kind of information can be indispensable to understand why a specific user selected that specific category. To get a more detailed information, Facebook provides an API to make a new request about one specific ID as seen in Figure 5.3.

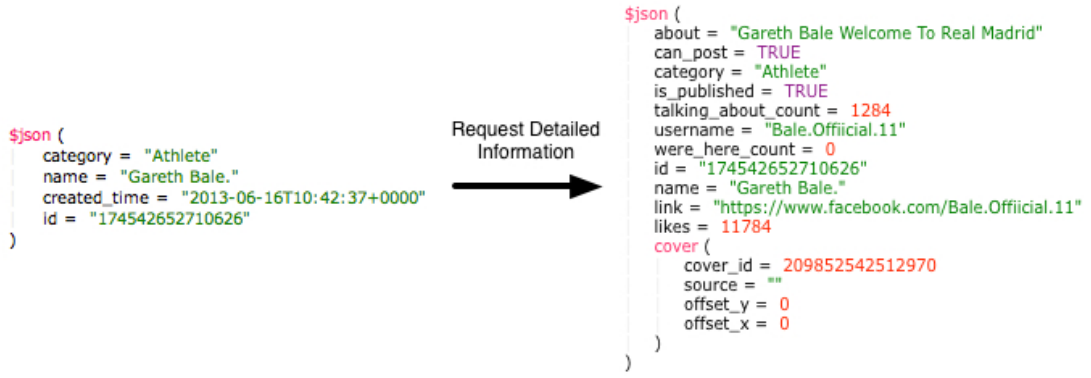


FIGURE 5.3: Request Detail Information Using 'ID'.

As can be seen, there is more detailed information on this example. Reaching this, the first step involves going through all users' 'likes', from training, and collect all detailed information associated to each 'like'. To collect the detailed data automatically from each like, was developed a PHP script to perform requests automatically and save the new content. All users' IDs and likes were saved into a database as well as the category chosen by each user. Having this data saved, the script follow some steps. First, it connects to the database to collect each user's ID and likes. The script iterated, user by user and like by like, took the like's ID and made a request to receive the detailed information about it. After receiving the response, the content of each like was replaced for this new response and saved into a file whose name is the user ID. The result of this can be seen in Figure 5.3.

Inside each user's likes, can be found dozens of categories. Because of this, each one of Facebook's category was included into a specific major category. All subcategories were grouped depending on its subject. Those which are related in some how were put together. Were only represented, with a small example of subcategories, the list of categories that were selected by the users, as seen in Figure 5.4.

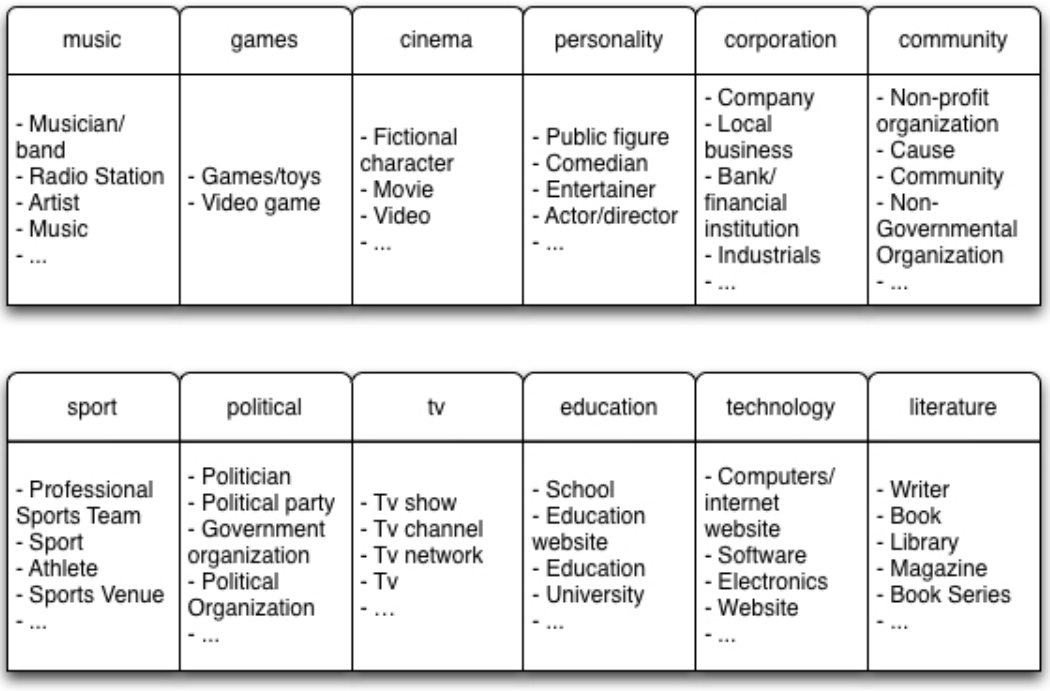


FIGURE 5.4: Distribution Of Facebook Categories.

Each category created with Facebook subcategories can have one meaning, e.g. corporation can be represented as a person that has an eCommerce business, only sell online and only through your own website or simply an online business or store; personality is a person that usually focused on and/or promoting an artist; music is a form of activity that holds the attention and interest of an audience, among others. Briefly, the aim of this process pass through gather the information about each user. This process involves the following steps:

- Creates a struture inside `http://www.cristianoronaldoofficial.com/` to allow the collection of user's data;
- Requests users' permissions to gather the Facebook's 'likes' of each user, as well as their IDs;
- Saves into the database a relation between the user ID, the information from each like and the selected category into the database;

After this process, once the data were stored, a script is responsible for two steps represented in Figure 5.5: collecting detail information about each 'like' and generate

the features that relates the user category and it's content. This is the first phase of the script, gather the detail information about each 'like' to generate all relevant features that will associate the users with some class of category. The process of getting detail information is represented in Figure 5.3. The second phase of the script will generate all features based on that information.

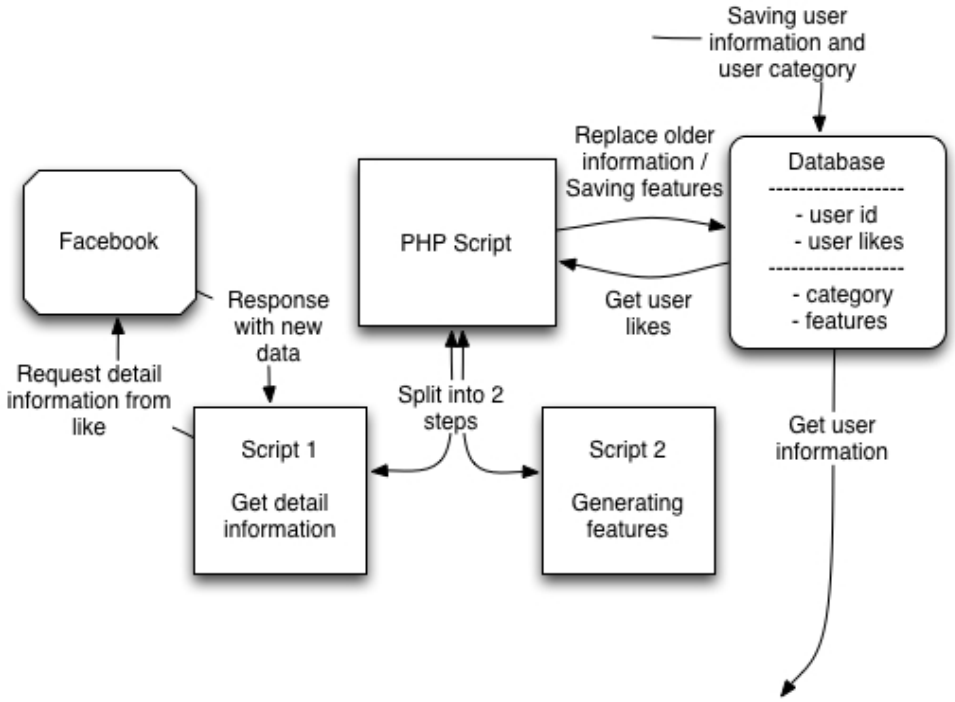


FIGURE 5.5: Getting Detail Information And User Features.

With all the information gathered from each user, the features that will be used to determine what each user represents in terms of class are:

- **count:** that will find how much 'likes' exists in each category;
- **likes:** that will determine how much people liked each category;
- **talking about count:** that will detect the numbers of people who really like and speaks of a certain category;
- **weight likes:** having the information of each like distributed by categories, determining the weight of each category based on 'likes';

- **weight talking about count:** having the information of each 'talking about count' distributed by categories, determining the weight of each category based on 'talking about count'.

Briefly, the aim of this second phase pass through gather the most relevant information of each like and generate the features that will be used for the ML algorithm. This phase involves the following steps:

- Collects users' information from database, comming from phase one;
- Implemented a PHP script to obtain a more detailed information about each 'like' through the ID (Figure 5.3), automatically;
- Replaces old information associated to an user with those that were obtained by the script;
- Implemented a PHP script to generate the features associated to each category and, consequently, to an user;
- Saves into the database a relation between the user id, likes, features and selected category.

Before presenting the last phase, will be presented an overview about the machine learning algorithm used to resolve the problem presented: determinate the class of each user based on it's content.

An overview to WiseRF™

Machine Learning is concerned about constructing and studying systems that can learn from data. This branch of artificial intelligence is a powerfull set of tools trying to find complex patterns inside heterogeneous and high-dimensional data. As it will collect more data, algorithms are learning with these new instances, enabling a more accurate assessment. But it can be a problem when the algorithms starting going down. Many reasons can be found like memory limitations or a poor performance.

WiseRF™ comes as version of the popular machine learning algorithm, Random Forest, and appears to resolve this problem of scability. WiseRF™ is fast, scalable, memory efficient and one of the most beloved machine-learning algorithms, Random

Forests. Nowadays, with the high quantity of data, there is a need to answer more complicated questions and make more informed and accurate decisions without compromise performance and accurate assessment. Quoting Richards [Joseph W. Richards, 2012], "Random Forest is a highly accurate method for predicting a response variable of interest (e.g., if an email is spam) from heterogeneous input data (e.g., the sender, subject, and content of the email), and is widely regarded as one of the best ML tools around. It works by employing a set of training data with known response variable to discover an optimal set of rules that relate the high-dimensional input data to the response.". Older implementations of Random Forest cannot have the same performance of WiseRF^{TM}, where problems with speed and memory limitations were always present. Chapter 3 presents some benchmarkings comparing WiseRF^{TM} algorithm and other implmentations.

Made the overview about WiseRF^{TM}, and before presenting the final phase of the process 5.6, let's make a summary about what was made before the implementation of the algorithm. First, was constructed a structure to gather the relevant information of each user presented on the training. That structure asked for certain permissions to avoid problems with privacy policies. However, the information collected about each 'like' from the users are incomplete and this led to the creation of a script presented on the second phase. This script was splitted into two steps: supplement the information gathered with the most relevant information and finnaly, prepare the features to the machine learning algorithm based on that information.

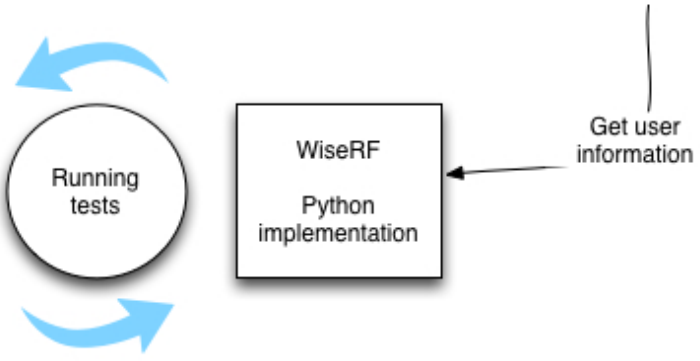


FIGURE 5.6: WiseRF^{TM} Implementation.

Having all of this process concluded and relevant information collected saved, it's time run the algorithm. To learn a prediction model on data and generate predictions

for future data the algorithm was all developed in Python. First, all data were obtained from the database and grouped into two arrays. An array of features and other of categories/classes/ground truth selected by the users. These arrays contain only floats. The features are represented as matrix where each line represents an user and each column represents the features 'count', 'likes', 'talking about count', 'weight like' and weight talking about, for each category. All categories presented in Figure 1.2 are represented by one number from one to twenty. This example of representation can be seen below in Figure 5.7, where each user will remain anonymous.

user id	Category 1 - count	Category 2 - count	Category 3 - count	...
1??????4	0	5	0	...
1???????0	2	2	0	...
1????????6	0	0	0	...
5????????7	4	9	3	...
5????????4	1	5	0	...

C10 - weight_likes	C11 - weight_likes	C12 - weight_likes	...	ground truth
0.016022071051963	0	0.0037667335362004	...	6
0	0.012977694239796	1.3097973942042E-6	...	5
1.4293139615086E-6	0	0.0014049330340874	...	6
0	0.12132875026493	1.9225398023681E-6	...	5
0	0.2241554256125	0	...	14

FIGURE 5.7: Features And Ground Truth By User.

To measure the algorithm accuracy, the algorithm was applied to each user and compared the results achieved with the known results/ground truth. Each time the algorithm is executed, one user is removed from the matrix as well as his ground truth from the array, and compared the result achieved with the result expected. The algorithm were tested with different estimators (40, 60, 80, 100 and 120), to find the best result. The optimal solution can be found with the estimator around 100. On a first stage, with around 4535 users, the best accuracy took 61.5%. This low accuracy happened because there are few users in some categories and can be improved by increasing the number of users. After boosting some categories, those who had worse outcomes, the accuracy increased to 92.2%. This mean that the more users you have, the better and more efficient results will be. WiseRF^{TM} Random Forest predictor parameters remain those default. These testing and evaluation are presented on the Chapter 6.

All these process presented above, can be seen below in Figure 5.8. Briefly and getting all the process until here, these three phases involves the following steps:

Phase one:

- Requests user permissions;
- Requests user category;
- Gather user likes as well as their IDs;
- Saves into the database a relation between the user id, likes and selected category.

Phase two:

- Collects users' information from database, comming from phase one;
- Implemented two PHP scripts: one to obtain a more detailed information about each 'like' through the ID and another script to generate the features associated to each category and, consequently, to an user;
- Saves into the database a relation between the user id, likes, features and selected category.

Phase three:

- Collects users' information from database, comming from phase two;
- Implemented the WiseRF^{TM} algorithm in Python;
- Generate an array containing each user features;
- Generate an array containing each user category;
- Run the Random Forest algorithm to each user, with the obtained information from previous phases, to determinate the algorithm accuracy;

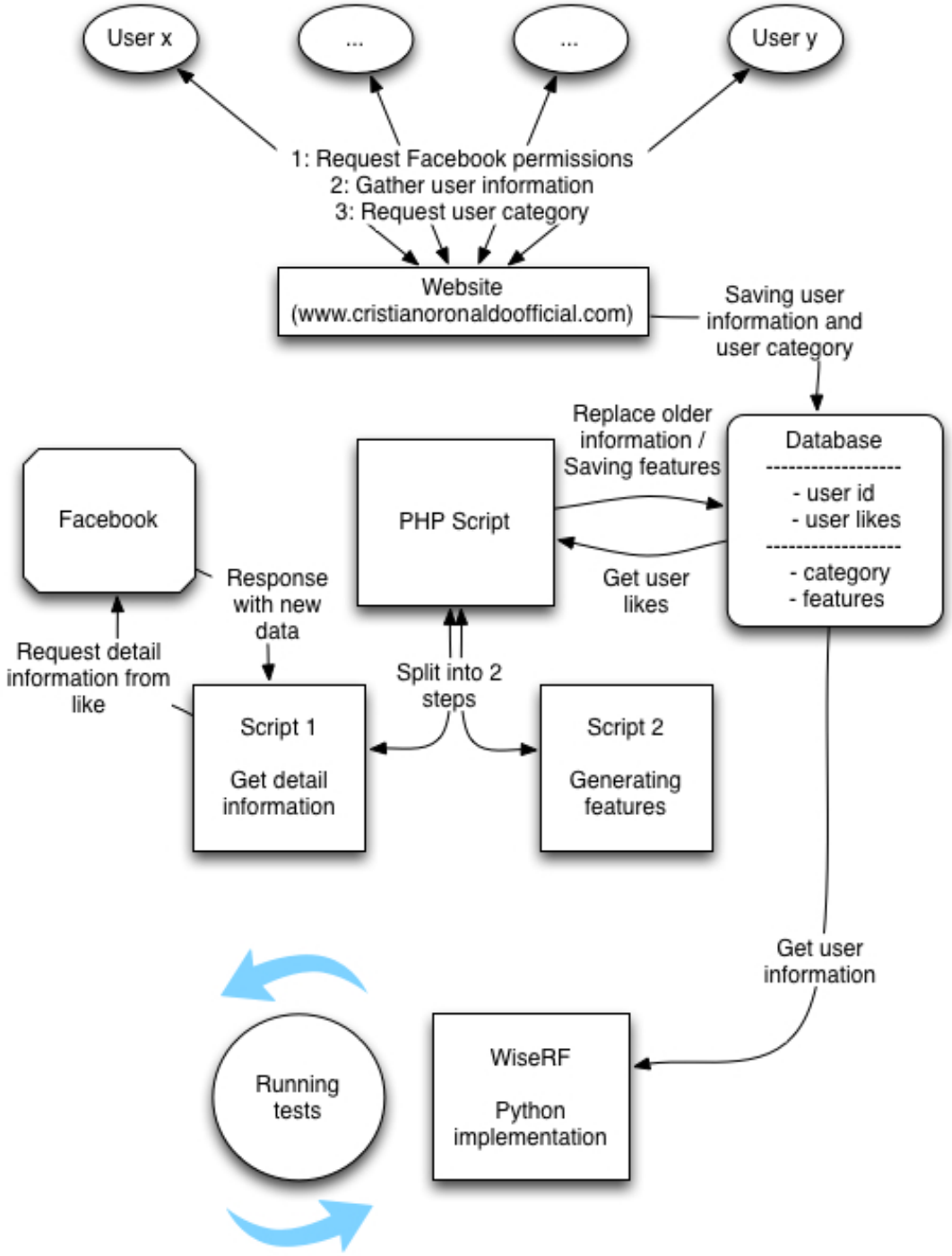


FIGURE 5.8: All Process.

Chapter 6

Testing and Evaluation

This chapter is devoted to testing and evaluation. Will be presented the results achieved and a explanation about that results.

As described on Chapter 5, before running the algorithm, much work was made like gather the relevant information and generate the features that will be associated with any decision made by the user, when he selected his category/class. The reason a user has chosen a specific category may be related to what users like. This collection of user information and generation of features corresponds to phase one and two, the process of getting authorization and user information is presented in Figure 5.1 and getting detail information and user features is represented in Figure 5.5, respectively. Having these two phases finished, the phase three pass through measure the algorithm accuracy. The algorithm were tested with different estimators (40, 60, 80, 100 and 120), after and before implementing boosting. The optimal solution can be found with the estimator around 100 in both cases.

WiseRF^{TM} Random Forest predictor parameters remain those default. To learn a prediction model on data and generate predictions the algorithm was all developed in Python. Lets start presenting the results for 4535 users before boosting in Figure 6.1.

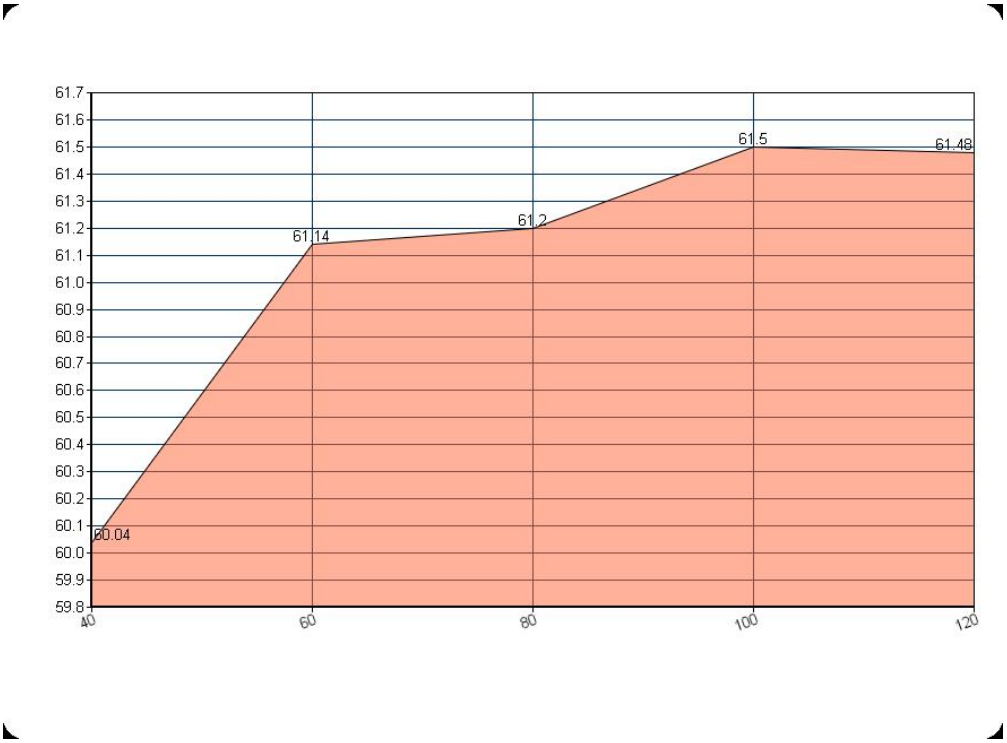


FIGURE 6.1: Estimator results before boosting.

As can be seen, the best precision value is presented with the estimator around 100, a precision of 61.5%. Reducing or increasing the estimator too much is not the key. There is a optimal estimator and you need to find it. There are some reasons that lead to these accuracies. There is always careful to note that the data collected need to have some relationship between the categories that users selected and the data associated with them. However, there is also the problem that the amount of data are not enough for the algorithm to learn. The results presented by WiseRF^{TM} Random Forest algorithm are too low. The categories precision have some poor results, more specifically, music and corporation. To understand the results achieved, can be seen in Figure 6.2 the accuracy obtained by category.

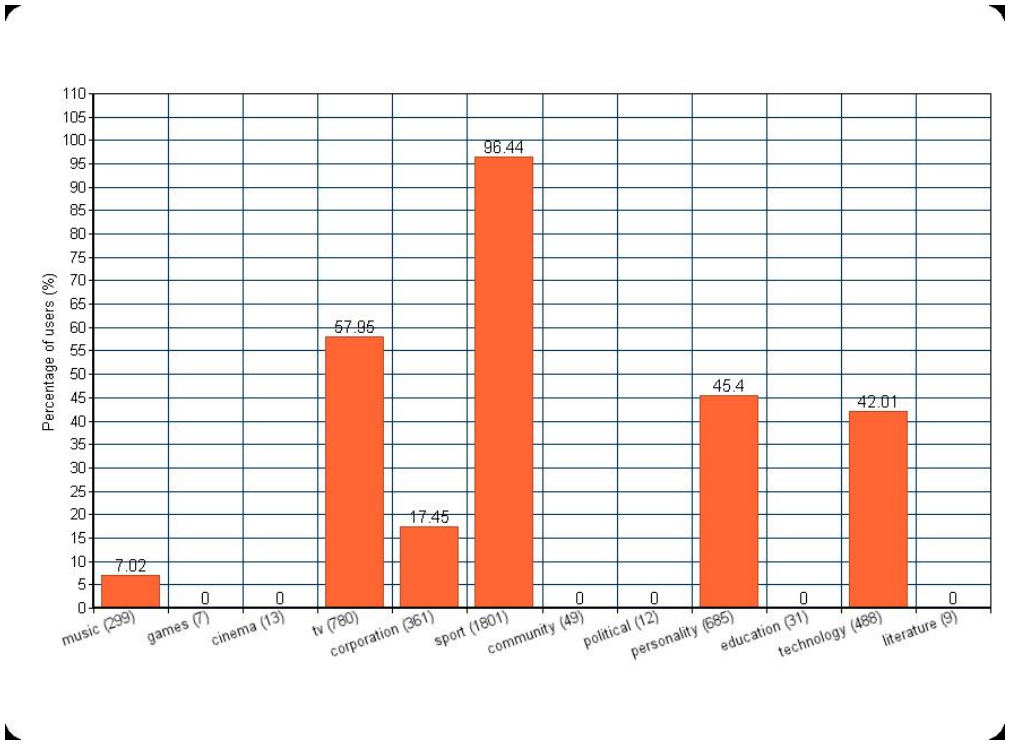


FIGURE 6.2: Categories results before boosting.

These results shows that categories with more users, tend to have a more accurate precision. According to this, users presented in categories with a low number of users, are not easily to classify because the training is poor. To solve this problem, was made a boosting to some categories that had few users for classification. Let's understand what boosting can make to change these results. Boosting one category is not only to increase the number of users on that category, but make the others categories understand that is not what the algorithm thinks. So lets make an explanation about what is the behavior of the algorithm to increase the accuracy on those categories that the precision is too low. All users presented on categories with low precision were duplicated. These users will not be tested but will be decisive to help the algorithm making a decision. Each time an original user appears to be classified, the copy associated will be removed from the training, or you knew exactly what would be it's classification. With this implementation the training will increase. Therefore the algorithm may still not know which category a particular user, but can determine which categories that user is not inserted, thereby increasing the probability of hitting in its category. These are the results for the algorithm with different estimator values, in Figure 6.3.

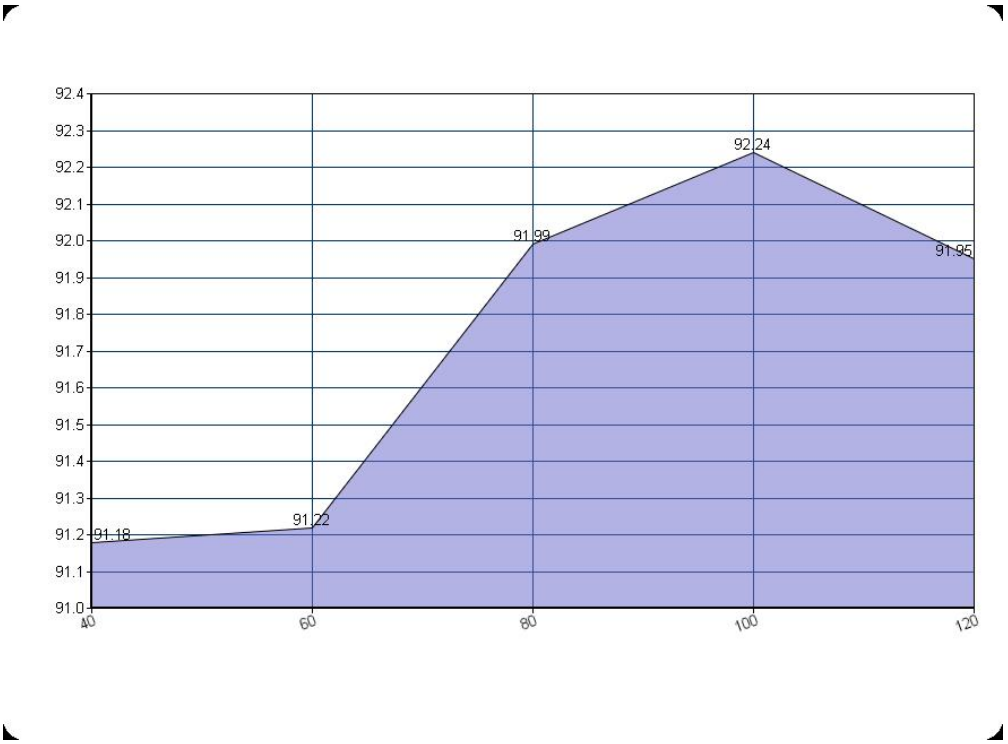


FIGURE 6.3: Estimator results after boosting.

The results are much better compared with those obtained before boosting. The best result is presented with the estimator equal to 100, with 92.24% of accuracy. The conclusion that can be made is that increasing the number of users of a certain category, will increase the accuracy of that category. Boosting the data ended up having influence on the algorithm decision at the time of assigning a category to a given user, helping the WiseRF^{TM} implementation making a better decision.

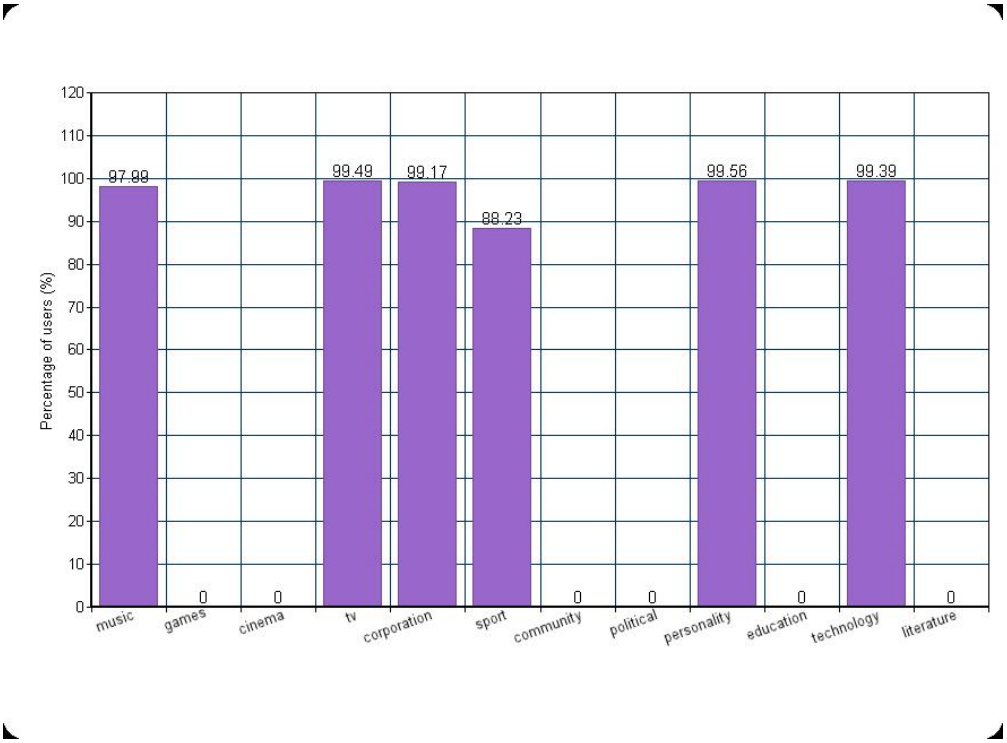


FIGURE 6.4: Categories results after boosting.

The decision of making boosting due to lack of training users, turns out to significantly decrease the error in the assignment of categories. only those categories that contain few users to workout is to eventually have a low precision, as expected. To solve these problems of having too few users in some categories and avoid a large discrepancy between the different categories, there is a need to achieve a training set of users so high that there is no shortage of users and low accuracy in these categories. Boosting was a necessity due to lack certain categories of users, more specifically, music and corporation. But the results achieved with the information gathered proved to be very good.

Chapter 7

Conclusion and future directions

This chapter aims to concluding the work that was developed and identify some possible directions of future work.

The focus of this thesis passed through identify user profiles within various SNS, more specifically Facebook. The problem that has been solved by collecting a certain private information of Facebook users. The information collected was the users 'likes'. According to the study that was made, can be concluded that there is no one best algorithm to identify user classes according to the information gathered, but each algorithm can be more efficient and powerfull depending on the problem that it is trying to solve. The comparison between some algorithms shows that Decision Tree Random Forest can have a great performance, more specifically WiseRF implementation. When comparing to others random forest implementations, is as good or even better than the others.

The experiments conducted suggests that WiseRF algorithm presented a great accuracy. With the evalution tests shown can be seen that increasing the number of users training, the more accurate will be the result. In a first step, with a total of 4535 users, the solution obtained presented an accuracy of 61.5%. After increasing the number of users present in the classes that had worse results, we found that the accuracy greatly increased, to a value of 92.2%. To create a great data set structure, one the most important thing was find the largest number of trusted users without ever adulterate the results.

Looking for the future work, there are too many things to do to increase the accuracy and reliability of the results. This implementation pass through gather some private

information of users and determinate in what class it is inserted. But this can go further. On a next phase, the work can go through analyze what the user usually share and verify in what category type is inserted such content. However, this decision can not just pass by identifying what he shared, but also with what intention the user shared certain content. Thus, it should also be possible to analyze what type of comment was used to identify that shared content and determine if he likes or dislikes it.

Bibliography

Adomavicius, G. and Tuzhilin, A. [2001], 'Using data mining methods to build customer profiles', *Computer* **34**(2), 74–82.
URL: *http://dx.doi.org/10.1109/2.901170*

Bagrow, J. [2008], 'Evaluating local community methods in networks', *Journal of Statistical Mechanics: Theory and ...* .
URL: *http://iopscience.iop.org/1742-5468/2008/05/P05001*

Belkin, N. J. and Croft, W. B. [1992], 'Information filtering and information retrieval: two sides of the same coin?', *Commun. ACM* **35**(12), 29–38.
URL: *http://doi.acm.org/10.1145/138859.138861*

Bhattacharyya, P., Garg, A. and Wu, S. [2011], 'Analysis of user keyword similarity in online social networks', *Social Network Analysis and Mining* **1**(3), 143–158.
URL: *http://dx.doi.org/10.1007/s13278-010-0006-4*

Bloedorn, E., Mani, I. and Macmillan, T. R. [1996], Representational issues in machine learning of user profiles, *in* 'Proceedings of the AAAI Symposium on Machine Learning in Information Access', AAAI Press.

Brandman, O., Cho, J., Garcia-Molina, H. and Shivakumar, N. [2000], 'Crawler-friendly web servers', *SIGMETRICS Perform. Eval. Rev.* **28**(2), 9–14.
URL: *http://doi.acm.org/10.1145/362883.362894*

Breiman, L. [1996], 'Bagging predictors', *Machine Learning* **24**(2), 123–140.
URL: *http://link.springer.com/10.1007/BF00058655*

Breiman, L. [2001], 'Random forests', *Machine learning* pp. 5–32.
URL: *http://link.springer.com/article/10.1023/A:1010933404324*

Breu, F., Guggenbichler, S. and Wollmann, J. [2008], 'A Bayesian Approach to User Profiling in Information Retrieval', *Vasa* .
URL: *http://medcontent.metapress.com/index/A65RM03P4874243N.pdf*

Caruana, R. and Niculescu-Mizil, A. [2006], 'An empirical comparison of supervised learning algorithms', *Proceedings of the 23rd international conference on Machine learning - ICML '06* pp. 161–168.
URL: *http://portal.acm.org/citation.cfm?doid=1143844.1143865*

Chen, Q., Norcio, A. F. and Wang, J. [2000], 'Neural network based stereotyping for user profiles', *Neural Computing & Applications* **9**(4), 259–265.

Clauset, A. [2005], 'Finding local community structure in networks.', *Physical review. E, Statistical, nonlinear, and soft matter physics* **72**(2 Pt 2), 026132.
URL: *http://www.ncbi.nlm.nih.gov/pubmed/16196669*

Criminisi, A. [2011], 'Decision Forests: A Unified Framework for Classification, Regression, Density Estimation, Manifold Learning and Semi-Supervised Learning', *Foundations and Trends in Computer Graphics and Vision* **7**(2-3), 81–227.

Delany, S. J. [2007], 'k -Nearest Neighbour Classifiers', pp. 1–17.

Delany, S. J. and Cunningham, P. [2004], An analysis of case-base editing in a spam filtering system, *in* 'Advances in Case-Based Reasoning', Springer, pp. 128–141.

Dietterich, T. G. [2000], 'An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization', *Mach. Learn.* **40**(2), 139–157.
URL: *http://dx.doi.org/10.1023/A:1007607513941*

Dumais, S., Joachims, T., Bharat, K. and Weigend, A. [2003], 'Sigir 2003 workshop report: implicit measures of user interests and preferences', *SIGIR Forum* **37**(2), 50–54.
URL: *http://doi.acm.org/10.1145/959258.959266*

Ellison, N. B., Steinfield, C. and Lampe, C. [2007], 'The benefits of facebook friends: social capital and college students use of online social network sites', *Journal of Computer-Mediated Communication* **12**(4), 1143–1168.
URL: *http://dx.doi.org/10.1111/j.1083-6101.2007.00367.x*

Fleuren, M. [2012], User Profiling Techniques: A comparative study in the context of e-commerce websites, PhD thesis, Utrecht University.
URL: *http://igitur-archive.library.uu.nl/student-theses/2012-0801-200525/UUindex.html*

Fong, S., Zhuang, Y. and He, J. [2012], Not every friend on a social network can be trusted: Classifying imposters using decision trees, *in* 'Future Generation Communication Technology (FGCT), 2012 International Conference on', pp. 58–63.

Freund, Y., Schapire, R. E. et al. [1996], Experiments with a new boosting algorithm, *in* 'ICML', Vol. 96, pp. 148–156.

Gauch, S., Speretta, M., Chandramouli, A. and Micarelli, A. [2007], 'User profiles for personalized information access', *The adaptive web* .
URL: *http://www.springerlink.com/index/y4g84202705577p3.pdf http://link.springer.com/chapter/10.1007/978-3-540-72079-9_2*

Gemmell, J., Schimoler, T. and Ramezani, M. [2009], 'Adapting K-nea-rest neighbor for tag recommendation in Folksonomies', *Proc of the 7th Intelligent Techniques ...* .
URL: *http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:Adapting+K+-Nearest+Neighbor+for+Tag+Recommendation+in+Folksonomies#0*

Gentili, G., Micarelli, A. and Sciarrone, F. [2003], 'Infoweb: An adaptive information filtering system for the cultural heritage domain', *Applied Artificial Intelligence* **17**(8-9), 715–744.
URL: *http://www.tandfonline.com/doi/abs/10.1080/713827256*

Hall, P., Park, B. U. and Samworth, R. J. [2008], 'Choice of neighbor order in nearest-neighbor classification', *The Annals of Statistics* **36**(5), 2135–2152.
URL: *http://projecteuclid.org/euclid.aos/1223908087*

Hastie, T., Tibshirani, R. and Friedman, J. J. H. [2001], *The elements of statistical learning*, Vol. 1, Springer New York.

Heckerman, D. [2008], *A tutorial on learning with Bayesian networks*, Vol. 1995.
URL: *http://link.springer.com/chapter/10.1007/978-3-540-85066-3_3*

Hoashi, K., Matsumoto, K., Inoue, N. and Hashimoto, K. [2000], Document filtering method using non-relevant information profile, *in* 'Proceedings of the 23rd annual

international ACM SIGIR conference on Research and development in information retrieval', SIGIR '00, ACM, New York, NY, USA, pp. 176–183.
URL: *http://doi.acm.org/10.1145/345508.345573*

Horvitz, E., Breese, J., Heckerman, D., Hovel, D. and Rommelse, K. [1998], The lumière project: Bayesian user modeling for inferring the goals and needs of software users, *in* 'Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence', UAI'98, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp. 256–265.
URL: *http://dl.acm.org/citation.cfm?id=2074094.2074124*

Joseph W. Richards [2012], 'Best-in-class Machine Learning from wise.io'.
URL: *http://continuum.io/blog/machine-learning*

Kelly, D. and Teevan, J. [2003], 'Implicit feedback for inferring user preference: a bibliography', *SIGIR Forum* **37**(2), 18–28.
URL: *http://doi.acm.org/10.1145/959258.959260*

Laudon, K. C. and Laudon, J. P. [2012], *Management Information Systems*, 12 edn.

Lenz, M., Hübner, A. and Kunze, M. [1998], 'Question answering with textual CBR', *Flexible Query Answering Systems* .
URL: *http://link.springer.com/chapter/10.1007/BFb0056005*

Lops, P., Gemmis, M. D. and Semeraro, G. [2011], *Recommender Systems Handbook*, Springer US, Boston, MA.
URL: *http://www.springerlink.com/index/10.1007/978-0-387-85820-3*

McCallum, A. and Nigam, K. [1998], 'A comparison of event models for naive bayes text classification', *. . . -98 workshop on learning for text categorization* .
URL: *http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.65.9324&rep=rep1&type=pdf*

Middleton, S. E., Shadbolt, N. R. and De Roure, D. C. [2003], Capturing interest through inference and visualization: ontological user profiling in recommender systems, *in* 'Proceedings of the 2nd international conference on Knowledge capture', K-CAP '03, ACM, New York, NY, USA, pp. 62–69.
URL: *http://doi.acm.org/10.1145/945645.945657*

Mislove, A., Viswanath, B., Gummadi, K. P. and Druschel, P. [2010], 'You Are Who You Know: Inferring User Profiles In Online Social Networks', *Proceedings of the third ACM international conference on Web search and data mining - WSDM '10* p. 251.
URL: *http://portal.acm.org/citation.cfm?doid=1718487.1718519*

Mladenic, D. [1999], 'Text-learning and related intelligent agents: A survey', *IEEE Intelligent Systems* **14**(4), 44–54.
URL: *http://dx.doi.org/10.1109/5254.784084*

Moukas, A. [1997*a*], 'Amalthaea information discovery and filtering using a multiagent evolving ecosystem', *Applied Artificial Intelligence* **11**(5), 437–457.
URL: *http://www.tandfonline.com/doi/abs/10.1080/088395197118127*

Moukas, A. [1997*b*], 'Amalthaea: information filtering and discovery using a multiagent evolving system', pp. 1–76.
URL: *http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.26.6435&rep=rep1&type=pdf*

Networks, B., Faltin, F. and Kenett, R. [2007], 'Bayesian Networks'.

Protalinski, E. [2012], 'Facebook has over 425 million mobile users'.
URL: *http://www.zdnet.com/blog/facebook/facebook-has-over-425-million-mobile-users/8384*

Quiroga, L. M. and Mostafa, J. [1999], Empirical evaluation of explicit versus implicit acquisition of user profiles in information filtering systems, *in* 'Proceedings of the fourth ACM conference on Digital libraries', DL '99, ACM, New York, NY, USA, pp. 238–239.
URL: *http://doi.acm.org/10.1145/313238.313409*

Rodríguez, J. J., Kuncheva, L. I. and Alonso, C. J. [2006], 'Rotation forest: A new classifier ensemble method.', *IEEE transactions on pattern analysis and machine intelligence* **28**(10), 1619–30.
URL: *http://www.ncbi.nlm.nih.gov/pubmed/16986543*

Salton, G. and Buckley, C. [1988], Term-weighting approaches in automatic text retrieval, *in* 'INFORMATION PROCESSING AND MANAGEMENT', pp. 513–523.

Schiaffino, S. and Amandi, A. [2009], 'Intelligent user profiling', *Artificial Intelligence An International Perspective* pp. 193–216.

URL: *http://www.springerlink.com/index/X1041G26Q1514851.pdf http://link.springer.com/chapter/10.1007/978-3-642-03226-4_11*

Sieg, A., Mobasher, B. and Burke, R. [2007], 'Learning ontology-based user profiles: A semantic approach to personalized web search'.

Spear, M., Lu, X., Matloff, N. and Wu, S. [2009], 'Inter-profile similarity (ips): A method for semantic analysis of online social networks', *Complex Sciences* pp. 320–333.
URL: *http://www.springerlink.com/index/r47q028u0x2x5361.pdf*

Tan, P.-n. and Steinbach, M. [2006], 'Introduction to Data Mining Instructor s Solution Manual'.

Teevan, J., Dumais, S. T. and Horvitz, E. [2005], Personalizing search via automated analysis of interests and activities, *in* 'Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval', SIGIR '05, ACM, New York, NY, USA, pp. 449–456.
URL: *http://doi.acm.org/10.1145/1076034.1076111*

White, R. W., Jose, J. M. and Ruthven, I. [2001], Comparing explicit and implicit feedback techniques for web retrieval: Trec-10 interactive track report, *in* 'Proceedings of the Tenth Text REtrieval Conference (TREC 2001'.

Widyantoro, D. H., Ioerger, T. R. and Yen, J. [2000], 'Learning user interest dynamics with a three-descriptor representation'.

Wilson, C., Boe, B., Sala, A., Puttaswamy, K. P. and Zhao, B. Y. [2009], 'User interactions in social networks and their implications', *Proceedings of the fourth ACM european conference on Computer systems - EuroSys '09* p. 205.
URL: *http://portal.acm.org/citation.cfm?doid=1519065.1519089*

WiseRF BENCHMARKS [2013].
URL: *http://about.wise.io/wiserf/*

Yang, H.-J., Roe, B. P. and Zhu, J. [2005], 'Studies of boosted decision trees for miniboone particle identification', *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* **555**(1), 370–385.

Yasdi [1999], 'Learning user model by neural networks', pp. 48–53.

Zhang, H. [2004], 'The optimality of naive Bayes', *AA* .

URL: *http://www.aaai.org/Papers/FLAIRS/2004/Flairs04-097.pdf*