

Konputagailuen Arkitektura eta Teknologia Saila
Department of Computer Architecture and Technology



Universidad Euskal Herriko
del País Vasco Unibertsitatea

INFORMATIKA FAKULTATEA
FACULTY OF COMPUTER SCIENCE

Combination of web usage, content and structure information for diverse web mining applications in the tourism context and the context of users with disabilities

Ph.D. Dissertation presented by
Aizea Lojo Novo

Supervised by
Olatz Arbelaiz Gallego
Javier Muguerza Rivero

Donostia, 2015

Konputagailuen Arkitektura eta Teknologia Saila
Department of Computer Architecture and Technology



Universidad Euskal Herriko
del País Vasco Unibertsitatea

INFORMATIKA FAKULTATEA
FACULTY OF COMPUTER SCIENCE

Combination of web usage, content and structure information for diverse web mining applications in the tourism context and the context of users with disabilities

Ph.D. Dissertation presented by
Aizea Lojo Novo

Supervised by
Olatz Arbelaiz Gallego
Javier Muguerza Rivero

Donostia, 2015

This work has been carried out with the grant of the Basque Government (EJ-GV) BFI-2010-106

*“Gizonen lana jakintza dugu ezagutuz aldatzea,
naturarekin bat izan eta harremanetan sartzea,
Eta indarrak ongi errotuz, gure sustraiak lurrari lotuz,
bertatikan irautea: ezaren gudaz baietza sortuz,
ukazioa legetzat hartuz beti aurrera joatea.”*

Xabier Lete

Irakurle zaren horri

Abstract

This PhD focuses on the application of machine learning techniques for behaviour modelling in different types of websites. Using data mining techniques two aspects which are problematic and difficult to solve have been addressed: getting the system to dynamically adapt to possible changes of user preferences, and to try to extract the information necessary to ensure the adaptation in a transparent manner for the users, without infringing on their privacy. The work in question combines information of different nature such as usage information, content information and website structure and uses appropriate web mining techniques to extract as much knowledge as possible from the websites. The extracted knowledge is used for different purposes such as adapting websites to the users through proposals of interesting links, so that the users can get the relevant information more easily and comfortably; for discovering interests or needs of users accessing the website and to inform the service providers about it; or detecting problems during navigation.

Systems have been successfully generated for two completely different fields: the field of tourism, working with the website of *bidasoa turismo* (www.bidasoaturismo.com) and, the field of disabled people, working with *discapnet* website (www.discapnet.com) from ONCE/Technosite foundation.

Resumen

La tesis doctoral que aquí se presenta se ha centrado en la aplicación de técnicas de aprendizaje automático al modelado del comportamiento en sitios web de diferente índole. Mediante técnicas de data mining se han afrontado dos de los aspectos problemáticos y difíciles de resolver en esta área: conseguir que el sistema se adapte dinámicamente a los posibles cambios de preferencias del usuario, e intentar extraer la información necesaria para poder realizar la adaptación de una manera transparente para el usuario, sin vulnerar su privacidad. El trabajo en cuestión combina información de distintas características como son la información de uso, información de contenido y la estructura del sitio web y las técnicas adecuadas de web mining para extraer la mayor cantidad de conocimiento de los sitios web. Conocimiento que se utiliza con distintos fines como adaptar los sitios web al usuario por medio de propuestas de links interesantes, de manera que pueda conseguir la información relevante de una manera más fácil y cómoda; descubrir intereses o necesidades de los usuarios accediendo al sitio web para informar a los proveedores del servicio; o detectar problemas surgidos durante la navegación.

Se han generado sistemas con éxito para dos ámbitos completamente distintos: el ámbito del turismo, trabajando con la página web de *bidasoaturismo* (www.bidasoaturismo.com) y el ámbito de las personas con discapacidad trabajando con la página web *discapnet* (www.discapnet.es) de la fundación ONCE/Technosite.

Laburpena

Ondorengo lerrotan aurkezten den tesi doktoralaren helburua, ikasketa automatikoaren bidez mota ezberdinetako webguneetan portaera modelatzea da. Datu meatzaritzako tekniken bitartez, alor honetako bi eremu problematiko eta konplexuei egiten zaie aurre: sistema, erabiltzailearen lehentasun aldaketa posibleetara dinamikoki egokitzea, eta egokitzapenak egiteko erabiltzailearen informazioa ateratzea ahalik eta modu garbienean, beti ere pribatutasuna errespetatuz. Lan honek ezaugarri ezberdinetako informazioa uztartzen du: webgunearen erabileraren informazioa, edukiaren informazioa eta webgunearen egitura, eta web meatzaritzako teknika egokiak erabilia webgunearen inguruko ahalik eta ezagutza zabalena erauztearaztea du helburu. Eraikitako sistemak helburu ezberdinetarako erabiltzen dira: erabiltzaileari lotura interesgarriak proposatuz webgunea egokitzeko informazioa baliagarria era errazago batetan jaso dezan; erabiltzaileen interes edo beharrak aurkitzeko eta zerbitzuaren hornitzailei horien berri emateko; eta azkenik, nabigazioan zehar sortutako arazoak detektatzeko.

Erabat desberdinak diren bi testuingurutarako arrakasta izan duten sistemak sortu ditugu. Turismoaren alorrean, *bidasoa turismoren* webgunerako (www.bidasoaturismo.com) eta adimen urriko pertsonen arloan, ONCE/Technosite fundazioaren *discapnet* webgunerako (www.discapnet.es).

Acknowledgements

Zaila da atal hau idaztea inor ahaztu gabe, hainbeste jenderi zor diot hemen aurkezten dudan lan hau... Baina tira saiatuko naiz inor ez ahazten.

Lehenik eta behin, mila mila esker nire bi zuzendariei. Eskerrik asko Olatz eta eskerrik asko Javi hemen aurkezten den lanean horrenbeste inplikatzegatik eta zuena balitz bezala hartzegatik. Plazerra izan da urte hauetan guztietan zuekin lan egitea eta etengabeko ikasketa batean egotea. Zuen laguntzarik eta aholkurik gabe hau ez zen posible izango eta.

Laborategiko kideei: oraingoei, Ainhoa, Igor, Lierni; eta pasatakoei, Carlos, Iñaki, Aritz, Oetzeta, Unai, zuei ere badut zer eskertua, lanaz haratagoko gaiak jorratzegatik eta nire kezkak entzun eta "jausten" nintzen momentu horietan altxatzegatik, azken etapa honetan batez ere. Aipamen berezia egin nahiko nioke proiektu honetako protagonista den Iñigori. Mila esker zuri Iñigo hainbeste laguntzegatik; beti behar izan dudan guztirako egoteagatik, aholku bat bestearen atzetik emateagatik, eta, mundutik oinutsik ibili daitekela erakustegatik eta beganoen mundua, eroen mundua ez dela erakustegatik. Egia esan esandako guztia gutxi da zuk lan honengatik emandakoarekin alderatuta. Handia zara! Bihotzez mila esker.

Mila esker ere ostiraleko "kafekideei" inoiz imajinatuko ez nituzkeen gai arraroenak ateratzegatik, oso une dibertigarriak pasa ditut zuekin, nahiz eta azken txanpa honetan gutxi konpartitu ahal izan ditugun.

Orokorrean ALDAPAKo kide zaretan guztieiei, esker berezia Ibai, tesiaren formatuarekin laguntzegatik.

Eskerrik asko ere *bidaso* *turismo*-ko zerbitzuaren hornitzaileari, gure ikerketa aurrera eramateko ezinbestekoak izan diren nabigazioaren datuak eskura uzteagatik, baita *discapnet* webguneko hornitzaileari ere.

I have to thank you, Richard, for allowing me to go to Angelu and for making me part of your computer science family. I have been very happy sharing with all of you 4 months of my working and personal life.

Sobre todo agradecer a la comunidad de habla castellana, Regi, Irvin, Joseba y Aritz. Nunca hubiera pensado que en cuatro meses se cogiese

VIII

tanto cariño y aprecio a la gente. Vosotros me habéis demostrado que soy parte de vuestra pequeña familia y que siempre estaréis ahí necesite lo que necesite igual que yo también lo estaría por vosotros. Espero que esta familia dure mucho mucho tiempo. Os agradezco todos esos momentos de risas hasta no poder más, no tanto los vaciles que me ha tocado sufrir. Aritz zuk aipamen berezi bat merezi duzu. Mila esker nigatik egin duzun guztiagatik, laneko erroreak zirela eta, zure lana utzi eta beti niri laguntzeko prest egoteagatik, nire psikologo pertsonal izateagatik (infinito ber infinito bajoietatik altzarazteagatik), aholkuak emateagatik, tesiaren zatiak irakurri eta zure iritzia emateagatik, Angelun jandako crêpe goxoengatik eta abar luze luze batengatik. Irundar guztiak ez zaretela berdinak erakutsi didazu.

Como no, tengo que agradecer a mis padres por aguantar todos los días mi nerviosismo, mis lloros, mis cambios repentinos de humor. Por obligarme día tras día a que cogiera unos momentos de descanso aunque no les hiciese mucho caso. Eskerrik asko AITA eta eskerrik asko AMA por estar ahí siempre que hace falta, ya sea para coger un ordenador y ponerse a trabajar conmigo, aún no sabiendo inglés, o bien para darme consejos y ánimo para seguir adelante. ESKERRIK ASKO. AMAIA ez uste zutaz ahaztu naizenik! Mila esker zuri ere liburutegira nirekin etortzegatik, zure konpainia oso lagungarria izan da niretzako. Eskerrik asko elkarrekin pasa ditugun momentu guztiengatik, ingelesezko zalantzekin laguntzeagatik eta orokorrean nik 4 urte nitunetik eta zu jaio zinenetik alboan eta gertu egoteagatik. Thanks sister. De paso darles las gracias a toda la familia y en especial a mi amona Lola por ponerme tantas velas para que me dieran suerte.

Kuadrilari ere eskertzen diot etxetik ateratzeko planak egiteagatik, eta noski liburutegitik atera eta Manun hartutako platanozko garagardoa eskuan pasatako momentuengatik. Baina bereziki eskerrik asko liburutegi team-ari.

Zuen txikitasunean handiak zaretelako mila esker Xalbat eta Mairi, nere bitamina izan eta marrazkiz marrazki eta dantzaz dantza gordetak zeuden irrifarrak atera dizkidazutelako. Baita aitatxo eta amatxo, eta amona eta aitonari ere.

Azkenik zer esan zutaz Alain, eskerrik asko nire "pataleta" guztiak jasateagatik, tesi honek emandako burukominak sufritu behar izan dituzun arren beti animatu nauzulako. Nire diseinatzaile grafikoa izan zara, eta agenda entsegu eta emanaldiz josita izan arren, nirekin korri egitera ateratzeagatik. Beste tesi bat idatzi ahalko nuke eskerrak emateko arrazoiekin, baina denak bi hitzetan laburtzen ditut: MAITE ZAITUT!!!

Contents

I	Introduction	1
0	Introduction	3
0.1	Motivation	3
0.2	Organization of the memory	7
II	Background information	9
1	Web personalization	11
1.1	Introduction	11
1.2	Data acquisition	12
1.3	Pattern discovery and analysis	13
1.4	Approaches for web personalization	16
1.4.1	Content-based filtering approach	16
1.4.2	Collaborative filtering approach	19
2	Web mining	23
2.1	Introduction	23
2.2	Categories of web mining	24
2.2.1	Web structure mining	24
2.2.2	Web usage mining	26
2.2.3	Web content mining	30
2.2.4	Combinations	32
III	Methods for web mining	35
3	Machine learning for web mining	37
3.1	Introduction	37
3.2	Learning algorithms in machine learning	38

3.2.1	Supervised learning	38
3.2.2	Unsupervised learning	39
3.3	Metrics used in machine learning	43
3.4	Performance metrics	45
3.5	Validation for machine learning models	47
3.5.1	Apparent error rate	48
3.5.2	True error rate	48
4	Content processing methods	51
4.1	Introduction	51
4.2	Data acquisition	52
4.2.1	Web content extractors	52
4.3	Content pre-processing phase	55
4.4	Document comparison techniques	55
4.4.1	Tools based on statistical methods	56
4.4.2	Tools based on ontologies	63
IV	Proposed systems	67
5	Application to a tourism context	69
5.1	Introduction	69
5.2	Related work	71
5.3	Motivation and overview of the work	72
5.4	<i>bidasoá turismo</i> website	75
5.5	Preliminary system	75
5.5.1	Data acquisition and pre-processing	76
5.5.2	Pattern discovery and analysis	79
5.5.3	Exploitation	82
5.5.4	Experiments: results and analysis	83
5.5.5	Summary	87
5.6	Global system	88
5.6.1	Navigation profiling	88
5.6.2	Enriched navigation profiling	95
5.6.3	Interest profiling	97
5.7	Preliminars of a new system	109
5.7.1	Enriched navigation profiling	110
5.7.2	Experiments: results and analysis	116
5.7.3	Summary	120

6	Application to a disabled people context	121
6.1	Introduction	121
6.2	Related work	123
6.3	<i>discapnet</i> website	125
6.4	Link prediction system	128
6.4.1	Data acquisition and pre-processing	128
6.4.2	Pattern discovery and analysis	129
6.4.3	Exploitation	131
6.5	Problem detection system	133
6.5.1	Data acquisition and pre-processing	133
6.5.2	Pattern discovery and analysis	138
6.5.3	Exploitation	141
6.5.4	Evaluation of the system	142
6.6	Summary	145
V	Conclusions	147
7	Conclusions and Further Works	149
7.1	Conclusions	149
7.2	Further Work	153
7.2.1	Tourism context: <i>bidasoa turismo</i>	153
7.2.2	Disabled people context: <i>discapnet</i>	154
7.3	Related Publications	154
	References	157
VI	Appendices	171
	Appendices	173
A	Web content extractor study	173
A.1	Web content extractor tools analysis	173
A.1.1	Mozenda	173
A.1.2	Web Info Extractor	175
A.1.3	Web Text Extractor	176
A.1.4	Web Data Extractor	177
A.1.5	Web Content Extractor	178

A.1.6	Wget	179
B	Seach engine study	181
B.1	Search engine tools analysis	181
B.1.1	MG4J (Managing Gigabytes for Java)	181
B.1.2	Indri	183
C	Keyword extractor study	185
C.1	Automatic keyword extractor tools	185
C.1.1	Kea (Key phrase extraction algorithm)	185
C.1.2	Keyword Analysis Tool	185
C.1.3	Keyword/Terminology Extractor	186
C.1.4	Maui	187
C.1.5	Topia Term Extractor	188
C.1.6	Yahoo Term Extractor	188

List of Figures

2.1	Web mining categories and examples of the used data.	24
3.1	Edit distance example	45
4.1	Illustration of the intuition behind latent dirichlet allocation	60
4.2	LDA Example: Selection of topic	62
5.1	Schema of the system architecture.	74
5.2	Appearance of the home page of BTw website.	76
5.3	Sample lines of <i>bidasoa turismo</i> log file.	77
5.4	Process of the navigation profile discovery.	80
5.5	The process of enriched navigation profiles.	81
5.6	Exploitation phase.	82
5.7	Division of the database.	83
5.8	Simulation of the real situation.	84
5.9	SP vs semantically enriched profile evaluation.	85
5.10	SP vs semantics enriched link prediction evaluation.	86
5.11	Example of BTw web page.	93
5.12	Number of URLs (X axis) per topic (Y axis)	94
5.13	The process of interest profiling.	97
5.14	Comparison of affinity levels for the whole database and the affinity levels of the profiles obtained with clustering.	101
5.15	Profile differences for the language-dependent profiles com- pared with global profiles.	104
5.16	Profile differences for the monthly profiles compared with global profiles.	107
5.17	Schema of the new system architecture.	110
5.18	The process of the semantic proximity calculation.	119
6.1	Appearance of the front page of the <i>discapnet</i> website.	126

6.2	Appearance of <i>Areas temáticas</i> within <i>discapnet</i> website.	127
6.3	Global approach to user profile discovery.	130
6.4	Modular approach to user profile discovery.	131
6.5	Architecture of the problem detection system	133
6.6	Problem discovery process and generation of the structure to be used in the automatic problem detection process.	138
6.7	Problem detection process for new users navigating the site.	142
A.1	Appearance of the Mozenda tool.	174
A.2	Appearance of the Web Info Extractor tool.	176
A.3	Appearance of the Web Text Extractor tool.	177
A.4	Web Data Extractor tool's output layout.	178
A.5	Web Content Extractor tool's output layout.	179
C.1	Interface of Keyword Analysis Tool.	186
C.2	Interface of Keyword/Terminology Extractor.	187
C.3	Interface of Maui.	188

List of Tables

3.1	Confusion Matrix	46
4.1	A summary of web content extractor analysis	54
4.2	A summary of the keyword extractor analysis	59
4.3	LDA example: document-topic relation	62
4.4	LDA example: word-topic relation	62
5.1	Semantics of some of the obtained profiles.	87
5.2	Number of requests and sessions after the different pre-processing stages.	88
5.3	Profile evaluation: average validation set results for the 10-fold cv.	90
5.4	topic-keyword list proposed by STMT and titles for each topic.	93
5.5	Evaluation of profiles and link prediction according to interests.	95
5.6	Evaluation of semantically-enriched link proposals according to interests	96
5.7	Sizes of the databases divided by language.	99
5.8	Sizes of the databases divided by months.	99
5.9	Comparison of average affinity values of the different topics for global and language-dependent profiles.	103
5.10	Comparison of average affinity values of the different topics for global and time-dependent profiles.	105
5.11	The navigation information we have.	113
5.12	Four different options for proposals	115
5.13	The rules of the four different options for proposals	115
5.14	Results of static and dynamic proposals	118
6.1	Evolution of the database.	129
6.2	Size of each zone within <i>Areas Temáticas</i>	131
6.3	Summary of the results.	132

6.4	Features extracted for problem detection and their description.	137
6.5	Statistics in the database for the extracted features.	137
6.6	Top 10 of the cluster with activated flags	140
6.7	Description of the extracted problematic profiles. Variables and conditions used to detect them, possible diagnosis and main source of the problems	141
6.8	Number of users with problematic profiles in the database and % detected analyzing only 25% of the clusters.	143
6.9	Summary of information of the experiment-based evaluation. Indicating the analyzed sessions (upper part) and problem detection rates for disabled users (lower part).	144

Part I

Introduction

Chapter 0

Introduction

0.1 Motivation

The Internet is known as a global system of interconnected computer networks that use the standard Internet protocol suite (TCP/IP) to link several billion devices worldwide. It is an international network of networks that consists of millions of private, public, academic, business, and government packet switched networks, linked by a broad array of electronic, wireless, and optical networking technologies. The Internet carries an extensive range of information resources and services, such as the inter-linked hypertext documents and applications of the World Wide Web (WWW), the infrastructure to support email, and peer-to-peer networks for file sharing and telephony.

Currently, almost every inhabitant of the world knows what the Internet is and, of course, what a website is. According to the survey done by Internet World Stats in 2014 [1], 70.5% of the European population, 87.7% of north Americans, 72.9% of Australians and in general 42,3% almost the half of the population of the world uses Internet.

The massive use of Internet has created a digital age where everybody is part of it. That is, now the population lives in a digital world. There is a big influence of technology on our daily life. Electronic devices (mobile phones, tablets, etcetera) and computers are things we have to deal with everyday; they are gadgets we cannot live without. Moreover, due to the 3G/4G network and the Wi-Fi network the Internet is everywhere and everyday in our pockets.

The advantages of the use of Internet are very notorious. Nowadays, a large amount of people use it for communicating and for keeping in touch with mates all over the world, even if they are far away from each other,

using chats and tools such as social networks. In addition, it is usable as well, as a source of information. Some years ago it was difficult to find information and it involved a significant dedication in libraries. Currently, thanks to the digital world everything is within some clicks.

Although everything related to Internet seems to be positive, it has negative points as well. The growth of the number of websites is huge, sometimes intractable to human capacity. In various surveys of the Web, e.g. Chakrabarti et al. [2], it was estimated that roughly one million new pages are added every day and more than 600 GB of pages change per month. In 2003, more than three billion web pages were available online; almost one page for every two people in the earth [3]. So, the amount of data is bigger and bigger and in order to make it valuable, the users must be able to find what they need, what can be a very difficult task nowadays. Currently, the term information overload is almost synonymous of Internet, referring to the large amount of information available in electronic format via the Internet and the inability of humans to consume it. Consequently, the task of the consumer of this content is increasingly more difficult, not only due to the need to assess the relevance of the information to the task at hand but also due to the need to assess the reliability and trustworthiness of the information available.

Many technologies have been researched to help people to reach or to find what they are looking for. For instance, information retrieval technologies have been developed and matured during this decade. Information retrieval is the name for the process or method whereby the users are able to convert their need for information into an actual list of citations to documents in storage containing information useful to them. Web search engines are the most visible information retrieval applications. Nowadays, there are for instance good search engines that do a good job indexing the content that is available through Internet. Search engines provide the users with a list of documents that are connected with the keywords they write as query. However, even if these kinds of tools are helpful, the information that search engines return is much more than what the user could possibly process. Once the users find the website about the topic they were searching for, it is not easy to skim the site in order to look for what they need within it, i.e. navigation within the site is not always easy.

So, added value to site visitors is achieved by providing easier access to required information at the right time and in the most appropriate form. That is the main goal of web personalization. Web personalization can be defined as the set of actions that are useful to adapt dynamically the presentation, the navigation schema and the contents of the Web, based

on the preferences, the interests, abilities or requirements of the user. The necessity of web personalization is even more crucial, for instance, in the case of using it for disabled people due to the fact that it will be the only way to break down technological barriers for them. Brusilovsky et al. [4] describe many research projects that are focused on the web personalization area, mostly in the context of e-commerce [4] and e-learning [5]. Important websites such as Google and Amazon are clear examples of this trend.

One of the possibilities for developing web personalization systems is the use of web mining techniques. Web mining techniques can be defined as the application of machine learning techniques to data from the Web. Normally for creating a web personalization system the user navigational information is used.

These systems can provide users and service providers with the most relevant information, more decision support, greater mobility and the most enjoyable navigation experiences. In any web environment, the contribution of the extracted knowledge from the information of user navigation in a site is twofold. On the one hand, it could be used for web personalization (i.e. using that information the site could be adapted to the users' requirements). On the other hand, for extracting knowledge of very diverse nature, for instance interest of the people browsing the system, people who is having problems within the navigation, possible design mistakes and so on, which will then be useful for the service providers.

So taking the idea of web personalization in mind, the aim of this dissertation is to build general web mining systems based on the combination of information from different sources, to work with any website in the two areas addressed in the previous paragraph, that is to improve peoples' navigational experience and to provide useful information of the user interests, navigation problems and so on to service providers.

More concretely the goal of the work described in this memory is to bring the benefits of personalization and web mining to different fields. In this case, two different fields are used to prove these benefits. The first one is a website of a tourism environment, whereas the second one is a website created for disabled people.

The first website (www.bidasoaturismo.com) was provided by the *bidasoa turismo* Destination Marketing Organization (DMO) (which is a tourism organization responsible for management the promotion of the destination of Bidasoa Txingudi bay). In the tourism environment the application of web personalization techniques is very important. Some years ago the first step of travellers for booking holidays was to find a travel agency in order to obtain information about different destinations, different offers, etcetera.

The digital age has revolutionized the tourism industry in the last decades. It is increasingly common to find travellers seeking information via the Internet for making any travel decision [6].

Gretzel et al. [7] introduced many years ago that the Internet is the main information source in the tourism environment. Steinbauer et al. [8] mentioned that the development of digital technologies has affected the tourism industry since the number of tourists that look for information online is increasing rapidly, there are more people nowadays looking for tourism packages online than in travel agencies. For this reason the DMOs should adapt to this trend and they must use their official websites in order to interact with travellers for promoting a destination or for providing information from it. For being able to adapt the offers to concrete users, they should extract information about them. This could be done in an invasive way, asking for information explicitly to the users or observing their actions such as using the navigation of the tourist in the web.

Web tourism personalization becomes essential and it can be positive for both the user and the business. As Apichai Sakulsureeyadej stated in [9], increasingly more travellers are expecting personalized products and services to meet their demands. Apichai mentioned that it is important for the tourism business to have tools that can store and monitor information in order to meet the individual needs of their clients. Literally according to [9], "The better you know your customer, the more likely you will retain them for longer period of time".

So the application of web mining techniques within this context will contribute to solve the tourism web personalization gap increasing the tourists' experience, making the website more interesting for them and of course for service providers which knowing the interest of the users of their site can offer interesting packages that fit better the user requirements.

The second website is *discapnet* (www.discapnet.es). It is created to promote the social and work integration of disabled people specially blind people or people with low vision rates. This website is financed by Technosite and the ONCE foundation. In this environment, the personalization of the web for adapting to people requirements is even more crucial for overcoming the technological barriers that disabled people have. *discapnet* is a website mainly aimed at visually impaired people. Visually disabled people normally make use of audio web interfaces in order to navigate through Internet, which is a time consuming and challenging task. Audio web interfaces read content in serial and as a consequence, it takes them 5 times longer than to a sighted users to have an overview of the site.

For this kind of webs the accessibility is a very important issue. Web accessibility means that people with disabilities can perceive, navigate and interact with the Web according to the Web Accessibility Initiative (WAI) [10]. But unfortunately being the website theoretically accessible, does not mean that the navigation through the site is going to be easy. Web personalization systems can be very helpful in this situations, for finding what the users need faster and more comfortably.

Furthermore the application of web mining techniques to the user navigational information could be used to discover problematic navigation profiles, what will be very valuable for the service providers in order to improve the structure of the site and as a consequence, raise the browsing experience of the users.

Taking these ideas into consideration, the work presented for *discapnet* website is a system that using web mining techniques, provides a tool for adapting the web to the user requirements and is able to automatically detect navigation problems. Consequently, it will contribute to improve user experience, and to overcome the technological barriers that disabled people have.

0.2 Organization of the memory

This memory has been divided into 6 different parts: Introduction, Background information, Methods for web mining, Proposed systems, Conclusions and Appendices.

Part I, Introduction, consists of a single chapter (the chapter where we are, Chapter 0) in which a brief introduction to the core of this thesis has been done, the use of different sources of information for web mining addressed to web personalization, recalling its motivation, as well as marking the main objectives of this thesis. In turn, a description of the structure of the memory, in which we are, is done from this point.

Part II, Background information has two different chapters. In Chapter 1, Web personalization, an introduction to web personalization environment is done. The main steps for the creation of a web personalization system are presenting. Even if there are more than one adaptation options we have centred the attention in the kind of web personalization more studied in recent researches, the recommender systems, presenting three different approaches for carrying out a recommender system: content based filtering, collaborative filtering and hybrid approach. Web mining techniques are the ones that are normally used for recommender systems and the web mining

environment is analysed in Chapter 2. In this chapter an introduction to web mining techniques is presented as well as the three main categories: web structure mining, web usage mining, and web content mining.

In Part III the methods used in web mining are explained. This part consists of two different chapters. In Chapter 3, Machine learning for web mining, is described, introducing the subject, reviewing the different learning algorithms centring the attention in unsupervised learning. The distances used in machine learning and the different validation methods are also analysed. Chapter 4, Content processing methods, provides information related to the processing of the content information, such as data acquisition, data pre-processing techniques and different options for similarity calculations between documents.

The main core of this thesis is presenting in Part IV. This part is devoted to describe the proposed systems. In Chapter 5 a system combining web usage mining and web content mining techniques is presented for the tourism environment. The system allows web personalization by link prediction and provides user interest profiles which are very valuable for the service providers. In Chapter 6 two systems are introduced: the first one is a link prediction system for helping disabled people finding what they are looking for. The second system presented, is a combination of the three different categories of web mining for navigation problem detection.

Finally, in Part V, Conclusions, the conclusions of this work are presented in a single chapter, as well as the open lines we hope to address in the future. After this chapter the referenced bibliography is presented and a series of appendices with additional information about different studies carried out.

Part II

Background information

Chapter 1

Web personalization

1.1 Introduction

Several studies indicate that users seem to find personalization on the Web useful and, according to Kobsa in his paper [11] people stay longer at personalized websites and visit more pages. Other researches demonstrate that personalization also benefits web vendors generating the conversion of visitors of a website into buyers [11], “cross-selling” [12], and customer retention and development [13].

A personalized website recognizes its users, collects information about their preferences and adapts its services in order to match the users’ needs. It improves the web experience of a visitor by presenting the information that the visitor wants to see in the appropriate manner and time. Brusilovsky et al. describe in [4] many research projects focused on the area.

The objective of web personalization is to provide users with what they need without requiring them to ask for it explicitly. That means that the system used for personalization must infer the preferences and needs of the user based on the current and past interactions with the system and must be able to adapt the website.

There are three main options to adapt a website [4]:

- Adaptation of the navigation by changing the navigation schema.
- Recommending the links that are estimated to be interesting for the user.
- Modifying the layout of the web pages in order to be adapted to the users’ preferences.

As mentioned at the beginning the purpose is to decrease the effort of the visitor making the site more comfortable to use. That is why the three options mentioned previously are so significant.

The first option tries to make more agile the process of selecting links and tries also to diminish the disorientation of the user. That could be for instance very helpful for people with slow selection capacity [14]. The second one can recommend links depending on the preference or needs of the user, making easier and faster the finding of what they are looking for, and finally, the last one allows to adapt the layouts to characteristics of the user.

Personalized systems use numerous techniques for making assumptions about users, such as domain-based inference rules, machine learning techniques, plan recognition methods, logic-based reasoning, and many more (see [15] for a survey). These techniques have different requirements regarding the data that must be available.

In order to create a web personalization system, some steps are compulsory according to Castellano et al. [16]. The first step is the data acquisition and the second step is the pattern discovery and analysis. These two steps are going to be analysed below.

1.2 Data acquisition

The information or data required to make a system capable of achieving the goal of web personalization can be collected implicitly or explicitly [17]. In both cases, the information must be assigned to a specific user. Although this might seem an easy task, it is complicated to relate a user with the data collected implicitly. In the case of explicit data collection the process of attributing the information to a single user is normally an immediate task.

Explicit collection normally asks the user for active participation. In this kind of systems a registration phase is used, where the users insert their personal information. The systems usually take the information using some forms fulfilled during registration or personal and financial information that users provide during a purchase.

Explicit collection has usually been implemented as rating of items and preference data inserted explicitly by the user among others. The first option, rating data may take the form of a discrete numeric value or using an unstructured textual form, for instance a review of products. In preference data, the users provide information to help the system with the prediction,

choosing which item or link could be convenient for them. In this case, it can take the form of keywords/product categories or values that describe the object. Explicit acquisition of user information has mainly one problem, users are generally not willing to provide information by filling in long forms.

There is an alternative option that avoids this type of problem: to implicitly acquire user information. The implicit collection of data refers to the data collected unobtrusively for the user. It is usually done analysing the action done by the user and the interaction with the website. For instance, Claypool et al. propose to use as an implicit indicator of interest in the object the time spent viewing an item/URL and its associated content [18]. In the e-commerce context there are several methods, for instance, taking into account the items selected in a shopping basket: addition of items, deletion of items, etc. Another example of an implicit collection of data is the information registered in a web server as a consequence of the activity or click-stream done by users. It can be extracted using cookies, server log information [19] and so on. It is also common to use Global Positioning Systems (GPS) or smart phones to acquire user information such as location, time and language [20]. In all these cases the user does not have to provide any personal information.

For both data collection options it is essential the pre-processing phase. Data is pre-processed to transform it into a format that is compatible with the analysis technique or pattern discovery process to be used in the next step. Pre-processing normally includes cleaning data from inconsistencies, filtering irrelevant information in accordance with the purpose of analysis and so on. For instance, in the case of server logs, they contain information not only directly related to the user activity but some information generated automatically as well. So, it is compulsory to find a method for identifying the entries directly related to user activity, separating those that are automatically recorded.

1.3 Pattern discovery and analysis

Generally, as it has been mentioned previously, the process of personalization is based on a data acquisition phase where the information about the users is collected and cleaned, and the pattern discovery and analysis phase where the user characteristics are discovered in order to adapt the website.

As it is commented in Section 1.1 in web personalization there are three main options to adapt a website: adaptation of the navigation by changing the navigation schema, recommending the links that are estimated to be

interesting for the user, and modifying the layout of the web pages in order to be adapted to the users' preferences. Even if the three options presented are important and useful, according to Leimstoll et al. [21] and Castellano et al. [16] one of the most used personalization technique is the use of recommender system. According to Hossein et al. [22] recommendation is becoming one of the most important methods to provide documents, URLs, merchandises, etc., as a response to user requirements. Following the criteria of these authors we are going to center our attention in recommender systems.

The main aim of the recommender system is to create significant suggestions and recommendations of the information, products or objects for users. Depending on the context the recommendation could be different, in e-commerce context, some products or objects are usually recommended; in the context of helping the users with their web navigation, normally URLs are recommended, etc. There are numerous examples of recommender system application, such as book recommendation on Amazon site and Netflix movie recommender, among others. These applications use recommender systems to identify users' tendencies and propose what they are looking for. According to Melville et al. [23] these systems attract users more and more.

There are different approaches which can assist recommender systems to create personalized recommendations, such as content-based filtering and collaborative filtering.

For understanding how these approaches work it is essential to explain some different learning techniques and some different manners for processing user data for being able to adapt the website to user requirements. The learning techniques can be classified into memory-based learning and model-based learning and the different manners for processing user data could be using only information from a single user (using only the individual information of a user for adaptation) or using information from multiple users (using information of a collective of people for the adaptation). The difference of the main web personalization approaches (content-based filtering and collaborative filtering) comes from the combination of learning techniques and the single/multiple user data.

- **Learning techniques**

Learning from data can be classified into memory-based learning (if the personalization process is done online while the system is performing the personalization tasks) and model-based learning (when the personalization is based on one offline stage where training data is used for

creating the model and one online stage for exploitation) [24].

Memory-based approaches basically memorize all the data and generalize from it at the time of generating recommendations. Therefore, these kind of systems are susceptible to scalability issues. Since predicting the rating of a given user on a given item requires the computation of the similarity between the given user and all its neighbours that have already rated the given item, its execution time may be long for huge datasets.

In order to reduce such execution time, model-based approaches have been proposed [7]. The general idea is to derive offline a model from the data in order to predict online ratings as fast as possible. Therefore, model-based approaches perform the computationally expensive learning phase offline and generally tend to scale better than memory-based approaches during the online deployment stage. These type of approaches use a two stage process for generating recommendations. The behavioural data that is collected from previous interactions is extracted in the offline stage and a model is generated that is going to be used in the online stage for future predictions. In the online stage the user starts the navigation in a website in a real time and combining the user interaction with the model previously built, the system is able to generate recommendations. This kind of application and techniques are less expensive computationally speaking than memory-based ones.

Most implementations of model-based systems appear in the context of web mining and implicit data collection. In this context the most used implicit data for building the model is the web navigation data of users recorded normally in the server (named web usage data), even if content and structure data is also used.

- **Single/multiple user data**

As it has been mentioned before, the personalization could be done using only the information of an individual user, taking into account only the information provided by the user or using only the track made by her, or, on the contrary, using the information of other users for the personalization step. In individual approaches, for instance in an e-commerce context, the system selects items based on the correlation between the content of the items and the user's preferences. In general this kind of systems use the history of each concrete user to build a profile that describes user characteristics. On the other hand, the

collaborative approach uses the information from multiple users and it is based on the following idea: if a person A has the same opinion as a person B on an issue, A is more likely to have B 's opinion on a different issue X than to have the opinion on X of a person chosen randomly. So, in this case the system estimates the needs and preferences of the users based on the experience acquired from other users with similar characteristics.

Below we will proceed explaining deeper the web personalization approaches previously commented such as content-based filtering approach and collaborative filtering approach.

1.4 Approaches for web personalization

In this section we are going to explain the main characteristics of the most extended web personalization approaches as well as some examples of real applications and the advantages and drawbacks of using each of them.

We will start with the content-based filtering approach, and we will continue with collaborative filtering approach.

1.4.1 Content-based filtering approach

Content-based filtering is normally based on memory-based learning techniques and it usually uses individual information in order to make recommendations. Content-based filtering approaches have the origin in information retrieval context and it is possible to find them in many fields of application, such as news, music, e-commerce, movies, etc. Most of the systems that implement a content-based filtering approach analyse web navigation trace or description items previously rated by a concrete user, and build an interest profile of the user based on the features of the information collected.

The user profile is a structured representation of the user interests, which will be used for the recommendation of new interesting items.

The process of recommendation basically consists of matching up the attributes of the user profile with the attributes of items. The result is a relevance judgement that represents the interest level that the user could have in that concrete item. If a profile accurately reflects the preferences of the user, is a big advantage for the content-based recommendation system due to the fact that the profiles could be used for filtering search results by deciding whether a user is interested in a specific item or not and, in the

negative case, preventing it from being displayed or display it in a second level.

Content-based filtering approaches need proper techniques for representing the items and producing the user profiles, and some strategies for comparing the user profiles with the item representation.

Items that can be recommended to the user are represented by a set of features, also called attributes or properties. For example, in a movie recommendation application, features adopted to describe a movie are: actors, directors, genres and subject matter among others. In most content-based filtering approaches, item descriptions are textual features extracted from web pages, emails or product.

There are many methods for the item representation step. Some examples are: keyword-based Vector Space Model (VSM), semantic analysis by using ontologies, semantic analysis by using encyclopedic knowledge sources and so on. The most common one is the use of the keyword-based VSM. In this case, the user profiles and new items, both are normally represented as weighted term vectors. So, the recommendation task in these kinds of systems usually involves the comparison of extracted features from unseen items with content descriptions in the user profile. This comparison usually is based on an exact string matching. In the case of semantic analysis, using ontologies or using encyclopedic knowledge, the main goal is not to pay attention only in the exact string matching but to the idea that different words can have similar meanings, emphasizing the semantic similarity.

What it appears as item and rates in the described examples could be directly translated into web page and time spent in the navigation context.

Many of the early adaptation or recommender systems are based on content-based filtering approaches specially in the web recommender area and they normally use keyword-based VSM. Some of the best known early recommender systems are: Letizia [25], Syskill & Webert [26, 27], Webmate [28] and WebWatcher [29] and in the following paragraphs they will be described.

Letizia [25] is a web-browser extension that tracks the user behaviour and builds a personal model based on the keywords related with the interest of the user. This application uses implicit feedback to infer user's preferences. For instance, when a user marks a web page as a favourite one it is a strong evidence of the preferences of the user. The WebWatcher application [29] follows the same criteria and builds the model or user profile based on the links the user has visited in a web page. In addition, it takes into account the links that the user has in the web pages but she does not use as a negative example or an example of information the user dislikes.

In the case of Syskill & Webert [26, 27] the application learns the profile from previously ranked web pages on a concrete topic to distinguish if the web page is interesting for the user or not. For learning the profile, Syskill & Webert represents documents with the 128 most informative words. Moreover, it uses a naïve Bayes classifier to predict the future and to classify the unseen pages as interesting or not for the user.

WebMate [28] is a personal browsing and searching agent. It accompanies the user when she uses the Internet, and provides her with information it gathers based on his user profile, which this created as user browses the Internet. The user profile in this case consists of keyword vectors that represent positive training examples.

Apart from the recommender systems analysed previously, specific recommender systems used in concrete environment exist as well. For instance, there are many content-based filtering approaches used in the field of recommending films or news like NewT [30], NewsDude [31] and a large etc.

There are some other applications that instead of using keyword-based Vector Space Model use semantic analysis by using ontologies or by using encyclopedic knowledge sources, for example, SiteIF [32], ITR [33], News@hand [34], etc.

1.4.1.1 Pros and cons of content-based filtering approach

For finishing the analysis of the content-based filtering approaches some advantages and disadvantages are going to be presented in the following lines.

The main advantages are:

- Single user based estimation. Content-based recommender systems exploit solely information provided by the active user to build her profile. It is not needed information about other users.
- Transparency. Explanations on how the recommender system works can be provided by explicitly listing content features or descriptions that caused an item to occur in the list of recommendations.
- New item. Content-based recommenders are capable of recommending items not yet rated by any user. comparing item's characteristics with user preferences.

But these systems have also some disadvantages:

- **Over specialization.** The primary drawback of such systems is their tendency to over specialize the item selection, since profiles are only based on the previous information of a single user. That means that if a user is looking for a different item than she used to find the system recommendations are not going to be appropriate for what the user is expecting.
- **Limited content analysis.** Content-based approaches have a natural limit in the number and type of features that are associated, automatically or manually, with the objects they recommend. Domain knowledge is often needed, e.g., for movie recommendations the system needs to know the actors and directors, and sometimes, domain ontologies are also needed.
- **Cold start problem.** Enough information of the user has to be collected before a content-based recommender system can really understand user preferences and provide accurate recommendations.

Some studies as the one presented in Sinha et al. [35] say that users find more useful the recommendation when they are unexpected recommendations. Consequently, the drawback of the over specialization is a serious problem of using content-based filtering approaches for recommender systems.

1.4.2 Collaborative filtering approach

Collaborative filtering is the process of filtering items using the opinions of other people. Goldberg et al. [36] first presented the collaborative filtering approach as an alternative to the content-based filtering approach. They introduced the idea of collaboration between people in order to help each other carrying out a filtering using their annotations. While the term collaborative filtering (CF) has only been around for a little more than two decades, CF takes its roots from something humans have been doing for centuries: sharing opinions with others, as it is said in Schafer et al. [37].

These approaches employ statistical techniques to find a set of users, known as neighbours, that have a history of agreeing with the target user (i.e., they either rate different items similarly, they tend to buy similar set of items or they are interested in semantically similar documents). Once a neighbourhood of users is formed, these approaches use different algorithms

to combine the preferences of neighbours to produce a prediction or recommendation for the active user.

Although CF can use other learning algorithms normally it uses model-based learning algorithms. Model-based algorithms, such as [38, 39] uses the rating done by the users or, in the case of the navigational data, the navigations done by the user to create a model. These models are used to predict ratings or links, respectively. The development of models (normally done using machine learning techniques) permits the system to learn and to identify complex patterns based on stored data. After that, the system could make intelligent predictions for new users, using the models built. The most used machine learning techniques in CF approaches are: Bayesian models, clustering models and dependency methods.

In the early 1990s, a manual collaborative filtering system was created Tapestry [36]. However, nowadays all the CF approaches are automatic, for instance, GroupLens [40] was one of the first research work which presented automatic CF. Resnick et al. used automatic CF approach in order to identify articles which are likely to be interesting to a particular user. In GroupLens users only needed to provide ratings or perform other observable actions; the system combined these information with the ratings or actions of other users to provide personalized results. With these systems, users do not obtain any direct knowledge of other users' opinions, nor do they need to know what other users or items are in the system in order to receive recommendations.

Collaborative filtering became a topic of increasing interest among human-computer interaction, machine learning, and information retrieval researchers, in the 1990s. This interest caused the production of a number of recommender systems used in different fields, such as Ringo [41] for music and the BellCore Video Recommender [42] for movies, among others.

In the late 1990s, the recommender system began to emerge in the commercial environment. Perhaps the most widely-known application of recommender systems technologies is Amazon.com. Based on purchase history, browsing history, and the item a user is currently viewing, they recommend items for the user to consider purchasing. Following the Amazon's adoption, many e-commerce and online systems have used CF. The motivation for applying this CF is to raise the amount of sales.

Some of the most widely used Collaborative Filtering (CF) approaches in the last years are [43, 44]. Some works derive user's preference to products by analysing user's navigational and behavioural data such as click-stream data [18, 45, 46, 47, 48]. Kim et al. [46] proposed a collaborative filtering based recommender systems that use the preference levels of a user for

a product, which are estimated from the navigational and behavioural patterns of users. Kim et al. [47] improved the work in [46] by using association rule mining to generate associations between products and further to derive user's preferences towards products.

Hybrid recommender systems [49] have also emerged combining content-based filtering and collaborative filtering. Some example of some hybrid recommender systems are Fab [50], P-Tango [51] and the one proposed in [52].

1.4.2.1 Pros and cons of collaborative filtering approach

We are going to end the analysis of CF presenting the advantages and drawbacks of the collaborative filtering approaches [22].

The main advantage is:

- Serendipity. This is the ability to make unexpected recommendations [53]. Most systems use a notion of inter-user distance, and thus can define neighbours for a user. If an item of a particular genre is highly preferred by a user's neighbours, then that item could be recommended even if the user has no previous experience with items of that genre. For instance, a user who enjoys heavy metal could be recommended a progressive rock album, despite having never heard progressive rock.

Although the CF approaches have pros they have cons as well, being the main ones listed below:

- Cold start problem. As in content-based filtering it concerns the issue that the system cannot draw any inferences for users about which it has not yet gathered sufficient information. The introduction of new users or new items can cause the cold start problem, as there will be insufficient data on these new entries for the collaborative filtering to work accurately. In order to make appropriate recommendations for a new user, the system must first learn the user's preferences by analysing past information. The collaborative filtering approach requires a substantial number of users to rate a new item before that item can be recommended.
- Gray sheep problem. It is hard to generate recommendations for people who do not belong to the part of an obvious group [54].
- Scalability. When numbers of existing users and items grow tremendously, CF approaches will suffer serious scalability problems, with computational resources going beyond practical or acceptable levels.

Chapter 2

Web mining

2.1 Introduction

In this section, we will discuss the issue of web mining as well as its three categories: web structure mining, web usage mining and web content mining. Moreover, we will present a bibliographical analysis about the combination of the three categories above.

Web mining [55, 56] is the automatic extraction of interesting and potentially useful patterns and implicit information from web documents and services using data mining techniques. Therefore, web mining can be defined as the discovery and analysis of relevant information that involves the use of techniques and approaches based on data mining in order to discover and extract automatically knowledge from documents and web services. This area of research is very wide today because of the tremendous growth of information sources available on the Web as it is mentioned in the introduction and due to the recent interest in e-commerce.

As Fürnkranz mentioned [57], like data mining, web mining is a multi-disciplinary field that draws techniques from areas like information retrieval, statistics, machine learning and NLP among others.

Web mining is roughly based in three categories: web structure mining, web usage mining and web content mining (see Figure 2.1). These three categories are going to be introduced in the next section.

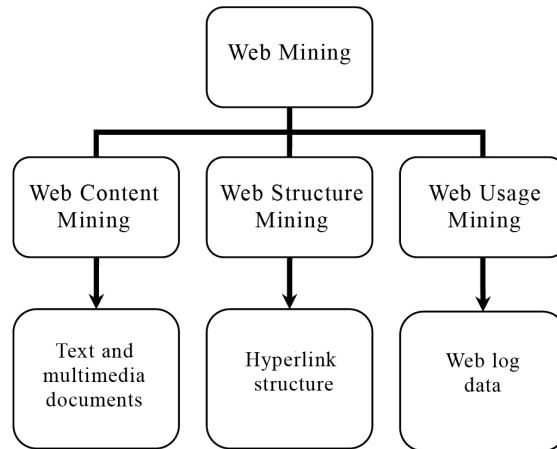


Figure 2.1: Web mining categories and examples of the used data.

2.2 Categories of web mining

Depending on the data source, web mining can be divided into web structure mining, web usage mining and web content mining. Web structure mining is the process of inferring knowledge from the World Wide Web organization and links between references and referents in the web. Web usage mining, also known as web log mining, is the process of extracting interesting patterns from the user-computer interaction using web access logs recorded in web servers. Finally, web content mining is the process of extracting knowledge from the content of documents on the web or their descriptions.

In the following subsections we are going to present each of the web mining categories in detail.

2.2.1 Web structure mining

Web structure mining is defined as the process by which it is discovered the model of link structure of the web pages. Web structure mining examines the link hierarchy of a site that can be used for improving navigation. The structure or the linkage information of the website can be represented in many different forms, for instance using a graph format or using a matrix format such as an adjacency matrix.

Web structure mining provides structural information about web documents and sites [58]. According to the type of web structural information,

web structure mining can be divided into two different kinds namely hyperlinks mining and document structure mining [56]. The first one is used for extracting patterns from hyperlinks in the web, where a hyperlink is a structural component that connects the web page to a different location. It could be to the same web page or to different web page. The second one, the document structure within a web page can be organized in a tree structured format based on the various HTML and XML tags that the page contains [58]. One of the algorithms used is DTD-Miner [59].

The challenge for web structure mining is to deal with the structure of the hyperlinks within the web itself. The different web pages are somehow linked. The appropriate handling of the links could lead to potential correlations among pages, and then improve the predictive accuracy of the learned models [60]. For instance, this inter-relations or correlations can be included in a link prediction system to improve the prediction performance.

Some different algorithms have been proposed to lead with these potential correlations. HITS (Hypertext Induced Topic Search) [61] and PageRank [62], which are explained below, are two popular ones.

HITS algorithm uses the link structure of the web in order to discover and rank relevant pages for a particular topic. HITS algorithm is used for instance, in ASK search engine (www.ASK.com).

In HITS algorithm two different types of pages are identified from the web hyperlink structure: authority pages and hub pages. Starting from a query Q (done by the user for finding web pages similar to the subject of the query) a web page i is called an authority for the query Q if it contains valuable information on the subject. However, there is a second category of pages relevant to the process of finding the authority pages, called hubs. The role of hub pages is to advertise the authority pages. They contain useful links towards the authority pages.

HITS algorithm [61], identifies good authorities and hubs for a topic by assigning two numbers to a page: an authority and a hub weight. These weights are defined recursively. A higher authority weight occurs if the page is pointed to by pages with high hub weights. A higher hub weight occurs if the page points to many pages with high authority weights.

Therefore given a query, HITS will find authorities and hubs. The main idea of HITS algorithm is to identify a small sub-graph of the web and apply link analysis on this sub-graph to find for the given query the authorities and hubs. It requires the web pages in relevance order (returning the pages ordered by high hub and authority weights).

PageRank [62] is an algorithm that calculates the importance of web pages using the link structure of the web. The PageRank algorithm is one

of the most popular algorithms for computing the relevance of web pages. It is used in many different contexts, such as in Google search engine.

The assumption of PageRank was that, the importance of any web page can be judged by looking at the pages that link to it. So, if a web page i includes a hyperlink to the web page j , this means that j is considered to be important and relevant for the topic of the web page i . If there are a lot of pages that link to j , this means that the common belief is that page j is important. Using this assumption it can iteratively assign a rank to each web page, based on the ranks of the pages that point to it.

2.2.1.1 Applications of web structure mining

There are many research works that use web structure mining. For instance, Vaughan and You [63] and Li et al. [64] are two examples of web structure mining application.

Vaughan and You [63] proposed a method that is based on the idea that co-links to a pair of websites are an indicator that the two sites are similar or related. The more co-links the websites of two companies have, the more closely related they are. Since related companies are competing companies, web co-link data can be used to cluster companies into a map of business competition.

C.Li et al. [64] have described the importance of web structure mining and its relationship with the usability. Li and Kit have described a web structure mining algorithm that can effectively extract web structures. They have developed the adaptive window algorithm for discovering the navigational structure in a website, using a set of n web pages of a website. For doing that they use an adjacency matrix using a hyperlink graph. Furthermore, task based usability experiments are conducted on three commercial websites. Results on task based usability correlates well with the users' questionnaires results.

2.2.2 Web usage mining

The second web mining category is called web usage mining. Web usage mining focuses on techniques that could model user behaviour while the user interacts with the web. The extracted knowledge can be used for many different objectives, for efficient reorganization of the website, better personalization as well as recommendations etc. The data can be acquired from different sources such as GPS applications, e-services and user-computer interaction data (e.g. web navigation data).

In this case, we are going to center our attention in web navigation data that is the one that concern in this dissertation.

Web usage mining can be used to discover patterns such as sets of users that access to similar collections of pages, objects or resources.

When the users navigate within the website, they generate and leave behind traces (logs) of their navigation at different places and in different formats. The traces are recorded, in web servers, client machines and in proxy servers. Depending on the data source, the collected log may record some trade generated automatically and not by the users. For this reason, it is impossible to apply machine learning techniques directly to the log records. As a result some pre-processing steps are necessary in order to clean the data. Once the data is cleaned, it is used for discovering patterns.

The web usage mining process can be divided into two phases: firstly, data acquisition and pre-processing phase, and secondly, pattern discovery and analysis phase. In the following subsections we will described each of the phases of the web usage mining process.

2.2.2.1 Data acquisition and pre-processing

This is the step where data is collected and pre-processed. In web usage mining implicit data acquisition is usually used (see in the Section 1.2 for more information about implicit data acquisition). The information or data used for navigation modelling, can be obtained from different sources such as client machines, proxy servers and web log servers. *Client machines* recorded the activities or events that happen within client machine, such as, mouse movements, scrolling on a concrete page and mouse clicks [65]. In *proxies servers*, the network traffic is routed through a dedicated machine. All the requests done by the users and all the answers of the server are serviced using this machine. In web usage mining the most frequent and significant used source is *web log server* data. This data is automatically generated in the web server when it servers the requests of the users that are navigating on the web. In web servers the information about a visitor's activity is recorded [55].

During the collection of data, it is essential to be very careful not to infringe the user's privacy. For dealing with the privacy problem an anonymity process must be applied to the data.

As the most common data source is the one obtained from web servers, we are going to center the attention on that data source. One of the difficulties of the web log server data is that there are considerable number of entries not directly related to the user activity. User clicks indirectly gener-

ate requests automatically in order to complete the requested web page such as images, videos, style (css) and functionalities (scripts). As a consequence, this data needs a cleaning process called data pre-processing phase. This is the phase in which the automatic requests are removed to extract the real user navigation data on which data mining techniques can be applied [66]. There are some steps to prepare data:

- **Data Cleaning.** This is the step where unnecessary and irrelevant records of log data are removed. The requests not related directly with user clicks or request automatically recorded in the server are removed. Finally, the entries occurred from the crawlers or spiders also need to be eliminated because there is not a user behind them.
- **User identification.** It refers to the relation between log records and the users. This process can be done paying attention to the IP address of the requests or using cookies among others.
- **Session identification.** A user session is a set of requests done by the same user in a certain period of time in the same website. In [67] two different methods are presented to have this process done. The former is called proactive, and the sessions are constructed using session ids obtained from cookies. The latter, is called reactive. There are two different ways to apply the reactive sessioning, the first one is to use a time gap between records, in the case that it exceeds certain threshold a new session is created. Some researchers who work with this technique say that the typical value may vary from 10 minutes to 2 hours [66]. The second reactive way is instead of using a single time gap value, use a different time gap for hub type pages and for authority type pages (see Section 2.2.1 for more information about hub and authority). Observing the time that the user spends in hub and authority pages the idea is to calculate a threshold for hub pages and a threshold for authority ones in order to use them as time gaps.

2.2.2.2 Pattern discovery and analysis

After data acquisition and pre-processing phase, the pattern discovery and analysis phase should be carried out. This phase consists of different techniques derived from various fields such as statistics, machine learning, data mining, pattern recognition, etc. applied to the web domain and to the available data.

The aim of this phase is to model the user navigational behaviour. For modelling, usually machine learning techniques are applied. The main techniques are: association rule mining, sequential pattern mining and clustering. The association rule mining is based on the identification of strong rules discovered in databases using different measures of interestingness. Sequential pattern mining is similar to association rule mining with the difference that in this case the order of the element's occurrence is taken into account. Clustering is the division of data into groups depending on the similarity between them. In the next chapter we are going to analyse the mentioned techniques in detail.

2.2.2.3 Applications of web usage mining

Web usage mining could be used for many different purposes, such as for link prediction [68, 69], recommending links to the new users, for web reorganization [70], changing the structure of the website depending on the use of links and for web prefetching [71, 72], among others.

One of the most widely pursued objectives of web usage mining applications has been web access pattern discovery. Although web access pattern discovery has many applications (such as improving web cache performance, personalizing the browsing experience of the users, recommending related pages, etc.) the most widely explored application in the web research community has been web page prefetching.

Many years ago, Kroeger et al. [73] showed that the performance improvement achieved by combining prefetching and caching (i.e. downloading pages that are likely to be visited in the future and storing them in the cache) can be twice that of caching alone. Since then, many approaches have been published that, taking user click sequences as a starting point, concentrate on predicting the next page that will be accessed in order to prefetch it before the user requests it and thus reduce web access latency [74, 75, 71, 76]. Common characteristics of these approaches are generally the use of clustering and/or Markov models to predict the next link to be accessed. In general, the results differ from paper to paper and they are difficult to compare.

Some different prefetching approaches can be found in [71] and [72]. The former combines the user log information and the structure of the website, while the latter proposes a solution that can be used when the navigation logs are not large enough. The order in which the users access the pages is important in differentiating the usage patterns. In fact, this is probably the reason for the popularity of using sequence analysis methods to predict web

access. Another approach would be to take into account the access sequence in a clustering process. The work of [77] proposes an efficient implementation of sequence alignment methods for grouping web access sequences that combine global and local alignment techniques.

2.2.3 Web content mining

The third category of web mining is web content mining. Web content mining aims to extract information related to the website page contents. That is, it extracts or mines useful information or knowledge from web page contents. In the web mining domain, web content mining is essentially an analogue of data mining techniques for relational databases, since it is possible to find similar types of knowledge from the data residing in web documents. The web document usually contains several types of information, such as text, image, audio, video, metadata and hyperlinks. However, the textual content of the web is the most widely investigated area, which is the reason why sometimes web content mining terminology is mixed with text mining terminology. Text mining refers to the process of deriving high-quality information from text. The techniques that are used commonly in web content mining are Natural Language Processing and information retrieval (IR) techniques.

Information retrieval is the study of finding needed information, i.e., IR helps users to find information that matches their information needs. Normally, the users expressed their needs as queries. Technically, IR studies the acquisition, organization, storage, retrieval, and distribution of information. For carrying out that, text mining is commonly used. The objective of text mining is not to understand the total meaning of a document or a text, but to extract patterns across a large number of documents. Text mining uses techniques principally based on machine learning and Natural Language Processing techniques. Machine learning is going to be explained in detail in the next chapter. NLP is the scientific discipline concerned with making natural language accessible to machines. NLP addresses tasks such as identifying sentence boundaries in documents, extracting relationships from documents, and searching and retrieving of documents, among others.

The web content data consist of structured data such as data in the tables, semi-structured data such as HTML documents and unstructured data such as free texts. According to Harmet et al. and Malarvizhi et al. [78, 79] the most used data type is the unstructured data.

According to Kosala et al. [55] the web content mining can be divided in two sections based on two different points of view: information retrieval

point of view and database point of view. The main goal of the content mining from information retrieval view is to improve the filtering and finding of the information to the users, whereas the main goal of database view is to manage the web data.

Kosala et al. summarized the research works done for unstructured data and semi-structured data from the information retrieval point of view. It seems that most of the researches use the bag of words approach, which is based on the statistics about single words in isolation, to represent unstructured text and take single words found in the training corpus as features. For the semi-structured data, many works utilize the HTML structures inside the documents and some utilized the hyperlink structure between the documents for document representation.

The database point of view deals with structure and semi-structure data. It is focused on techniques for organizing content data. In the case of semi-structured data the aim is to organize it into more structured collections. It mainly tries to model the data in the web and to integrate it so that more sophisticated queries other than the keywords based search could be performed.

Even if textual data is the most usual input of web content mining applications, there are works where multimedia data is also used. Multimedia data mining is part of the content mining, which is engaged to mine the high-level information and knowledge from the large online multimedia sources. Multimedia data mining on the web has gained many researchers' attention recently. Working towards a unifying framework for representation, problem solving, and learning from multimedia is really a challenge.

2.2.3.1 Applications of web content mining

The web content mining can be applied to many different fields due to the amount of information that exists from many different topics.

Taddesse et al. [80] present a system to merge different RSS information from different sources. They propose a system that calculates similarities and relationships between RSSs. They identify RSSs which are about the same issue, and merge them in a single one for presenting a single RSS with the information of different sources to the user.

In the Zakaria Suliman Zubi's paper [81] the author presents a system to extract the relationships between Arabic documents using web content mining techniques.

Chaovalit et al. [82] propose an algorithm for detecting sentiments on movie user reviews, based on naive Bayes classifier. They make an analysis

of the opinion mining domain, techniques used in sentiment analysis and its applicability. They implemented the proposed algorithm and they tested its performance, and suggested directions of development.

Schedl et al. [83] present a paper for automatically detecting music band members and instrumentation using web content mining techniques. They combine a named entity detection method with a rule-based linguistic text analysis approach extended by a rule filtering step. They report on the results of different evaluation experiments carried out on two test collections of bands covering a wide range of popularities. They evaluate the proposed system using precision and recall measures.

Apart from the works explained before there are many more such as [84, 85, 86, 87].

2.2.4 Combinations

Although most researchers have concentrated in one of the different categories of web mining i.e. web structure mining, web content mining and web usage mining, some approaches that combine them in order to get more valuable results are also available. The following lines are examples of the mentioned combination works.

Taherizadeh proposed in [88] a system to find useful association rules by integrating web content mining into web usage mining. The textual content of web pages is captured through extraction of frequent word sequences, which are combined with web server log files to discover useful information and association rules about users' behaviours. The main hypothesis is that web page contents can be used to increase in quality of web usage mining results.

The authors of Mobasher et al. [89] claim that both usage and content attributes of a site must be integrated into a web mining framework and used by the recommendation engine in a uniform manner. In the paper they present such a framework, distinguishing between the offline tasks of data preparation and mining, and the online process of customizing web pages. They describe effective techniques based on clustering to obtain a uniform representation for both site usage and site content profiles. Finally, they show how these profiles can be combined with a user session to perform real-time personalization.

Chen et al. describe in their paper [90] a novel representation technique which makes use of the web structure together with web content techniques to better represent knowledge in actual web documents. They named the proposed technique as semantic virtual document. In the paper they discuss

how the proposal can be used together with a suitable clustering algorithm to achieve an automatic content-based categorization of similar web documents. So, the main objective of this research is to investigate how the web structure together with summarization techniques can be used to address the challenging problem of location of relevant web information effectively and efficiently with the help of search engine technologies.

Babu and Sathish [91], have classified web pages in three categories, depending on the number of hit counts (number of clicks). Excellent category to the web pages with highest hit counts, medium to the web pages with moderate hit counts and weak to the pages with smallest hit count. So, users who have visited websites can be classified as class A (excellent), class B (medium) or class C (weak). After that work they make a decision about the structure of the website. The excellent pages will be moved very near to the home page, the medium pages to a mid range distance and the weak pages in the furthest positions. So, they claim that the heap of the tree can be generated based on hit counts available in the log file during a session.

Senkul et al. [92] investigated the effect of semantic information on the patterns generated through web usage mining in the form of frequent sequences. To do this, they developed a framework for integrating semantic information into the web navigation pattern generation process, where frequent navigational patterns are composed of ontology instances rather than web page addresses. They measured the quality of the generated patterns through an evaluation mechanism involving web page recommendation. Although this work combines web usage and content information, it requires having previously built an ontology representing the concepts in the website explored.

Part III

Methods for web mining

Chapter 3

Machine learning for web mining

3.1 Introduction

Machine learning is a branch of artificial intelligence dedicated to developing techniques that allow computers ‘to learn’ automatically. Machine learning is a subfield of computer science that addresses the following question [93]: ‘How can we program systems to automatically learn and to improve with experience?’ learning in this context is to recognise complex patterns and make intelligent decisions or predictions based on data. The difficulty lies in the fact that the set of all possible decisions given all possible inputs is too complex to describe. To tackle this problem the field of machine learning develops learning algorithms that discover knowledge from specific data and experience, based on statistical and computational principles.

So, machine learning is a subfield of computer science that analyses the construction and study of learning algorithms that are able to learn from data [94]. Such algorithms operate by building a model from example inputs and they use the model in order to make predictions or decisions, rather than following strictly static program instructions.

The field of machine learning integrates many distinct approaches such as probability theory, logic, combinatorial optimization, statistics, reinforcement learning, control theory among others. The developed methods are at the basis of many applications, ranging from vision to language processing, forecasting, pattern recognition, games and robotics.

3.2 Learning algorithms in machine learning

The learning algorithms develop to learn from data are very diverse. Although different taxonomies exist, one of the most traditionally used is the distinction between supervised and unsupervised learning.

3.2.1 Supervised learning

Supervised learning is also called classification or inductive learning as it is said in [95]. Supervised learning is the task of inferring a function from labelled training data. Training data includes the input and the desired results. For some examples the correct results (known as dependent variable, class) are known and are given as input to the model during the learning process.

The goal of supervised learning is to build an artificial system that can learn a function for mapping between the input and the output, and can predict the output of the system given new inputs. If the output takes a finite set of discrete values that indicate the class labels of the input, the learned mapping leads to the classification of the input data. If the output takes continuous values, it leads to a regression of the input [96].

There are different supervised learning techniques, such as decision trees [97], Bayesian networks [98], neural networks [99], Support Vector Machines (SVM) [100], k-nearest neighbours (kNN) [101], among others. We are going to present kNN technique due to the fact that is the one we use in the work that will be presented in the further chapters.

kNN classifiers find usually the k objects in the training data that are closest to the test object (they form what is called the neighbourhood of the test object), and base the assignment of a label on the predominant class in this neighbourhood. There are three key elements of this approach: a set of labelled objects, i.e., a set of stored records; a distance metric to compute the distance between objects; and the value of k , the number of nearest neighbours. To classify an unlabelled object, the distance of this object to the labelled objects is computed, its k-nearest neighbours are identified, and the class labels of these nearest neighbours are then used to determine the class label of the object. The kNN algorithm is showed in Algorithm 1.

There are several issues that affect the performance of kNN. One is the choice of k . If k is too small, then the result can be sensitive to noise points. On the other hand, if k is too large, then the neighbourhood may include too many points from other classes. Another issue is the approach to combining the class labels. The simplest method is to take a majority vote, but this

can be a problem if the nearest neighbours vary widely in their distance and the closer neighbours indicate more reliably the class of the object. A more sophisticated approach, which is usually much less sensitive to the choice of k , weights each object's vote by its distance, where the weight factor is often taken to be the reciprocal of the squared distance.

The choice of the distance metric is another important issue. Although various metrics can be used to compute the distance between two points, the most desirable distance metric is one for which a smaller distance between two objects implies a greater likelihood of having the same class. Section 3.3 will present some of the most used distances in machine learning.

Algorithm 1 kNN algorithm

- 1: **for** each object X in the test data **do**
 - 2: Calculate the distance $D(X, Y)$ between X and every object Y in the training data (or a consistent reduced version of it)
 - 3: $neighbourhood \leftarrow$ the k neighbours in the training data closest to X
 - 4: $X.class \leftarrow SelectClass(neighbourhood)$
 - 5: **end for**
-

3.2.2 Unsupervised learning

The second group is called unsupervised learning. In this case the input data is not labelled and does not have a known result, a model is prepared by deducing structures present in the input data. Generally the unsupervised learning methods are divided into association rule learning, sequential pattern mining (that is a further extension to the concept of association rule learning) and clustering.

The work presented in this dissertation makes intensive use of unsupervised learning methods, mainly of clustering algorithms and sequential pattern mining methods which are going to be described in the following sections.

3.2.2.1 Clustering methods

Clustering is considered as one of the most popular unsupervised learning method. It deals with finding a structure in a collection of unlabelled data. The definition of clustering could be the process of organizing objects or entities into groups automatically whose members are similar in some way. Being a cluster a collection of objects which are 'similar' between them and are 'dissimilar' to the objects belonging to other clusters.

Clustering algorithms can be applied in many fields, for example in marketing, finding groups of customers with similar preferences; in biology, classification of plants and animals given their features; in libraries for book ordering; in document classification; and in weblog data, discovering groups of similar access patterns, among others.

There exist many different clustering algorithms. Some of the most popular algorithms are: K-means and K-medoids (it is an adaptation of K-means). These algorithms will be presented in the following lines.

3.2.2.2 K-means

The K-means algorithm [102] is a well known technique for performing clustering of objects. Each cluster is centred about a point called the centroid, where the centroid's coordinates are the mean of the coordinates of the objects in the cluster. The process is described in Algorithm 2.

For the application of K-means, a dataset and a K value (number of clusters) needs to be given as input. The algorithm consists of four different steps. First of all the K centroids are selecting randomly. After that, the objects are grouped into the nearest cluster. Once every object is grouped, the centroids are recalculated and starts again the same process as it can be seen in Algorithm 2. This process is repeated until the centroids do not move, or until a maximum number of iterations.

Algorithm 2 The steps of K-means algorithm

- 1: Randomly select K points into the space represented by the objects that are being clustered. These points represent initial group centroids.
 - 2: Assign each object to the group that has the closest centroid.
 - 3: When all objects have been assigned, recalculate the positions of the K centroids.
 - 4: Repeat Steps 2 and 3 until the centroids no longer move or until a maximum number of iterations has been carried out.
-

3.2.2.3 K-medoids

The K-medoids algorithm is an adaptation of the K-means algorithm. Rather than calculating the mean of the items in each cluster, a representative item, or medoid, is chosen for each cluster at each iteration.

There are two main differences between them, firstly there is no need for repeated calculation of distances at each iteration, since the K-medoids

algorithm can simply look up distances from a distance matrix. Secondly K-medoids can be used with sequential data. The K-medoids algorithm is composed of the steps showed in Algorithm 3.

Algorithm 3 The steps of K-medoid algorithm

- 1: Choose K objects at random to be the initial cluster medoids.
 - 2: Assign each object to the cluster associated with the closest medoid.
 - 3: Recalculate the positions of the K medoids.
 - 4: Repeat Steps 2 and 3 until the medoids become fixed or until a maximum number of iteration.
-

One of the most common implementation of K-medoid clustering is the partitioning around medoids (PAM) explained below.

3.2.2.3.1 Partitioning around medoids (PAM). PAM [103] also clusters objects using K-medoids, where K is specified in advance. The PAM algorithm divides a dataset into a number of K clusters. Both the dataset and the value of K need to be given as input to the algorithm. The algorithm uses a distance matrix, and its aim is to minimize the overall distance between the representative object of the cluster (the medoid) and its members.

The PAM algorithm has two phases, the first one, called build phase, where the medoids are chosen, and the second one, swap phase. In each iteration, it is searched if any of the non-medoids objects of the cluster lowers the average dissimilarity coefficient in comparison with the medoid of the cluster, if it does, it is selected as a new medoid. This process is repeated until the medoids are fixed or until a maximum number of iterations. The algorithm is showed in Algorithm 4.

Different metrics can be used for the calculation of the distance matrix required for PAM algorithm, such as the euclidean, which are the root sum-of-squares of differences, manhattan distance that are the sum of absolute distances and edit distance to deal with sequences, among others (see Section 3.3).

3.2.2.4 Sequential pattern mining

Sequential pattern mining (SPM) has been emerging as an important data mining task since it is broadly applicable to market and customer analysis, web log analysis, intrusion detection systems and mining protein, gene and in DNA sequence patterns [104]. SPM algorithms address the problem of

Algorithm 4 The steps of PAM algorithm

- 1: **Build phase:**
 - 2: Choose K entities to become the medoids, or in case these entities were provided use them as the medoids.
 - 3: Calculate the distance matrix if it was not informed.
 - 4: Repeat Steps 2 and 3 until the medoids become fixed.
 - 5: Assign every entity to its closest medoid.
 - 6: **Swap phase:**
 - 7: For each cluster search if any of the entities of the cluster lowers the average dissimilarity coefficient, if it does, select the entity that lowers the most this coefficient as the medoid for this cluster.
 - 8: If at least the medoid from one cluster has changed go to (3), else end the algorithm.
-

discovering the existent maximal frequent sequences from a given database [105]. The mining process finds frequent sub-sequences from a set of sequential data set [106].

SPM is very similar to association rule learning (ARL) [107], the difference between them is that in the case of sequential pattern mining the items are linked with time. A simple example could be the case of a super market bag, if normally people who buy bread and ham also buy cheese, in ARL it does not matter if the bread is bought first or the ham is first. In the case of the SPM it is important to pay attention to the order.

The algorithms for SPM mainly differ in the way in which candidate sequences are generated and stored, and in the method of testing the candidate sequences for frequency. Based on these conditions SPM is divided into two different branches [108]. The former is based on the apriori algorithm such as GSP [109] and SPADE [110] and the latter in Frequent Pattern growth (FP growth) such as PrefixPan [111]. According to Just [112] the most used algorithms in sequential pattern mining are apriori algorithms. So, we will center the attention in these algorithms. We are going to explain more deeply the SPADE algorithm due to the fact that it is the one which is used in further chapters.

3.2.2.4.1 SPADE. As it is said SPADE is a sequential pattern mining algorithm based on the popular apriori algorithm. The SPADE process is explained in the following lines.

The first step of SPADE is to compute the frequencies of 1-sequences, which are sequences with only one item. This is done in a single database

scan. The second step consists of counting 2-sequences. This is done by transforming the vertical representation into a horizontal representation in memory, and counting the number of sequences for each pair of items using a bi-dimensional matrix. Therefore, this step can also be executed in only one scan.

Subsequent n -sequences can be formed by joining $(n-1)$ -sequences using their lists. The size of the lists is the number of sequences in which an item appears. If this number is greater than a minimum support value (or the minimum frequency value required to a sequence), the sequence is a frequent one. The algorithm stops when no frequent sequences can be found any more. The algorithm can use a breadth-first or a depth-first search method for finding new sequences [110] and has the steps showed in Algorithm 5.

Algorithm 5 The steps of SPADE algorithm

- 1: Make the first pass over the sequence dataset D to yield all the 1-element frequent sequences.
 - 2: Repeat until no new frequent sequences are found.
 - 3: Candidate Generation: Merge pairs of frequent frames sub-sequences found in the $(k-1)^{th}$ pass to generate candidate sequences that contain k items.
 - 4: Candidate Pruning: Prune candidate k -sequences that contain infrequent $(k-1)$ -subsequence.
 - 5: Support Counting: Make a new pass over the sequence dataset D to find the support for these candidate sequences of the frames.
 - 6: Candidate Elimination: Eliminate candidate k -sequences whose actual support is less than $minsupport$.
-

3.3 Metrics used in machine learning

In supervised learning and unsupervised learning methods, the dissimilarities or distances are used to compare objects or items. These distances can be based on a single dimension or multiple dimensions. The most straightforward way of computing distances between objects in a multi-dimensional space is to compute euclidean distances [113] even if many other distances are also used, being some of the most common ones: cosine distance [114], Hellinger distance [115] and KL divergence [116]. In the case of sequential data type where comparisons are mainly based on alignment, the previous distances can not be applied, there are distances like edit distance for this

kind of data. The mentioned distances will be described in the following lines.

- **Euclidean distance:** This is probably the most commonly chosen type of distance. It simply is the geometric distance in the multidimensional space. It is computed as:

$$\text{distance}(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (3.1)$$

Where x and y are two vectors, x_i is a dimension of vector x , and y_i is a dimension of vector y .

- **Cosine distance:** This is a measure of dissimilarity between two vectors of an inner product space that measures the cosine of the angle between them. The cosine of 0° is 1, and it is less than 1 for any other angle. It is thus a judgement of orientation and not magnitude: two vectors with the same orientation have a cosine dissimilarity of 0, two vectors at 90° have a dissimilarity of 1, and two vectors diametrically opposed have a similarity of -1, independent of their magnitude. Cosine distance is particularly used in positive space, where the outcome is neatly bounded in $[0,1]$. The cosine distance is computed as:

$$\text{distance}(x, y) = 1 - \frac{\sum_{i=1}^n x_i \cdot y_i}{\sqrt{\sum_{i=1}^n (x_i)^2} \cdot \sqrt{\sum_{i=1}^n (y_i)^2}} \quad (3.2)$$

- **Hellinger distance:** In probability and statistics, the Hellinger distance (also called Bhattacharyya distance as this was originally introduced by Anil Kumar Bhattacharya) is used to quantify the similarity between two probability distributions. It is a type of f-divergence and it is calculated as:

$$\text{distance}(x, y) = \sqrt{\sum_{i=1}^n (\sqrt{x_i} - \sqrt{y_i})^2} \quad (3.3)$$

- **KL divergence:** In probability theory and information theory, the Kullback–Leibler divergence (KL divergence) is a non-symmetric measure of the difference between two probability distributions. The KL

divergence of the probability distributions on a finite set is defined as shown in Equation 3.4.

$$\text{distance}(x, y) = \sum_{i=1}^n x_i \cdot \log\left(\frac{x_i}{y_i}\right) \quad (3.4)$$

- **Edit distance:** Edit distance [117] is a way of quantifying how dissimilar two sequences (e.g., words) are by counting the minimum number of operations required to transform one sequence into the other. The allowed operations are the removal or insertion of a single character, or the substitution of one character for another. The operations could have the same or different costs depending in the environment that it is used.

Figure 3.1 shows an example of the edit distance between two strings. Being d deletion, s substitution and i insertion. In the example if the cost of each operation is 1 the distance between these strings would be 5. For instance, in the case that the substitution costs 2 the distance would be 8.

```

I N T E * N T I O N
| | | | | | | | |
* E X E C U T I O N
d s s   i s

```

Figure 3.1: Edit distance example

3.4 Performance metrics

Classifiers need to be evaluated and usually some functions, called performance metrics, are used with this aim. But before explaining the measures, it is important to explain what is a Confusion Matrix (also referred to as cross tabulation or crosstab), since much of the measures are based on that table.

Confusion Matrix [94] is a table where rows show the number of instances belonging to each class and columns the number of instances classified as belonging to each class.

In bi-classical problems (problems in which there are only two classes), the minority class or class with fewer examples is also known as positive and

the majority class as negative. In these cases, the matrix will have two rows and two columns, and therefore, four possible values:

- True Positive (TP): Number of positive instances that are positively classified.
- True Negative (TN): Number of negative instances that are negatively classified.
- False Positive (FP): Number of negative instances that are positively classified.
- False Negative (FN): Number of positive instances that are negatively classified.

The matrix is completed as in Table 3.1:

		Prediction	
		P	N
Reality	P	TP	FN
	N	FP	TN

Table 3.1: Confusion Matrix

There are different metrics that use Confusion Matrix to evaluate the goodness of a classifier (the range of possible values is [0-1], being 1 the best value and 0 the worst):

- Hit rate (Accuracy). Percentage of correctly classified instances.

$$\text{Hit rate} = \frac{TP + TN}{TP + FP + FN + TN} \quad (3.5)$$

- Error rate. Percentage of wrongly classified instances.

$$\text{Error rate} = \frac{FP + FN}{TP + FP + FN + TN} \quad (3.6)$$

- Sensitivity (Recall, TPR: True Positive Rate). Percentage of correctly classified positive instances.

$$\text{Recall} = \frac{TP}{TP + FN} \quad (3.7)$$

- Precision. Percentage of instances that are actually positive among those who have been classified as such.

$$\text{Precision} = \frac{TP}{TP + FP} \quad (3.8)$$

- Specificity. Percentage of negative cases classified as such.

$$\text{Specificity} = \frac{TN}{TN + FP} \quad (3.9)$$

- F-measure. It is a measure that tries to balance the successes in both metrics, precision and recall. It is calculated using the harmonic average of the precision and recall.

$$\text{F-measure}_\beta = (1 + \beta^2) \frac{\text{precision} \cdot \text{recall}}{(\beta^2 \text{precision}) + \text{recall}} \quad (3.10)$$

It is common to find other variations of the F-measure, trying to give more importance to one or the other. For instance, F-measure_2 gives more weight to recall, while $\text{F-measure}_{0.5}$, gives it to precision. During this dissertation if β is not specified it means that $\beta = 1$.

- AUC (Area Under ROC Curve). The ROC (Receiver Operating Characteristic) curve is a graphical representation of the relationship between the percentage of true positives (TPR: True Positive Rate) and false positive (FPR: False Positive Rate) as the threshold of discrimination changes (probabilistic value that determines whether a case is considered of one class or another).

To get the performance metric of a classifier the defined area under the ROC curve is calculated, hence the name AUC (Area Under ROC Curve). In the case of the ideal classifier, the area under the curve would be 1, while in the case of a random classifier it would be 0.5. Generally, the classifier with greater AUC is chosen as the best classifier.

3.5 Validation for machine learning models

One of the important issues when designing effective machine learning models is the validation and refinement of the acquired knowledge [118]. So, evaluation is one of the key points in any data mining process. The goal of every classifier is to correctly classify new cases submitted once the learning

is done. The future behaviour of a classifier is estimated using statistical sampling tools and then the possible bias in the estimation (optimistic, pessimistic) and its variability are analysed. It can serve for two purposes: the prediction of how well the final model will work in the future, and as an integral part of many learning methods, to find the model that best represents the training data.

Below some known validation techniques will be discussed. Although they are presented in the estimation of the error, all the performance metrics commented in the previous subsection could be estimated by the following methods.

3.5.1 Apparent error rate

Apparent error rate is the error rate obtained by the classifier after classifying the cases that have been used for training. That is, when the data used for training the model is used for estimating the performance of that model. This generates a non-realistic and overoptimistic or over-fitted prediction considered unacceptable by the data mining community.

If the training sample (a data set used for training the classifier) was infinite (or the whole population) the apparent would be equal to the true error, but it is common to have a sample much smaller than the population.

3.5.2 True error rate

For a correct estimation, the instances of the samples must be selected randomly. Moreover, for the estimation the cases not used in the learning phase must be used. When these assumptions are carried out, it is said that a true error estimation is obtained. There are different techniques to perform this estimation. Here are some of the more commonly used ones:

3.5.2.1 Holdout method

The holdout method consists on splitting into two groups the dataset. The training set, used for training or building the model, and the test set, a subset of unseen examples used for assessing the performance of the model. In addition to that, sometimes another subset of the dataset called validation set is used to assess the performance of the model built in the training phase. The latter set provides a test platform for fine tuning model's parameters and selecting the best-performing model. Normally, the 2/3 of the data set is used as a training set (or training set and validation set depending on the application) and 1/3 as test set. The partition could be done using a

stratified database (maintaining the same percentage of data for each class) or not stratified database.

The holdout method has two basic drawbacks. On the one hand since it is a single train-set experiment, the holdout estimate of error rate will be misleading if we happen to get an ‘unfortunate’ split. On the other hand, in problems with sparse dataset, it may not be able to afford the cost of setting aside a portion of dataset for testing. So, when only a limited amount of data is available, more complex methods are necessary to achieve an unbiased estimate of the model performance over unseen examples. The limitations of the holdout can be overcome with a family of resampling methods such as random subsampling, K-fold cross-validation, leave-one-out cross-validation and bootstrap.

3.5.2.2 Random Subsampling

In this method in order to improve the estimation, the holdout estimation could be repeated several times, using different training-test partitions randomly generated. The final estimation will be the average of the values obtained from each run.

3.5.2.3 K-fold cross-validation.

The original dataset is divided into K disjoint subsets (fold-s) approximately of the same size and used to carry out K experiments.

For each of the K experiments, the method can use K-1 folds for training and the remaining one for testing or some for training, some for validation and some for testing (as it has been explained in holdout technique). The final estimate will be the average of all estimates obtained. The estimated need to be weighted if the subsets are not of the same size.

K-fold cross-validation is similar to random subsampling. However, the advantage of K-fold cross-validation is that all the examples in the dataset are eventually used for both training and testing. On the other hand the disadvantage is that this method presents a high computational cost, because the process is repeated K times.

3.5.2.4 Leave-one-out cross-validation

This is a specific case of K-fold cross-validation, where K is chosen as the total number of examples. So, for a dataset with N examples, perform N experiments using N-1 examples for training and the remaining example for testing. The computational cost of this method is even higher than the one

of in K-fold cross-validation, which is the reason why it is used only with small datasets.

The estimation done by leave-one-out is obtained by calculating the average obtained for each of the N cases and is an unbiased estimation of the true value.

3.5.2.5 Bootstrapping

Despite the fact that in the long term the estimation would be unbiased, the estimation done by the leave-one-out method has large variances in small samples.

For dealing with this problem the bootstrapping method is usually used. In this method from a set of N examples, N examples are chosen for training randomly and with replacement, i.e., instances can be repeated, therefore, there will be examples not chosen. It is common practice to use the latter for testing phase. This process is repeated multiple times (around 200 for proper estimation) and the final estimation will be the average of the values obtained. The computational cost is higher than for leave-one-out.

Chapter 4

Content processing methods

4.1 Introduction

The area of document comparison is a very important one in recent years. As it has been presenting until this section the accessible digital information is intractable manually. So, it has become very important for humans that computers are able to compare documents in order to help the users finding the information they need.

Document comparison is applicable in many research areas, for instance, in the fields analysed in Chapter 2, such as text mining, information retrieval and Natural Language Processing.

However, it is not an easy task for computers to relate documents due to the fact that they do not have the ability to understand natural languages. For comparing content of documents, the computers should have the capacity to identify the documents' context and using it to relate different texts. Unfortunately this is not possible yet.

Even if computers can not understand the natural language, many methods are being studied and used in the last decades for document comparison and document relation (document comparison techniques). In this chapter some of these methods will be presented. On the one hand, tools based on statistical techniques: search engines, keyword extractors and topic modelling, and on the other hand, tools based on ontologies.

Before applying document comparison techniques, documents must be prepared. Furthermore, although the documents' content could be given directly by the providers, there are some cases such as the web content environment where sometimes the texts must be collected. So, they require a data acquisition and pre-processing phase before the document comparison

techniques are applied.

4.2 Data acquisition

The document comparison techniques could be applied in many different fields. It could be used for research article relation, for book relation in a library and for URL comparisons, among others. Depending on the data source the acquisition phase will change.

Even if the data is normally provided directly, sometimes the data must be extracted automatically using tools such as web content extractors. We are going to analyse in detail what a web content extractor is and what it is for in the following subsection.

4.2.1 Web content extractors

Web content extractor also known as web crawlers or web spider are concerned with extracting the relevant text from web pages. Within the past few years there has been an increase of web content extractor datasets. In the literature there are different tools for extractions as for example: Mozenda [119], Web Info Extractor [120], Web Text Extractor [121], Web Data Extractor [122], Web Content Extractor [123]. These tools usually help to download the essential information of a website. However, many of them are not capable to extract a big number of web pages. For large number of web pages, the Wget [124] crawler is commonly used [58].

A web content extractor [125] is a relatively simple automated program, or script that methodically scans or ‘crawls’ through Internet pages to create an index of the data that is looking for or simply to download the information or pages of the site. Normally these kinds of programs are used only once because usually the crawlers download all the site pages in each run. But they can be programmed for long-term usage as well for detecting the changes of the site.

There are different uses for web content extractors but in general and essentially a web content extractor may be used by anyone seeking to collect information out on the Internet. For instance, tools like search engines frequently use web content extractors to collect information. The main purpose of the search engines is to collect information so that when an Internet surfer enters a search term, they are able to provide as soon as possible the relevant information or the relevant site (it is going to be presented later in this chapter). Linguists may use web content extractors to perform a textual analysis; that is, they may comb the Internet to determine which

words are commonly used today. Market researchers may use a web content extractor to determine and assess trends in a given market.

In the case of URL comparison purposes, the output of the crawlers are normally some collections of HTMLs of a site in the case of internal web content extractors (crawlers that download the links that are inside the site) and a list of HTMLs that go outside the site in the case of the external crawlers.

A study of web content extractor tools has been done and it is presented in the subsection below.

4.2.1.1 Study of the web content extractors

The purpose of this study is to analyse the different tools listed in the previous section: Mozenda, Web Info Extractor, Web Text Extractor, Web Data Extractor, Web Content Extractor and Wget.

The analysis has been done focusing on several points:

1. An overview of the product itself.
2. The operating system it accepts.
3. The kind of input it requires.
4. The kind of output it provides.
5. Format options for output files.
6. Some comments.
7. The URL of the product.

In this section the summary of the study is going to be presented (for more detail about the analysis done for each tool see Appendix A).

Summary of analysed software

The aim of the revision is to find an adequate tool to extract textual content from a website. Table 4.1 shows a summary of the tools analysed. In the table, *W* means Windows, *L* means Linux, *Y* means yes and *N* means no.

	Operating system	Is the input a URL?	Does it perform recursive analysis of the website?	Does it provide an URL list?	Does it provide an structured URL list?	Does it provide the text related to each of the explored URLs?	Does it provide concrete information such as e-mails, telephones...?	Format options for output files	Does it exist Demo
Mozenda	W	Y	N	N	N	N	Y	CSV, TSV, XML, EXCEL	Y
Web Info Extractor	W	Y	N	N	N	N	N	CSV, in a text file or in a dataset (access,MySQL, SQLServer)	Y
Web Text Extractor	W	N	N	N	N	N	N	Text file	Y
Web Data Extractor	W	Y	?	?	N	Y	Y	CSV, txt, html, EXCEL	Y
Web Content Extractor	W	Y	Y	N	Y	N	N	EXCEL(CSV), Access, txt, html, XML, SQL script, MySQL script and to any ODBC data source	Y
Wget	W & L	Y	Y	Y	N	Y	N	html, txt, pdf	Free

Table 4.1: A summary of web content extractor analysis

Depending on the use given to the extracted information, it would be convenient to use one tool or another. For instance, centring in using the information for designing a link prediction system (as it is our case), it could be considered a good tool the one that has the option to get a local copy of the site due to the fact that the site structure and content could change rapidly and as the logs are normally taken in a certain moment, it is important to have the image of the site at that moment.

Moreover, for the selection of one specific tool we considered essential not being tied to one specific operating system.

In addition to that, even if it could seem more comfortable to have a tool that extracts the real content of each URL of the site, we considered that this can be done in a post-process later specifically for each site. Consequently, we considered that it is more convenient to have the HTML files in order to decide the post-process needed in each site.

Therefore, the best option would be Wget: it is the tool more used in the literature for web content extraction and it is an open source tool. Moreover, giving as input the main URL of a site it downloads the whole copy of the site. In addition, it meets the requirements previously commented, it gives the chance to download a local copy of the site, it can be applied in different operating systems and it provides HTML files of the site.

4.3 Content pre-processing phase

In order to apply any kind of document comparison techniques, the documents have to be pre-processed, removing, for instance, the part of the texts that are not relevant.

The steps that the web content pre-processing phase should have are:

1. **HTML parser.** An HTML parser is used for extracting all real text content. These tools give the option to filter parts of the HTMLs that are not relevant, extracting only the significant text.
2. **Stop words removal.** Many of the most frequently used words in any language are useless in information retrieval (IR) and text mining; these words are called stop words. For instance, words like: the, of, and, to; are considered stop words, generally words that do not provide with extra information. This step is a very important one because stop words accounts 20-30% of the total word counts and erasing them will increase the efficiency and effectiveness of the system.
3. **Stemming.** This is an NLP technique and it consists of a normalization based on morphology. Stemming is the term used in linguistic morphology and IR to describe the process for reducing inflected (or sometimes derived) words to their word stem, base or root form. The stem needs not to be identical to the morphological root of the word, it is usually sufficient that related words map to the same stem, even if this stem is not in itself a valid root. This step improves effectiveness of IR and text mining. Merging words with same roots may reduce indexing size as much as 40-50%. For languages with relatively simple morphology, the influence of stemming is less than for those with a more complex morphology. Most of the stemming experiments done so far are for English and other west European languages [126]. Anjali et al. have done a study about different stemming algorithms, and it is said that the most used stemming algorithm in literature is Porters stemmer [127].

4.4 Document comparison techniques

Once the pre-processing phase is done, a possible objective is to calculate the similarity between documents. The technologies that are commonly used to compare documents are NLP and IR, as it is mentioned in the introduction.

The goal when applying techniques of NLP is to identify the most relevant semantic information from texts, which can be used for finding similarities between different documents.

In document comparison technologies, documents are usually represented as vectors. In [128] is mentioned that the most common method for text representation is to represent documents algebraically as vectors in multidimensional space using Vector Space Model (VSM) [129].

There exist many different technologies that are used for document comparison. Different types of tools can be found as for example the ones based on statistical methods and the ones based on ontologies.

4.4.1 Tools based on statistical methods

These kinds of tools take into account the exact word matching, i.e., they calculate the similarity or relation between documents taking into consideration the common words between documents. So, they do not keep in mind that different words could have similar or the same meaning. Some instances of statistical tools are: search engines, keyword or key phrases extractors and topic modelling, among others.

In statistical methods term weighting algorithms play a very important role. The most widely used term weighting algorithm is the one called TF-IDF. On the one hand, TF (Term Frequency) is defined as the number of times a term occurs in a document. As it is obvious it is calculated for each document of the collection. On the other hand, IDF (the Inverse Document Frequency) consists of counting the number of documents in the collection being analysed when the term appears. The formula would be the following:

$$tf - idf_{ki} = (tf_{ik}/tfmax_k) * \log(N/n_i) \quad (4.1)$$

Where tf_{ik} is the frequency of the term i in document k ; $tfmax$ is the maximum term frequency in document k ; N is the number of documents and n_i is the number of documents containing term i .

Some of the tools listed above use this term weighting method for extracting the relation between texts. In the next subsections the listed tools are going to be explained.

4.4.1.1 Search engines

A web search engine is designed to search for information on the World Wide Web. Search engines are tools that search documents similar to the specified keywords and return an ordered list of similar documents. The

ordered list is often called *hits*. The information may consist of web pages, images, information and other types of files. Some search engines also mine data available in databases or open directories.

There exists the possibility to use a commercial search engine or an open source one. For most websites, the use of a commercial search engine is not a feasible alternative because of the fees that are required and because they focus on large scale sites. On the other hand, open source search engines may give the same functionalities (some are capable of managing large amount of data) as a commercial one, with the benefits of the open source philosophy: no cost, software maintained actively, possibility to customize the code in order to satisfy personal needs, etc. There is a research work that compares some open source search engines [130].

Normally the search engines use metrics to calculate the similarity of texts, such as TF-IDF and BM25 [131] among others. Generally, the most used in the literature is the TF-IDF metric (see Section 4.4.1). Even if search engines are prepared for finding query-document relationships, as they use distances as the ones mentioned previously, it is possible to use them as document-document similarity calculation tools as well. This can be done using as query the complete textual information of a document and comparing it with the rest of the documents in the collection. Using the search engine in this way, a list of the most similar documents for the query can be obtained. To use search engines for document comparison, first of all an index with the whole document collection needs to be created.

There are 11 features that it should be specify for the creation of the indexer: *Storage*, indicates the way the indexer stores the index, either using a database engine or a simple file structure (e.g. an inverted index); *Incremental*, indicates if the indexer is capable of adding files to an existent index without the need of regenerating the whole index; *Results Excerpt*, if the engine gives an excerpt (“snippet”) with the results; *Results Template*, some engines give the possibility to use a template for parsing the results of a query; *Stop Words*, indicates if the indexer can use a list of words used as stop words in order to discard too frequent terms; *File Type*, the types of files the indexer is capable of parsing (the common file type of the engines analysed was HTML); *Stemming*, if the indexer is capable of doing stemming operations over the words; *Sort*, ability to sort the results by several criteria; *Search Type*, the type of searches it is capable of doing, and whether it accepts query operators; *Item Indexer Language*, the programming language used to implement the indexer; *License*, determines the conditions for using and modifying the indexer and/or search engine. Each of the search engines can be characterized by the features it implements as well as the

performance it has in different scenarios.

The research work presented by Christian et al. [130] compares 17 search engines (from the 29 search engines found). Once they execute the tests, only 4 search engines were selected as the best; Zetair, XMLSearch, MG4J and Indri. According to the results of this paper and according to the experts, the most convenient search engines for the application context of this work are MG4J and Indri.

An analysis of these two search engines has been carried out (see Appendix B for the details of the analysis) whose main goal was to select the most convenient search engine for the similarity measurement, taking into account that the results must be very intuitive and easy to normalize. Taking into account the condition mentioned before, the best software is MG4J. Moreover, MG4J gives the chance to use different metrics for calculation the similarity between texts as BM25 and TF-IDF. In the case of Indri, the results are not intuitive and hard to be normalized. That is the reason why we reject it.

4.4.1.2 Keyword or key phase extractors

The second option, for calculating the similarities between documents, is the extraction of keywords. Keywords and key phrases (multi-word units) are widely used in large document collections. They describe the content of single documents and provide a kind of semantic metadata that is useful for a wide variety of purposes. The task of assigning key phrases to a document is called key phrase indexing. For example, research papers are often accompanied by a set of key phrases or key words freely chosen by the author. On the Internet, digital libraries, or any repositories of data (flickr, del.icio.us, blog articles etc.) also use key phrases to organize and provide a thematic access to their data. This kind of techniques are usually combined with distances like cosine distance (explained in Section 3.3) in order to extract relationship between documents.

Once the pre-process of the document collection is carried out a keyword or key phrase extractor tool can be used for being able to relate some documents in a semantic way.

We evaluated the available keyword extractor tools for calculating the similarity between documents from the semantic point of view. We analysed the following tools: Kea (Key phrase extraction algorithm) [132], Keyword Analysis Tool [133], Keyword/Terminology Extractor [134], Maui [135], Topia Term Extractor [136] and Yahoo Term Extractor [137] focusing on some characteristics:

1. An overall of the product.
2. The type of input it requires.
3. If manual work is required.
4. If it is prepared for a document collection.
5. The number of languages it accepts.

The main features of each one are presented in Appendix C. In Table 4.2 a summary of the analysis is showed.

	Can be the input a URL?	Is a manual work required?	Is a stop word list included?	Is there any option to do it automatic?	Does it accept different languages?	Format option for input
Kea	Y	Y	N	N	N	Text
Keyword Analysis Tool	Y	Y	Y	N	N	URL
Keyword-Terminology Extractor	Y	Y	N	N	Y	HTML, Text
Maui	N	Y	N	N	N	PDF, Text, Word
Topia Term Extractor	?	Y	N	N	N	Text
Yahoo Term Extractor	N	N	Y	N	N	Text

Table 4.2: A summary of the keyword extractor analysis

After the analysis of the keyword extractor tools, we have realized that most of the applications do not allow the treatment of document collections, and some of them only return the most frequent words. We consider that the tool for being selected must be automatic or in the contrary must provide the option for making it automatic. It should be intuitive as well. Even if some tools give the chance to include a stop word list, we consider that this is a step that can be done in the pre-process phase, so it is not an important characteristic. The one that covers the conditions commented is the Yahoo Term Extractor tool and this is the tool that has been selected.

The term extraction process needs to be combined with some distances for calculating the distance between texts or documents. Although distance metrics as the one described in Section 3.3 can be used, in the case of keyword dissimilarity the most used one is the cosine distance [114].

color they assigned them. That list of words would be a topic, and each color represents a different topic (see Figure 4.1).

The example used in [138] is an article entitled "Seeking Life's Bare (Genetic) Necessities". This article is about using data analysis to determine the number of genes that an organism needs to survive. The topics that are extracted from the example are: words about data analysis like "computer" and "prediction" highlighted in blue, words about evolutionary biology, such as "life" and "organism", highlighted in pink; words about genetics, such as "sequenced" and "genes," are highlighted in yellow. So that, the main sense of the document is extracted.

The LDA topic modelling algorithms take as input a set of documents and a number of topics for finding and providing as output a list of topics represented by the words belonging to them sorted by importance and a matrix with the probability that each document has to belong to each of the topics (document-topic matrix). The LDA algorithm is showed in Algorithm 6.

Algorithm 6 LDA algorithm

```
1: Initialization of topic assignment randomly.
2: for each iteration do
3:   for each document do
4:     for each word do
5:       Resample topic for the word, given all the other words and
       their current topic assignment.
6:     end for
7:   end for
8: end for
```

The main core of the algorithm is the 5th step. For resampling the topic of the word two different phases, are required. The first one is concerned to the topics occur in the document and the second one is concerned to how many times the word is related with each topic. Although different probabilities needs to be calculated in order to determine the topic for a word a graphical explanation is going to be used in this case for a better understanding of the algorithm. Imagine the following example, in Table 4.3 a document containing 5 terms are presenting and the corresponding topic for each word.

Topic	3	?	1	3	1
Doc.Words	Etruscan	Trade	Price	Temple	Market

Table 4.3: LDA example: document-topic relation

In this example the topic of the word *trade* is looking for. Furthermore, the Table 4.4 shows the number of times that the word *trade* is related with each of the topics.

	Topic1	Topic2	Topic3
Trade	10	7	1

Table 4.4: LDA example: word-topic relation

Figure 4.2 shows how a topic is determined for the current word *trade*. The blue rectangles (the ones with spots) represent the number of words in the document related with each topic, in our example in the document there are 2 words related with *Topic 1* and 2 words related to *Topic 3*. There is not any words related with *Topic 2* so the minimum value is given. The green rectangles (the dashed ones) represent the number of times that words are classified as being of each topic (in this case using the information of Table 4.4). So for selecting the topic for the word *trade* the area of this two rectangles is calculated (in this case the red rectangle, the one in the middle). According to the figure the word *trade* would be classify as *Topic 1*. This step is repeated for each word of the document, for every documents of the dataset and for the iterations decided by the users.

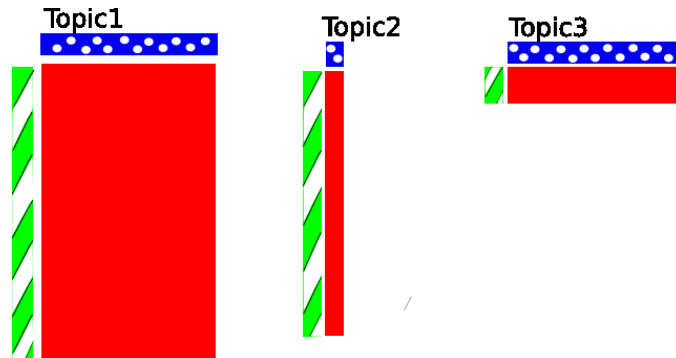


Figure 4.2: LDA Example: Selection of topic

The output of the LDA process such as document-topic probability vector can be used to deduce the bias of the content of each document. This information combined with a distance metric (see the Chapter 3) could be used to compare documents with each other and also to group them by thematic structure.

4.4.2 Tools based on ontologies

These kind of tools take into account that different words could have similar or the same meaning, instead of considering only the exact word matching they pay attention to the semantic meaning of the words as well. Many semantic tools use ontologies to extract and to improve the document relationship measures.

The techniques presented until this moment such as search engines, keyword extraction and topic modelling are statistical techniques for document comparison and they are based on the exact word matching. That is, if two documents have many terms in common they would be highly related using the techniques listed above. Therefore, it is known that different words could have the same or similar meaning that is the case of the synonym words, and these techniques do not have it into consideration. Having the idea that different terms could have similar meaning, in the document comparison process the extracted relation would be much richer and better, what means that the document comparison would be better as well and will propose links more interesting for the user interest.

Although many definitions of ontologies can be found, one of the most used one is the following "an ontology is the specification of conceptualizations, used to help programs and humans to share knowledge" according to Tom Gruber, an artificial intelligence specialist [142]. These ontologies capture semantic relationships between concepts or vocabulary used in a particular domain and can potentially be used to discover inherent relationships between descriptions of entities. That is, ontologies could be used in order to relate documents, taking into account relationships between words instead of finding an exact word matching.

Ontologies have been of great interest for the semantic similarity research community as they offer a structured and unambiguous representation of knowledge in the form of conceptualizations interconnected by means of semantic pointers. These structures can be exploited in order to assess the degree of semantic proximity between terms. According to the principle in which the similarity/relatedness computation is based and the way in which the ontology is exploited and/or complemented with other sources

(e.g., thesaurus, domain corpora, etc.), different families of methods can be identified. In Sánchez et al. [143] they survey and compare most of the ontologies-based similarity/relatedness measures developed in recent years.

Thiagarajan et al. [144] extend the notion of semantic similarity to consider inherent relationships between concepts using ontologies. They propose simple metrics for computing semantic similarity using spreading activation networks with multiple mechanisms for activation (set based spreading and graph based spreading) and concept matching (using bipartite graphs). They evaluate these metrics in the context of matching two user profiles to determine overlapping interests between users.

Thanks to initiatives such as the semantic web, which brought the creation of thousands of domain ontologies [145], ontologies have been extensively exploited in knowledge-based systems [146] and, more precisely, to compute semantic likeness.

The most used ontology for research works is Wordnet ontology [147].

It groups English words into sets of synonyms called synsets, provides short definitions and usage examples and records a number of relations among the synonym sets or their members. Synsets are interlinked by means of conceptual-semantic and lexical relations. The resulting network of meaningfully related words and concepts can be navigated with the browser. Wordnet is also freely and publicly available for download. Its structure makes it a useful tool for computational linguistics and NLP.

Wordnet superficially resembles a thesaurus, in that it groups words together based on their meanings. However, there are some important distinctions. First, Wordnet interlinks not just word forms, strings of letters, but specific senses of words. As a result, words that are found in close proximity to one another in the network are semantically disambiguated.

Ontologies like Wordnet can be used for text comparison. There are many different methods in which ontologies can be used for text similarity. One of them for instance, is to create a vector for each pair of texts where the relation of each term of each text is related with the terms of the rest of the texts. The vector's length could be the size of the vocabulary of the both documents. If the term appears in both texts the weight of that term in the vector would be 1. In the case that the term does not appear in both, the similarity between the term and the rest of terms in the other text are calculated using the ontology. If the term does not have any relation with the terms of the other text according to the ontology would have the value 0 in the vector.

Normally once the similarity vectors are calculated for the whole document collection a distance metric is used (see Chapter 3) for extracting the relationship between documents.

Part IV

Proposed systems

Chapter 5

Application of web mining techniques to a tourism environment

In this chapter a brief introduction of the relation between the tourism area and technology is presented as well as a work done in the tourism area for improving and filling the gap that exists in the tourism web personalization environment.

5.1 Introduction

As technology is evolving faster than ever before, it has made most travellers around the world much more technology-savvy than in the past. The Internet has revolutionized the tourism industry more than any other factor in the last few decades. Also, as more people are connected to each other, with access to the vast pool of information available online, an increasing number of travellers are seeking information via the Internet prior to making any travel decisions.

For tourists, the decision on which destination or product to choose requires a considerable time and effort [148] because, as Ballantyne et al. [149] stated, tourism services are a class of product regarded as high risk and consumers are often led to engage in extensive information search. As the online user absorbs information from a variety of sources, it is usually the site or information source that can best stimulate the viewer to travel that will be remembered by the user [9].

Destination Marketing Organizations (DMO) are the tourism organizations responsible for management and promotion of a destination. Previous works, (for example [7]) identified the importance of DMOs understanding new challenges and the meaningful use of new technologies to seek excellence in destination marketing. Marchiori et al. [150] affirmed that in order to maintain their share of the market tourism organizations have to respond not only by adopting new technologies, but also by interpreting and using the knowledge created by Internet users.

The tourism industry has experienced a shift from offline to online travellers. Experts underlined many years ago that the Internet is the main source of information in the tourist domain [7]. An increasing number of travellers are no longer dependent on travel agencies to look for information for their next trip; they have replaced using agencies by the use of the Internet [6]. Steinbauer et al. [8] affirmed that the development of information communication technologies during the last decade has affected the tourism industry, as a growing number of travellers have begun to look for tourism information online. As the experts pointed out in the ENTER 2013 eTourism conference held in Innsbruck in January 2013, ‘these systems have significantly changed the travel industry’. As a consequence, DMOs must use their official websites to interact with tourists in order to promote a destination and provide information on it and, furthermore, they should extract knowledge from this interaction. As e-Destinations serve as platforms where consumers can be inspired, get all the information they need about the desired destination and eventually book the holiday, the presence of destinations in the Web is crucial [151]. Moreover, as Hsu et al. [152] concluded in their work, the web facilities provided to tourists affect their loyalty.

The success of electronic commerce, especially for the less well-known companies, is largely dependent on the appropriate design of their website [153]. In the paper of Chaffey et al. [154] it is stated that a good website should begin with the users and understanding how they use the channel to shop. This confirms that understanding the needs and preferences of the website audience will help to answer questions about what the content of the website should be, how it should be organized and so on.

In this context, web personalization becomes essential in industries such as tourism and it can be positive for both the user and the business.

Within this context, the use of intelligent systems in the tourism sector has become crucial [155]. These information systems can provide tourism consumers and service providers with the most relevant information, more decision support, greater mobility and the most enjoyable travel experiences.

As stated in the previous paragraphs, the Internet has become one of the most widely accepted technologies and there is currently a wide range of systems related to it such as recommender systems, context-aware systems, web mining tools, etc. Moreover, travel agents are among those service providers for whom adoption of the Internet could be the best marketing device for their business and a tool to give them a competitive advantage [156].

In any web environment, the contribution of the knowledge extracted from the information acquired when the users navigate in a website is twofold: it can be used for web personalization (i.e. for the adaptation of the website according to the user requirements) and also to extract knowledge about the interests of the people browsing the website, which will then be useful for the service provider; in the case of tourism websites, the staff of the DMOs.

The use of web mining techniques could help in the web tourism personalization, making the navigation for travellers more satisfactory and providing useful information to service provider in order to make more interesting and suitable offers to the travellers, facilitating the understanding between each other.

5.2 Related work

Navigation profiles are part of many electronic tourism applications and particularly recommender systems. The information needed to build a navigation profile can be obtained explicitly or by observing the actions of the user [17]. In the area of tourism, navigation profiles are mainly generated by asking the users to fill in a questionnaire or by an interface [157]. The users must usually complete some steps in order to create the profile; for instance, by selecting photos that they like [158, 159]. Thus, the most common user profiling strategy in tourism is to use information provided by the user. The profiles created are generally used to recommend some tourist plan or information based on a collaborative filtering approach.

As it has mentioned in Chapter 1 explicit acquisition of the data has the problem that normally people are not willing to provide information by filling long forms. The alternative then is to use implicit data collection. It can be done for instance, using the web server log data or using GPS or smart phone for collecting location data. Web usage mining can be used to extract knowledge from observed actions using different techniques as has been explained in Chapter 2. Similar techniques have also been applied to the area

of tourism but, to our knowledge, they have always been applied to extract knowledge from explicitly acquired user information. For example Hsu et al. [152] presented a system that uses an integrated Bayesian network mechanism, using a linear structural relation model (LISREL) to predict tourism loyalty based on 425 valid answers to a poll (explicit information requirement) collected from tourists about their holiday experience at the Toyugi hot spring resort in Taiwan. In another work, Brejla and Gilbert [160] used a data-driven approach to knowledge discovery. Their aim was to achieve a deeper understanding of guest-to-guest and guest-to-staff interactions on board cruise ships. They used holiday reviews retrieved through web content mining from CruiseCritic.com. Although the previous works describe the application of web mining techniques in the tourism context they are not based on user navigation logs but on user reviews or information explicitly acquired from users.

5.3 Motivation and overview of the work

As it has been previously mentioned, one of the objectives of this work is to discover navigation profiles and use these to personalize the browsing experience of the user, thus making it easier and more convenient. We consider that this can be done by adapting the navigation scheme by providing the users with a list of links of interest to them in the early stages of the navigation, so that they can achieve their objective faster. This would probably help them to have a shorter and more satisfactory navigation experience by skipping some of the intermediate pages on the way to their objective. In contrast, prefetching, one of the most widely pursued objectives in the web usage mining, would take the users through the same path they would navigate without adaptation but faster.

Furthermore, we aim to combine the knowledge extracted from web usage information with knowledge extracted from the web content through web content mining techniques to provide the staff of the Bidasoa Txingudi bay DMO with information about the types of users browsing their website, according to interests. This information will be useful to provide a better service or for future marketing campaigns.

The characteristics of the system described in this research work are not included in the works described until this section: a general, automatic and non-invasive system that combines usage and content information from a tourism website to make the optimum use of it. The works that had the aim of predicting or recommending links had no further aim and they did

this based only on usage information. In the cases where usage information was combined with content information, the content information was not extracted automatically; some prior knowledge of the content structure of the website was required. Finally, focusing in the context of tourism, the web mining applications we found in bibliography were invasive; they required information obtained explicitly from the customers. In contrast, our work proposes a system that, based on usage and content information and without any prior knowledge of the content structure of the website, is able to predict links and to provide information about the types of users that navigate in a website according to their interests.

The work that is going to be introduced in this chapter, presents the design of a system, built using the minimum information stored in a web server (the content of the website and the information from the log files stored in Common Log Format (CLF) [161] and its application to the *bidasoatourismo* website (BTw). The proposed work combines web usage and content mining techniques with the three following main objectives: generating user navigation profiles to be used for link prediction; enriching the profiles with semantic information to diversify them, which provides the DMO with a tool to introduce links that will match the user's taste; and moreover, obtaining global, month-dependent and language-dependent user interest profiles, which provides the DMO staff with important information for future web designs, and allows them to design future marketing campaigns for specific targets.

This process requires a data acquisition and pre-processing phase. Then, the machine learning techniques are mainly applied in the pattern discovery and analysis phase to find groups of web users with common characteristics related to the Internet and the corresponding patterns or user profiles. Finally, the patterns detected in the previous phases are used in the exploitation phase to adapt the system and make navigation more efficient for new users or to extract important information for the service providers.

The system was developed in several stages. First we analysed the navigation of users (web usage mining) and built user navigation profiles that provide a tool to adapt the web to new users while they are navigating (through link prediction). We then automatically extracted thematic information from the content of the URLs (web content mining) and combined this with usage information to obtain information about the interests of the users browsing the website (i.e. we extracted user interest profiles).

Figure 5.1 shows the different stages of the user modelling platform for working with data from *bidasoatourismo*. As it can be seen in Figure 5.1, the modelling based on the web navigation of tourists has been carried out

in three different ways: navigation profiles based on usage information (in green in the right side), semantic profiles combining the usage information with the content information (in yellow in the left side), and the enriched navigation profiles using content information (in orange in the middle of the graphic).

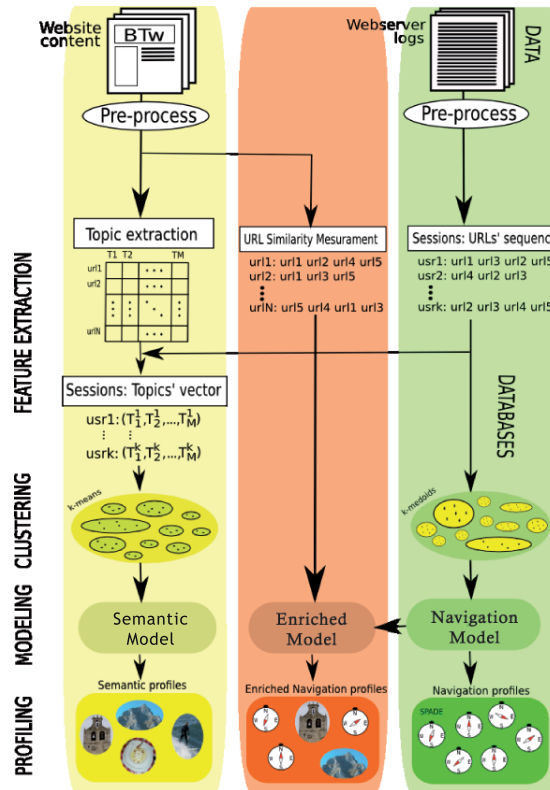


Figure 5.1: Schema of the system architecture.

For making the most of BTw the work has been developed in three different steps. In the first one, a preliminary system was designed to analyse the suitability of our proposals, the second one is a global system (which is an extension of the previous system) where each of the steps presented in the Figure 5.1 are going to be implemented: navigation profiling, semantic or interest profiling and enriched navigation profiling. The third one is a preliminary work of a new system which uses different technologies for content processing.

5.4 *bidasoa turismo* website

The research environment was the Bidasoa Txingudi bay DMO, which is located at the western tip of the Pyrenees, straddling two countries (France and Spain) and linking the Basque provinces of Gipuzkoa and Lapurdi. The Bidasoa river has had the effect of socially and culturally linking the three towns surrounding the bay (Hendaye, Hondarribia and Irun). The area offers the opportunity of a wide range of tourism activities and the *bidasoa turismo* website (BTw) (www.bidasoaturismo.com) includes all sorts of practical tourist information on the area: thematic tourism, professional tourism, gourmet tourism, agenda, recommendations, etc. A screen shot of the website is shown in Figure 5.2.

The usage data for BTw was provided by the staff of the DMO. The information contained in this database consists on the web server logs of requests stored in Common Log Format [161]. Apart from the usage data, we also acquired and used the content information of the website; i.e. the text appearing in the website.

In addition, we used a third source of information: the language used when accessing the website. The web page can be accessed in four different languages (Basque, Spanish, French and English), which provides information about the origin of the users.

5.5 Preliminary system

In this section we present the system based on a collaborative filtering approach that adapts the BTw website to improve the browsing experience of the users: it generates automatically interesting links for new users. In this work firstly based on previous works carried out in our research group [162] a system based just on the usage information is built and then it is combined with the content information to improve the performance of the system.

The system enhances the performance of a web usage mining application including a semantic analysis of the content information. As every web usage mining process, it can be divided into three main steps: data acquisition and pre-processing, pattern discovery and analysis and exploitation. These phases are going to be explained in detail in the following subsections.

Although this is a preliminary system we claimed that the combination of both usage and semantics can lead to more accurate and richer recommendations and moreover it gives to the travel agents greater insight about the real interests of the tourism.

The screenshot shows the home page of the HONDARRIBIA-IRUN Turismoa website. At the top, there are logos for 'HONDARRIBIA-IRUN Turismoa' and 'CALIDAD TURISTICA'. A navigation menu includes links for Bidasoa, Thematic tourism, Professional Tourism, Gourmet tourism, Where to sleep, Agenda and news, Our suggestions, Practical Information, and Downloads and Multimedia. A search bar is located on the right. The main banner features an aerial view of the Bidasoa bay with the slogan 'a little bit of everything'. Below the banner, there are several content blocks: 'Calendar' with a link to 'SABOREA IRUN (Irún)' and a 'Cross-border Agenda' button; 'Our proposals' with two items: 'An ocean of possibilities' and 'Txingudi in GPS'; 'TOP Experiments' featuring a dish of food; and 'Highlights' with a list of items: 'Visit Gipuzkoa', 'Traditional restaurants', 'Innovative Cuisine', 'Hotels', and 'The Basque Coast seen from the Air'.

Figure 5.2: Appearance of the home page of BTw website.

5.5.1 Data acquisition and pre-processing

Two types of data were acquired, on the one hand, the usage data, and on the other hand, the content data.

Concerning to the first group we obtained log information of the BTw site from January 9, 2012 till April 30, 2012 (a total of 897,301 server requests). In addition to that the content information of the site was also used.

For acquiring the content data, based on the conclusions of the analysis presented in Chapter 4, we used the Wget [124] computer program to retrieve content from the BTw web server. We downloaded the HTML files of the whole website using recursive downloading.

The data, the usage and content data, needs to be pre-processed for being able to use it in the pattern discovery and analysis phase. In the following lines the usage pre-processing and content pre-processing are going to be explained.

5.5.1.1 Usage pre-processing phase

Web server log files follow a standard format called Common Log Format [161]. This standard specifies the fields all log files must have for each request received: remotehost, rfc931, authuser, date, request, status and bytes. The fields we used for this work are the remote host IP address, the time the request was recorded, the requested URL and the status field that informs about the success or failure when processing the request. Figure 5.3 shows some sample lines of a log file.

IP address	Time of record	Requested URL	Status
207.46.13.48	- - [22/Feb/2012:00:04:05 +0100]	"GET /index.php?...&lang=es HTTP/1.1"	200
207.46.19.49	- - [22/Feb/2012:00:04:07 +0100]	"GET /index.php?...&lang=en HTTP/1.1"	200
207.46.19.49	- - [22/Feb/2012:00:04:07 +0100]	"GET /index.php?...&lang=es HTTP/1.1"	200
66.249.72.32	- - [22/Feb/2012:00:04:09 +0100]	"GET /index.php?...&lang=es HTTP/1.1"	200
207.46.99.49	- - [22/Feb/2012:00:04:12 +0100]	"GET /index.php?...&lang=fr HTTP/1.1"	200
207.46.19.49	- - [22/Feb/2012:00:04:13 +0100]	"GET /index.php?...&lang=en HTTP/1.1"	200
207.46.19.49	- - [22/Feb/2012:00:06:06 +0100]	"GET /index.php?...&lang=eu HTTP/1.1"	200
73.224.15.77	- - [17/Sep/2012:00:00:00 +0200]	"POST /administ...index.php HTTP/1.1"	301
13.4.215.228	- - [17/Sep/2012:10:21:58 +0200]	"GET /templates/...logo.gif HTTP/1.1"	304
194.69.224.7	- - [18/Sep/2012:09:16:31 +0200]	"GET /templates/...uery.js HTTP/1.1"	200
194.69.224.7	- - [18/Sep/2012:09:16:33 +0200]	"GET /images/...Button.gif HTTP/1.1"	200
194.69.224.7	- - [18/Sep/2012:09:16:33 +0200]	"GET /templates/...logo.gif HTTP/1.1"	200
194.69.224.7	- - [18/Sep/2012:09:16:33 +0200]	"GET /templates/...ogo2.gif HTTP/1.1"	200
194.69.224.7	- - [18/Sep/2012:09:16:33 +0200]	"GET /templates/...pttl.gif HTTP/1.1"	200
194.69.224.7	- - [18/Sep/2012:09:16:35 +0200]	"GET /templates/...lrun.png HTTP/1.1"	200

Figure 5.3: Sample lines of *bidasoat turismo* log file.

The pre-processing phase is a very important phase, the deeper we do this process the better will data adequate to our final objective.

In this phase, first of all we removed erroneous requests, those that had an erroneous status code (client error (4xx) and server error (5xx)). We therefore only took into account successfully processed requests. The next step consisted of selecting the requests directly related to the user activity. User clicks indirectly send many web browser requests to complete the requested web page with for example images, videos, style (css) or functionalities (scripts), as it is commented in Chapter 2. All these indirect requests were removed. We also removed requests related to the web administration activity, which could be detected in our case by analysing the URLs' sub-fields. We then selected and normalized the format of the parameters of the URLs so that only those that really influence the web pages' final appearance were taken into account. As a result of this processing, different URLs with the same appearance are considered to be identical. We then carried

out the user identification and the session identification.

We completed the user identification process based on IP addresses and we fixed the expiry time of each session to 10 minutes of inactivity [163]. We selected the most relevant sessions obtained (those with a minimum activity level; 3 or more clicks), and removed the longest sequences (those with more than 86 requests; out of 98% percentile), with the assumption that long sequences are outliers and might be caused by some kind of robot, such as, crawlers, spiders or web indexers.

Finally, the last stage of the pre-processing consisted of removing requests related to cultural programming and news. We noticed that nearly 35.9% of the requests belonged to this type of pages but since the information they contain is volatile and they would request another kind of treatment, techniques and goals, we decided to keep their analysis outside the scope of this work.

After the whole phase the database contained 55,454 user requests divided in 9,549 sessions, with an average length of 5.8 requests.

5.5.1.2 Content pre-processing phase

The pre-processing of the content consisted of filtering the HTML files. We used an HTML parser in order to extract the textual parts of the website and we filtered the menus of each page in order to work with the real content of the web pages. Then we followed the standard pre-process carried out for Natural Language Processing: the stop word deletion and Porter stemming algorithm [127] (see Section 4.3 for more information).

Although we downloaded every HTML file of the website, the website can be accessed in 4 different languages: French, Spanish, English and Basque. The website of BTw is symmetric, that is, the site is identical for each language and there are the same number of pages in each language. In this work we have used only the English part of the site. So, we removed all HTML files belonging to other languages.

Moreover, all the dynamic part of the website (the one that changed the content constantly as news, agenda and so on) was erased. As in this preliminary work the aim was to create a link prediction system, it has no sense to propose links that are no longer available in the site. We have only worked with the static part of the site (the part that has the main information of the site).

After the pre-processing phase we obtained a 231 URL collection for working with.

5.5.2 Pattern discovery and analysis

We are going to split the pattern discovery and analysis phase into two different subsections. On the one hand we are going to explain the navigation profile discovery and on the other hand the enrichment that we carried out using semantic information.

5.5.2.1 User navigation profile discovery

This is the stage that, taking the user click sequences as input, is in charge of modelling users and producing user navigation profiles (Figure 5.4 shows the process of navigation profile discovery). Most commercial tools perform statistical analysis on the data collected. They extract information about the most frequently accessed pages, average view times, average lengths of paths, etc., which are generally useful for marketing purposes. But the knowledge extracted from this kind of analysis is very limited. Machine learning techniques are generally able to extract more knowledge from data. In this context, unsupervised machine learning techniques have shown to be adequate to discover navigation profiles [164].

We used the PAM (partitioning around medoids) clustering algorithm [103] and the edit distance sequence alignment method [117, 77] as a metric to compare sequences. As it is explained in Section 3.2.2.1 the clustering method used for sequences is PAM, which is the most common implementation of K-medoids where instead of using centroids, medoids are used. In addition edit distance was selected due to the fact that it is the distance for comparing sequential data.

The aim of using PAM was to group users that show similar navigation patterns. PAM requires the K parameter to be estimated. This parameter is related to the structure of the database and the specificity of the generated profiles, where the greater its value the more specific the profiles will be. We fixed, after an analysis, the maximum number of clusters for PAM algorithm to 50 in this preliminary system.

The outcome of the clustering process was a set of groups of user sequences that show similar behaviour. However, we intended to model those users or to discover the associated navigation patterns or profiles; i.e. to find the common click sequences appearing in the sessions in a cluster.

For profiling we used SPADE (Sequential Pattern Discovery using Equivalence classes) [110], an efficient algorithm for mining frequent sequences, which we used to extract the most common click sequences in the cluster (see Section 3.2.2.4). In order to adapt SPADE to the sequential nature of

the data we matched each user session with a SPADE sequence, with events containing a single user click. The application of SPADE provides for each cluster a set of URLs that are likely to be visited in the sessions belonging to it. The number of proposed URLs depends on parameters related to the SPADE algorithm, such as minimum support and maximum allowed number of sequences per cluster.

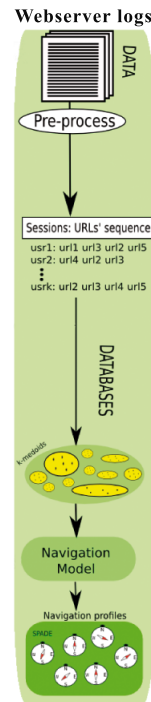


Figure 5.4: Process of the navigation profile discovery.

The optimum value for the parameters depends on the characteristics of the clusters (such as size, structure, compactness, etc.), which will vary from one cluster to another. Hence, it is important to regulate the system so that it finds an adequate number of URLs to propose.

The generated navigation profiles will be used for link prediction. In this context, proposing a large number of links would stress the user and we therefore decided to fix the maximum number of URLs per profile to 4.

5.5.2.2 Enriching the profiles with semantics

The profiles have been generated up to this point using only usage information but we propose to enrich the profiles with semantic information to improve the performance (see Figure 5.5). Obviously the navigation pattern of the users depends on their interests, and, as a consequence, URLs with similar or related content to the ones appearing in the navigation profile will also be interesting for the user.

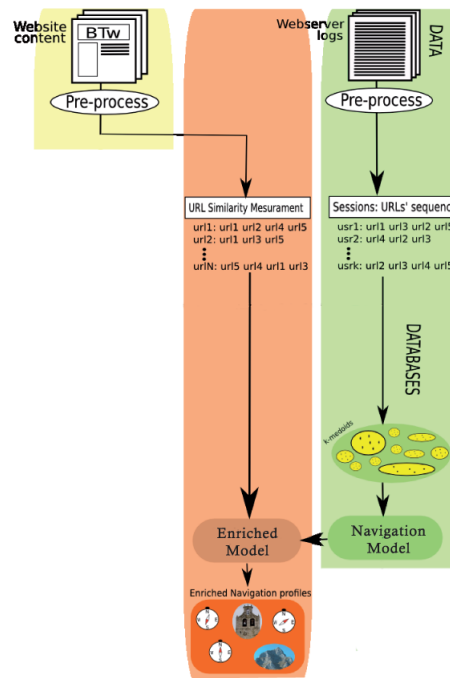


Figure 5.5: The process of enriched navigation profiles.

In this preliminary system, for the enrichment process of the navigation profiles using content mining tools, we used two types of tools for finding similarities between URL contents. The first option used was a search engine based approach using MG4J [165] (see Section 4.4.1.1). We used this tool with TF-IDF distance to obtain similarity values between every possible pair of URLs in the website. This gives us the chance to obtain for each URL a list of URLs ordered by semantic similarity. The second option used in this preliminary system was a keyword extractor based approach (KYWD) based on Yahoo Term Extractor (see Section 4.4.1.2). Once the keywords

of each URL were extracted, we used the cosine similarity [114] to compare URLs.

The URLs with larger similarity value will be the semantically more similar ones. The two previous approaches were used in the same way to enrich usage information based navigation profiles: we added to the profiles generated using SPADE two extra links, the most semantically similar ones, for each proposed link. In case these URLs already appeared in the profile we did not taken them into account. Furthermore, the KYWD gives the option to extract semantic information from the obtained profiles. With this aim we analysed which were the most important (frequent) keywords in each of the generated profiles (in the next subsection the process of the assignation of keywords to clusters is explained).

5.5.3 Exploitation

This is the part that needs to be done in real time. Up to now, the system identifies groups of users with similar navigation patterns and generated navigation profiles or most common paths for each of the groups. At this point we need to use that information to automatically propose links to new users navigating in the web. We propose the use of kNN [166] learning approach to calculate the distance of the click sequence (average linkage distance to the medoid based on edit distance [117]) of the new users to the clusters generated in the previous phase.

Our hypothesis is that the navigation pattern of that user will be similar to the navigation profile of its nearest clusters. As a consequence the system will propose to the new user the set of links that models the users in the clusters (see Figure 5.6).

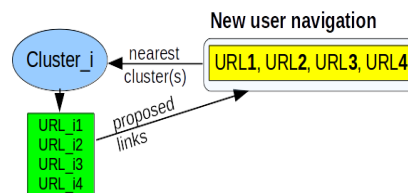


Figure 5.6: Exploitation phase.

5.5.4 Experiments: results and analysis

5.5.4.1 Experimental setup

In order to evaluate the performance of the whole process, we applied the holdout method dividing the database into two parts. One part was used for generating the clusters and for extracting user profiles, and, another part for testing. To simulate a real situation we based the division of the database on temporal criteria: we used the oldest examples (66% of the database, 6366 user sessions) for training and the latest ones (33%, 3183 user sessions), for testing (see Figure 5.7). The test examples were used for computing statistics (in this case precision, recall and F-measure are used) based on results for each one of the new users. We compared the number of proposed links that were really used in the test examples (hits), the number of proposals that were not used (misses), and the number of links used by the test users.

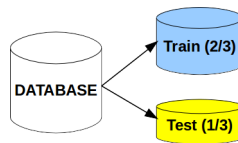


Figure 5.7: Division of the database.

We validated the system in two different situations: when no content information was used (SP) and when profiles were enriched with semantic information extracted from the content. For the latter, we used two tools to compare URLs: the search engine based comparison (MG4J) and the keyword based comparison (KYWD).

We validated the system from two points of view the user point of view and the service provider point of view. For the first part we used the test examples as described in the exploitation section and then we compared the automatically generated links with the real click sequences of the users. For the latter, we analysed the description based on the most frequent keywords for each cluster. For calculating the most frequent keywords of the clusters, the keywords of the profiles of each cluster, i.e., the automatically generated links were extracted and the most frequent ones of this list were considered to be the cluster keywords. This was done with the assumption that if the automatically generated links are the most frequent ones of the clusters, their keywords will determine the theme of the cluster.

We performed the evaluation taking into account that in real executions, when a user starts navigating, only its first few clicks will be available to be used for deciding the corresponding profile and proposing new links according to it. We simulated this real situation using 10%, 25% and 50% of the user navigation sequence in the test examples to select the nearest cluster or profile (see Figure 5.8).

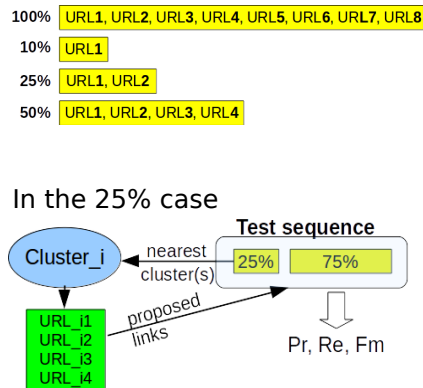


Figure 5.8: Simulation of the real situation.

An ideal system would maintain precision and recall as high as possible. But, in order to compare the enriched system and non enriched system, we kept the number of proposed links not too high (around 50% of the average sequence length) and focused on recall because it gives us an idea of the achieved coverage, i.e., the number of links really used in the test examples that our system proposes. In order to ensure that precision values do not suffer a sudden drop in this preliminary system we present values for two statistics: recall (Re) and F-measure (Fm). Note that the obtained values could be seen as a lower bound because, although not appearing in the user navigation sequence, the proposed links could be useful and interesting for her. Unluckily their usefulness could only be evaluated in a controlled experiment using the user feedback.

5.5.4.2 Results and analysis

We calculated two values for the used statistics: the profile evaluation (Re-Pro, FmPro) which takes into consideration the whole test sequence, and the link prediction evaluation which takes into account the values calculated using only the clicks in the test sequence that were not used to select the

nearest profile (Re, Fm); that is, taking into account the remaining 90%, 75% or 50% (for the cases 10%, 25% and 50% respectively).

We present in Figures 5.9 and 5.10 recall and F-measure values for the usage based system (dashed lines, called SP system) and the system combining usage and content information (MG4J and KYWD systems). In order to analyse these results it is important to note that a first analysis of the results shows that most of the users start navigating from the initial page of the website and visit the agenda. Although this is interesting information for the travel agent, the proposal of these two URLs as part of the profiles inflates the values of the calculated statistics, and, as a consequence we removed them from the generated profiles.

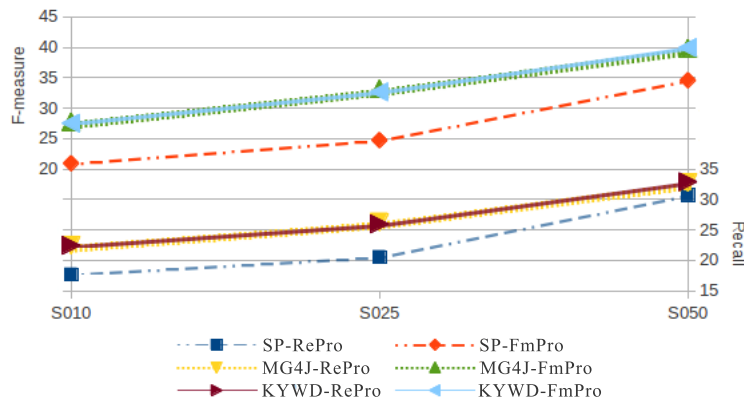


Figure 5.9: SP vs semantically enriched profile evaluation.

The first conclusion we can draw from the results is that even if the values of the measured metrics vary depending on the selected option, all of them are able to predict a certain percentage of the links a new user will be visiting.

We adjusted the parameters so that both systems proposed a similar number of links to the new users: 3.5 in average for SP option and 3.8 in average for MG4J and KWYD options. Note that when larger the proposed number of links is, smaller is the support of some of them, so the system is risking more and, as a consequence, a drop in the F-measure value is very probable as a consequence of a drop in precision.

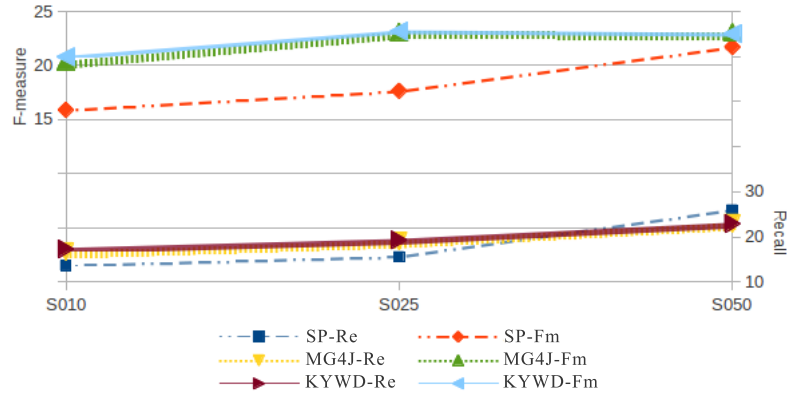


Figure 5.10: SP vs semantics enriched link prediction evaluation.

Graphics in Figures 5.9 and 5.10 show that when enriching the system with content information recall and F-measure values increase, that is, the system guesses more links among the ones really used by the users (Re). The improvement is larger in the case of profile evaluation but even in the link prediction scenario, content information seems to be important. On the other hand, this improvement is more evident at early stages of the navigation when the usage information is very limited. Those are the moments when the prediction can probably contribute more to the user navigation experience becoming more satisfactory.

Finally, if we analyse the semantics of the generated profiles based on the extracted keywords, we realize that most of the clusters group users with similar interests. Table 5.1 shows an example of the semantics of some of the clusters (each row of the table corresponds to a cluster) and some of their related keywords. The names of the themes appearing in the left hand column in Table 5.1 were assigned manually whereas the keywords on the right hand column were obtained automatically. The results in the table clearly show that the users clustered in different groups based on their usage patterns, besides navigating in a different way, have clearly different interests and our system is able to extract information about them. Calculating statistics of the number of users in each cluster the service providers could obtain very useful information about the main interests of the people accessing BTw and use it in the future for marketing campaigns or modifications in the website.

Theme/Cluster	Keywords
Mountain	cycle trails, mountain paths, path cycle
Sea	bay, river, ocean bay, river boat trips, natural treasures
Accommodation	hotel, accommodation, youth hostels
Cuisine	sugar, eggs, recipes typical product, innovative cuisine
History	walled city, borda, palace
Events	events activities, markets fairs, rural sports

Table 5.1: Semantics of some of the obtained profiles.

5.5.5 Summary

The system presented was designed without disturbing the users and based just on server log information, content information and machine learning techniques. It identifies different groups of users, builds the corresponding profiles, generates automatically useful link proposals for new users, and moreover, it gives insight about the users' preferences to the tourism agents.

The results of this preliminary system, showed that the use of the semantic knowledge extracted from the website content information improves the performance, recall and F-measure values, of the system. Furthermore, using content information gives the option to enrich the generated profiles with semantic information that can be very useful for service providers.

This work opens the door to many tasks. First of all the previous results showed having a topical structure of the website to be interesting. Unfortunately a lot of websites do not have an underlying ontology containing their thematic structure and many times the communication with the service providers is not fluent. Thus the thematic structure needs to be obtained automatically. The topic modelling technologies could be another option to extract semantic knowledge from the websites' content. Furthermore, more sophisticated strategies to build semantic could also be explored. Secondly a sounder evaluation methodology could be used such as 10-fold cross-validation and a deeper analysis for the selection of the number of cluster K could be also done. As a result, a system including all these characteristics is going to be presented in the next section.

5.6 Global system

The global system, which is going to be presented in this section, is an expansion of the preliminary system, where using the acquired information, usage based navigation profiling, semantically enriched navigation profiling and interest profiling are carried out.

5.6.1 Navigation profiling

This system was built using log information of BTw recorded during 10 months: from January, 2012 to October, 2012. So, 5 months more of data were added to the database used in the preliminary system. They contained 3,636,233 requests, which were reduced to 168,556 after the data pre-processing phase described in Section 5.5.1. Table 5.2 shows a summary of the number of requests and sessions after the different pre-processing stages we applied.

	Requests	Sessions
In log files	3,636,233	
Valid requests	2,002,827	
Sessions after sessioning and filtering	765,712	66,897
Sessions after removing cultural programming and news	168,556	21,917

Table 5.2: Number of requests and sessions after the different pre-processing stages.

According to the numbers in Table 5.2 the final database contained 21,917 sessions with an average length of 7.7 clicks.

Based on the experience acquired with the preliminary system, we first generated navigation profiles by combining PAM with SPADE and compared these profiles to those for new users navigating the website. In order to carry out the evaluation we used a 10-fold cross-validation methodology, dividing each folder into a training set (15,341 examples), validation set (4,380 examples) and test set (2,196 examples). We used the validation set to select K (the number of clusters) and the test set to evaluate the performance of the system.

As new sessions were included to the database, the internal structure of the data is completely unknown and we therefore tried a wide range of values for K to select the optimum number of clusters; we tried values ranging from 20 to 700. Although an usual exploration limit for the number of clusters is \sqrt{n} , which makes it advisable to explore values of K up to 124 for 15,341 training examples, and having more profiles makes the phase of identifying the profiles of new users navigating the website more costly, the exploration

area was so extensive because we prioritized finding the real structure of the data and the acquirement of good quality profiles.

The best way to select the optimum number of clusters would be to use a Cluster Validity Index (CVI). However, probably because most CVIs were designed and tried for a small number of clusters, the CVIs reported to be the best in [167] did not provide any coherent result. We therefore evaluated the quality of the partitions according to the precision, recall and F-measure performance metrics as it was done in the preliminary system already explained.

The generated profiles were evaluated by comparing them to new users navigating the website. We simulated this real situation using 25% (more or less 2 links, because the click sequences have on average 7.7 links) of the validation or test sequences to select the profile for the new user according to the built model. 50% and 75% were discarded because we considered more important for the system to react in early stages of the navigation. 10% was also discarded due to the fact that taking into account that the click sequences have 7.7 clicks in average the 10% of the navigation would be 0-1 clicks and what does not ensure a minimum number of clicks.

According to our previous experience in a similar work [168] new users might not be identical to any of the profiles discovered in the training set; their profile might have similarities with more than one profile and, as a consequence, the diversification helps; it is better to build the profiles of the new users dynamically based on some of their nearest profiles. We used 5-*NN* to select the nearest clusters and combined the profiles of the two nearest clusters with defined profiles, fixing URL selection probabilities according to their distance. We combined these to propose profiles containing at most 4 URLs; those with the highest support values. If there were not enough URLs exceeding the minimum support value the profiles could have less than 4 URLs.

The greater the number of URLs proposed as profiles, the smaller will be the significance of some of them and the risk taken by the system will thus be greater. As a consequence, the values for precision will probably drop. In contrast, by limiting the maximum number of URLs proposed for each profile to 4 the recall values will never reach 1. Since the average length of the sequences is 7.7, if we propose a profile (4 URLs) based on 25% of the navigation sequence (more or less 2 URLs), we would have sequences of $4 + 2 = 6$ URLs, and the value of recall would be at most 0.78.

Table 5.3 shows the average results for the evaluation of the navigation profiles (precision, recall and F-measure) obtained for the validation set in the 10 folds for different values of K. The PrIncr, ReIncr and F1Incr

columns show the increment of the performance metrics per extra cluster. The second column, $\text{avg}(\text{nURL}/\text{prof})$, shows the average number of links proposed in each profile. As we explained in the previous paragraph, this value limits the maximum recall value that can be obtained; in this case to 0.64.

K	avg(nURL/prof)	Pr	PrIncr	Re	ReIncr	F1	F1Incr
20	2,91	0,532		0,274		0,361	
40	2,93	0,556	0,00123	0,290	0,00081	0,381	0,00099
80	2,89	0,590	0,00085	0,309	0,00046	0,405	0,00060
120	2,93	0,603	0,00033	0,321	0,00032	0,419	0,00035
160	2,95	0,615	0,00029	0,331	0,00024	0,430	0,00027
200	2,89	0,633	0,00046	0,337	0,00016	0,440	0,00025
300	2,90	0,648	0,00015	0,347	0,00010	0,452	0,00012
400	2,90	0,656	0,00008	0,352	0,00005	0,458	0,00006
500	2,90	0,662	0,00006	0,356	0,00004	0,463	0,00005
700	2,96	0,663	0,00000	0,362	0,00003	0,468	0,00003
Test(300)	2.89	0.642		0.344		0.448	

Table 5.3: Profile evaluation: average validation set results for the 10-fold cv.

The values of the performance metrics in Table 5.3 improve as the number of clusters increases, what means that the users of the system are very diverse and, as a consequence, smaller clusters better capture the different types of existing profiles. However, if we analyse the improvements in the performance metrics for each new cluster added, or normalized improvement differences, we detect an elbow, or change in order of magnitude of the improvement (marked in bold in Table 5.3), in $K=300$. We therefore fixed the number of clusters to 300. Thus, from this point on, we set the system to 300 clusters, with a maximum of 4 URLs provided as profile. The last row in 5.3 shows the average results of the 10 folds, for the test set and $K=300$.

The results show that the values obtained for the test samples are similar to those obtained for the validation samples, thus confirming that the behaviour of the system is stable. Moreover, if we focus on the values of the performance metrics we can say that the profiles proposed to the new users fit in more than 60% with the real navigation sequence and that the generated profiles therefore have good quality.

Nevertheless, a qualitative analysis of the generated profiles showed that many of the users visited the home page at intermediate points of the navigation. We consider this provides information about a user's navigation behaviour; it could suggest situations such as being lost or changing their mind about their interests... so we still consider this as part of the profile. However, we consider that if the profiles are used to propose links to new

users the proposal of the home page would not provide any help for them and we therefore decided to remove the home page from the generated profiles.

In this situation we generated new profiles in the same conditions in which we generated the previous ones ($K=300$, maximum number of URLs = 4, minimum support = 0.2) but not admitting the home URL as a proposal. This will obviously affect the precision and recall values because having a forbidden URL reduces the highest achievable values. In order to obtain more realistic values, misses in the home page have not been counted when calculating recall. This process reduced the average number of URLs proposed per profile to 2.64, precision to 0.509 and recall and F-measure to 0.262 and 0.346 respectively. What this means is that half of the proposed URLs were used by the new users and the system was able to propose more than a quarter of the URLs that were actually used.

In order to evaluate how good the generated profiles would be for link prediction, we simulated the real situation by using 25% of each test sequence to select the profile of the new user according to the built model and compared the profile with the rest of the sequence (75%).

As the home page of BTw can be reached from any point in the website, it did not make any sense to predict or propose the home URL. The performance metrics were thus calculated for the option that does not admit the home URL as a proposal, where the values obtained were 0.267 for precision, 0.155 for recall, and, 0.196 for F-measure. These values showed that, even limiting the system to not proposing the home URL, our system is able to predict some of the links the new users are using and thus help users to have a more pleasant navigation experience.

5.6.1.1 Evaluation using semantic structure

As we mentioned in the evaluation of the preliminary system, the values obtained when evaluating user profiles or the link prediction system should be seen as a lower bound, since the user could find the proposed links to be of interest even if they do not appear in the user navigation sequence. The semantic structure of the website could undoubtedly provide hints about whether or not the links proposed are of interest to the new users and it could also be used to explore more generic navigation profiles that represent the navigation of the users through different interest areas, instead of specific URLs. With this aim we combined the semantic structure of the website, with the navigation profiles obtained by the system; i.e. the sets of URLs that the user is likely to visit.

First of all the structure of the BTw must be extracted. This could be provided by the website designers, stored in an ontology or extracted automatically using Natural Language Processing (NLP) techniques, such as topic modelling techniques.

5.6.1.1.1 Automatic extraction of the semantic structure of BTw

In order to obtain the thematic structure of the website, we pre-processed the document collection of the web BTw as it is explained in Section 5.5.1.2 and used the Stanford Topic Modelling Toolbox (STMT) [169]. There are more tools that implement topic modelling techniques but STMT has some advantages: it is intuitive and it is an open source tool what provides the option to modify the application.

We gave as input to STMT the dataset containing all the content information of each URL. After running STMT we obtained a list of topics represented by the keywords related to them (topic-keyword list) (see Table 5.4) and a vector for each of the URLs in the database, containing the probability or affinity to each of the topics (URL-topic probability vector). For BTw we obtained a URL-topic vector for each of the 231 URLs, which could be represented as:

$$UT_d = (A_{u_d}^{T_0}, A_{u_d}^{T_1}, \dots, A_{u_d}^{T_N}) \quad (5.1)$$

Where N represents the number of topics, UT_d represents the vector corresponding to URL_d and $A_{u_d}^{T_n}$ represents the affinity of URL_d with topic T_n .

We performed several experiments to determine the optimum number of topics. The decision was made by analysing the coherence of the keywords proposed by the STMT tool for each of the topics and trying to find a trade-off between the number of topics and the coherence of the keywords proposed for each topic; i.e. we selected the minimum number of topics with a coherent set of keywords: 10 main topics or abstract themes. Once the STMT tool had extracted the different topics from the URL collection we named them manually, inferring a topic title based on the keywords grouped under each topic. In this way the presented results will be more readable. Table 5.4 shows the titles we selected for the 10 topics proposed by STMT and some of the related keywords.

topic Title	topic-keyword List
Nature (Na)	mountain, river, beach, bay, ...
Historical Monuments(HM)	church, chapel, castle, history, ...
Cuisine (Cu)	cuisine, restaurant,cider-house, ...
Accommodation Camping (AC)	accommodation, pilgrim, camp, sleep, ...
Accommodation Hotel (AH)	room, accommodation, countryside, ...
Events (Ev)	festival, exhibition, theater, event, ...
Culture (Cl)	culture, organization, artist, visitor, ...
Sea&Sports (SS)	sea, surf, sport, kayak, ...
Sports (Sp)	sport, golf, tennis, pelota, ride, ...
Tradition (Tr)	tradition, celebration, typical, activity, ...

Table 5.4: topic-keyword list proposed by STMT and titles for each topic.

Slightly changing the number of topics would change the structure, but not too much. For example, if 9 topics were extracted instead of 10 the system would group the Sea&Sports and Sports topics together. Furthermore, we discussed the topics obtained with the staff of the DMO and they confirmed that the proposed structure was coherent with their aims when designing the website.

The screenshot shows the website for Hondarribia-Irún Turismooa. The header includes the logo, navigation links in Euskara, Español, and Français, and a search bar. The main navigation menu includes: Bidasoa, Thematic tourism, Professional Tourism, Gourmet tourism, Where to sleep, Agenda and news, Our suggestions, Practical Information, and Downloads and Multimedia. The current page is 'Home > Thematic tourism > Mountains, paths and cycle trails'. The page title is 'Mountains, paths and cycle trails'. The content includes a list of sub-topics: 'Mountains, paths and cycle trails', 'The bay and the river', and 'The ocean'. The main text describes the benefits of hiking in the countryside, mentioning 'Bidasoa-Txingudi' and 'rural footpaths'. A photograph shows a group of people hiking on a path through a forest.

Figure 5.11: Example of BTw web page.

The other output that the STMT tool provides, a URL-topic probability vector per URL, could be used to automatically analyse the theme of the content associated with each of the URLs. Although there could be URLs containing various topics, in this work we assigned a single topic per URL: the one with the greatest probability in the URL-topic vector. This decision was taken as the difference between the most likely topic and the second

one was more than one order of magnitude. For example, the concrete URL shown in Figure 5.11, is about nature and if we analyse its URL-topic vector, the highest topic value is Nature with 0.96 and the second largest Sports with 0.005. Based on this labelling we analysed the distribution of URLs of the topics in BTw showed in Figure 5.12.

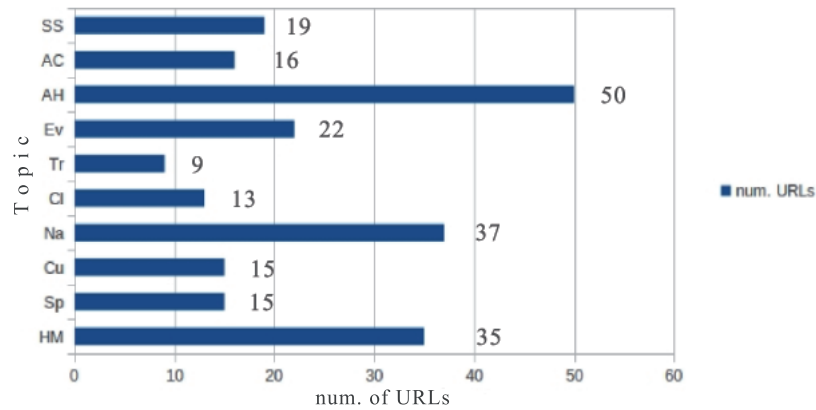


Figure 5.12: Number of URLs (X axis) per topic (Y axis)

As it can be seen in Figure 5.12, the number of URLs per topic is not balanced so if the user accesses are to be carried out at random, some of the topics such as Accommodation Hotel (AH) or Nature (Na) would be much more likely to be accessed over other as Tradition (Tr) or Culture (Cl).

5.6.1.1.2 Combining semantic structure with navigation profiles

This combination will allow us to evaluate the navigation profiles according to interests and will also allow a further evaluation of the quality of the link prediction tool. It will allow us to measure to what extent the links proposed to the new user are of interest for her.

With this aim, we first used the URL-topic probability vector provided by the STMT tool (see Section 5.6.1.1.1) to automatically analyse the theme of each of the URLs. In the generated navigation profiles we replaced the URLs by their main topic and calculated precision, recall and F-measure in

the same way as we did in the previous section, but using the new profiles. Table 5.5 shows the results obtained and compares them to the performance metrics calculated according to the URLs.

Evaluation	Criteria.	N. prop	Pr	Re	F1
Profile	URL	2.68	0.642	0.344	0.448
	Topic		0.909	0.730	0.810
Link Prediction	URL	2.68	0.267	0.155	0.196
	Topic		0.736	0.598	0.660

Table 5.5: Evaluation of profiles and link prediction according to interests.

As can be observed our intuition was correct. Precision and recall values, for example, soar up to 90.9% and 73% when evaluating profiles and up to 73.6% and 59.8% in the case of link prediction, which means that the generated navigation profiles are accurate according to interests and, moreover, that a high percentage of the links proposed by our system is of interest to the new users. This will obviously contribute to make their browsing experience more pleasant and will help them to achieve their objectives faster.

5.6.2 Enriched navigation profiling

Web usage information can be combined with content information to enrich navigation profiles as it was done in the preliminary system. In this case, instead of using MG4J or KYWD to calculate distance values between URLs, topic modelling techniques are going to be used.

5.6.2.1 Topic structure based URL distances

The output of the topic modelling process can be used to compute the semantic similarity of URLs. Since they extract only the thematic structure of the documents, but not the dissimilarity values we need to include some distance calculation process. The URL-topic vector provides a semantic characteristic representation of each URL and it can be directly used for document comparison. Different metrics are used for calculating the distance between numerical vectors, such as euclidean distance, Hellinger distance, cosine distance and KL divergence among others (see the Section 3.3).

The distances that are listed above have been proved for calculating the distance between URL-topic vectors. The most common one for this type of works is Hellinger distance [115] and we have verified that this is the distance which more coherent results returns giving small values to similar documents

and large values to dissimilar ones. Therefore, the similarity values of our URLs were calculated using Hellinger distance.

5.6.2.2 Enrichment of navigation profiles

The system we designed offers the option to enrich the link proposals by combining the profiles generated using clustering + SPADE with links proposed based on semantic information; i.e. the system would enrich profiles generated based on usage information with semantic information.

We implemented and evaluated this new option where in order to locate the system in the same point of the learning curve, we limited the usage profiles to two URLs and enriched them with a single URL; the semantically most similar URL to the one with greatest support among the ones obtained with usage information. When the proposed URL already appeared in the profile, we selected the next one with higher similarity.

This variant of the system for link prediction was evaluated by calculating the same performance metrics we calculated to evaluate the previous system. Table 5.6 shows the values of the performance metrics when the link proposal is diversified using semantic information and when link proposals are made based only on the navigation profiles. The number of proposed URLs is in the same range for both options although it is slightly greater for semantically enriched link proposals. The values of the three performance metrics calculated are also in the same range for both cases, although they are slightly better for the new system.

Following the idea presented previously in the preliminary system, we could conclude that when using semantic information, such as the use of topic modelling in this case, to diversify the links proposed to new users and introduce other products that might be of interest to the service provider, the system is still able to propose links to the new users that seem to be of great interest to them. In this case the system is achieving a double objective: it is helping the DMO staff in their campaigns and it is also providing a tool to enable the users to achieve their objectives faster and in a more pleasant way.

System	N. prop	Pr	Re	F1
Usage	2.680	0.736	0.598	0.660
Usage+Semantic	2.902	0.736	0.603	0.663

Table 5.6: Evaluation of semantically-enriched link proposals according to interests

5.6.3 Interest profiling

The third branch of the created system is the generation of semantic profiles (see Figure 5.13). Using web usage information and content information it is possible to profile users' interests. In order to be able to profile the interests of users accessing BTw, we first need to link the information appearing in each URL to interests. We combined the information extracted in Section 5.6.1.1.1 with usage information to deduce the interests of the users accessing BTw.

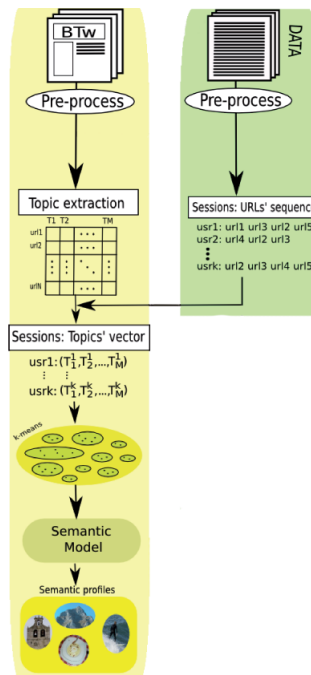


Figure 5.13: The process of interest profiling.

5.6.3.1 User semantic profile discovery

In this phase the combination of web usage and content information is used in order to extract knowledge about the interests of the users accessing BTw. The process consisted in detecting sets of users with similar interests and extracting their semantic profiles.

Before starting the profile discovery process we discussed the information provided by the topics accessed with the staff of the DMO. Although topics related to accommodation might be very important for the tourist, they are nearer to a requirement than to a preference or interest. We therefore decided to delete them from the interest profiling phase. However, the analysis carried out on sessions accessing only accommodation-related URLs provided information of interest to the staff of the DMO: 17% of the BTw user sessions were completely devoted to accommodation. We removed these sessions from the database for discovering the semantic profiles of the users.

The web page can be accessed in four different languages (Basque, Spanish, French and English) and we used the access language as an indicator of the origin of the users. The access language will help us to differentiate between local people (those accessing the site in Basque), Spanish people or people from Spanish-speaking countries (those accessing the site in Spanish), French people or people from French-speaking countries (those accessing the site in French) and, finally, people from the rest of the world (those accessing the site in English).

Moreover, we performed a monthly analysis for being able to detect if people look for different things depending on the time period. Taking this into account, we carried out three analyses of the users' interests: a global analysis, an analysis that took into account the access language and the analysis using the time period. The comparison of the language and monthly analysis with the global one can provide extra information to the staff of the DMO to be used in future marketing campaigns or redesigns of the website and it can also be used to propose specific adaptations depending on the origin of the new user and so on.

In order to perform origin-dependent analysis, we separated the examples in the database depending on the language used. We first identified the language of each link based on the *lang* parameter of the URL. We then assigned a language to each session or sequence of URLs, according to the language with the highest proportion in the sequence. Sequences or sessions with at least 70% of their navigation in the same language were labelled with that language; otherwise they were labelled as multilingual sessions. Thus, bearing in mind that the first access of the page might not be in the desired language, even in the sessions labelled with one of the languages, we allowed a certain degree of mixture of languages. After dividing the database into languages we obtained 5 sub-databases, the sizes of which are shown in Table 5.7. For further analysis, as we could not presuppose anything about the origin of people generating multilingual sessions, we ignored their sessions.

English (en)	Spanish (es)	Basque (eu)	French (fr)	multilingual (mul)
2,630	10,198	2,616	2,784	4,557

Table 5.7: Sizes of the databases divided by language.

The data shows that, as expected, the number of accesses in every language is not the same; accesses in Spanish are more frequent than accesses in other languages. In order to avoid imbalances and to avoid obtaining results biased by the interests of the users navigating in Spanish, we obtained global profiles with a stratified sample of more or less 2,650 examples for each language. To obtain the sample we randomly discarded some of the sessions in Spanish.

In order to perform the monthly analysis, we divided the original database into 9 different sub-databases. One for each month of the data acquired from the DMO staff. We identified the month using the time-stamp that common log format provided. After dividing the database into different months we obtained 9 sub-databases, the sizes of which are shown in Table 5.8.

Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep
2,087	2,706	3,136	2,766	3,049	3,731	5,852	4,307	2,934

Table 5.8: Sizes of the databases divided by months.

5.6.3.2 Session representation

As a first step to discover user semantic profiles we analysed the session-topic relationship and represented the sessions as session-topic vectors. As described in Section 5.5.2.1, the usage data was organized such that each session is represented as a URL sequence. In addition, as described in Section 5.6.1.1.1, by applying topic modelling to the BTw content information we obtained for each URL a URL-topic vector that represents its degree of affinity to each of the 10 topics extracted from the whole website. In order to model the user interests according to their navigation we combined both sets of information: we added the probabilities of the topics for every URL appearing in the session and obtained a vector representation of the user sessions: session-topic vectors.

For a session s of length L , the session-topic vector can be represented as:

$$ST_s = (ST_s^{T_0}, ST_s^{T_1}, \dots, ST_s^{T_N}), \quad (5.2)$$

where $ST_s^{T_n} = \sum_{l=0}^L A_{u_s^l}^{T_n}$ and $A_{u_s^l}^{T_n}$ represents the affinity of URL_l of session s with Topic T_n .

Each session-topic vector represents the degree of affinity of that session to each of the topics (it has been normalized so that the sum of the vector elements is 1). Topics with higher values will denote a higher interest of the user in that topic.

5.6.3.3 Semantic profile generation

This stage is in charge of modelling users and producing user profiles, taking as input the session-topic vectors (i.e. a vector representation of user sessions). We used a clustering algorithm (K-means) [170] to group users with similar navigation patterns and euclidean distance to compare two sessions. Using these techniques, we grouped users showing similar interests into the same segment.

The outcome of the clustering process was a set of groups of session-topic vectors. We used this information to deduce the probability of each topic for the cluster. S_j represents the number of sessions in a cluster j , the cluster-topic vector would be calculated by adding (for each topic) the topic-probability of the S_j sessions in the cluster, as can be seen in equation 5.3.

The cluster-topic vector for cluster j with S_j sessions would be:

$$CT_j = (CT_j^{T_0}, CT_j^{T_1}, \dots, CT_j^{T_N}), \quad (5.3)$$

where $CT_j^{T_n} = \sum_{s=0}^{S_j} ST_s^{T_n}$.

The cluster-topic vector will allow us to identify the most significant topics for each cluster. We could thus use the titles related to these topics to label each cluster; in this work we assigned a single label per cluster. In order to consider a topic to be representative of a cluster, and thus to be able to select it as a label, we required at least 40% cluster-topic affinity; i.e. $CT_j^{T_n} \geq 0.4$.

5.6.3.3.1 Global profiling

In order to evaluate the interest profiling carried out, we compared the topic preferences of all the users calculated without clustering to the topic preferences extracted from the profiling process. The comparison was made by calculating the degrees of topic affinities in two different ways: using the whole dataset and using the output of the global clustering (the clustering performed with the whole dataset). For the first option we computed the topic distribution for all the sessions in the database. For the second option we extracted interest profiles as described above; we then grouped the sessions using the K-means clustering algorithm and assigned the topic with the highest affinity to each cluster, only labelling them if the affinity was greater than 40%. Hence, for clusters labelled with a topic *A*, we could say that all the users in the cluster are interested in topic *A* with its corresponding degree of affinity.

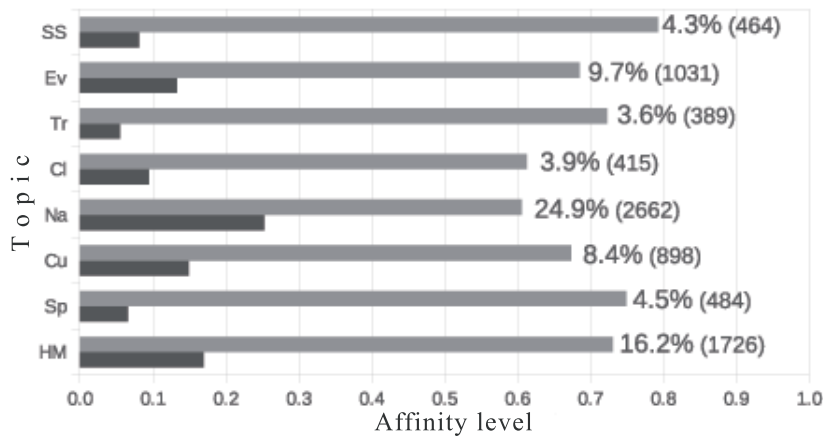


Figure 5.14: Comparison of affinity levels for the whole database and the affinity levels of the profiles obtained with clustering.

Figure 5.14 shows the affinity levels obtained for the two options (whole database, WD.a; and, after the profiling, CL.a). The figure shows in the Y axis the 8 topics selected as meaningful for the staff of the DMO and in the X axis the level of affinity to each of these topics. The figure includes for the profiling option the number of sessions covered by each topic; i.e. the number of users linked to each topic and what percentage of the whole

database this represents.

As can be observed, the topic preference rates obtained for the whole database are very low; the one with the highest rate is around 25%. However, these values risen when profiling is used. For example, there is a group of 464 users whose affinity to the Sea&Sports topic reaches nearly 80% and a group of 2,662 users whose affinity to the Nature topic is 60%. Although some of the clusters were discarded in the CL.a option because we considered them not to be representative, 75.55% of the users were covered by those represented in Figure 5.14.

If we compare the distribution of topics obtained after profiling to the distribution obtained for the whole database, we can easily infer that the use of clustering is a good option because it finds a clear structure in the data; i.e. it finds profiles with a high level of interest in specific topics and thus opens the door to future personalization strategies.

5.6.3.3.2 Language-dependent profiling

We divided the database into 4 segments according to the access language and repeated the profiling phase described in the previous section for each of them. This profiling is important in order to determine the existence or otherwise of any common pattern among users with the same origin and whether or not this pattern differs from the profiles obtained for other languages. If a clear structure existed in the data for each access language the process would discover the interests of users depending on their origin, which would be interesting from two points of view: it would allow specific adaptations to be proposed depending on the origin of the new user and it would provide very useful information to the service provider for use in future marketing campaigns or redesigns of the website.

Table 5.9 shows the results of this experiment. The table shows, for the whole sample and for each of the languages, the popularity or number of sessions profiled as belonging to each of the topics and the average degree of affinity shown by these users. In order to make the results more visual we marked the 4 topics with most users interested as: 1 - , 2- , 3- , 4- . The values in the table show that by dividing the database according to the language used, our profiling methodology was able to find groups of users with high levels of affinity to specific topics. The degree of affinity is on average about 70% and the number of users covered by the significant clusters is on average 78% of the total number of examples (row Covered % in Table

5.9). It can thus be stated that the profiles obtained for each language are robust, in the sense that they have a high degree of affinity with one of the topics and also that they cover the majority of the users who have used the website. We cannot forget that, as shown in Figure 5.14, when no clustering is applied the topic with highest affinity is Nature (with values of around 25%) and for the rest of topics the affinity values are under 15%.

Topic	Global		En		Es		Eu		Fr	
	size	affi.	size	affi.	size	affi.	size	affi.	size	affi.
SS	464	0.79	109	0.83	228	0.76	144	0.77	143	0.79
Ev	1031	0.69	186	0.76	879	0.65	261	0.66	402	0.64
Tr	389	0.72	75	0.89	194	0.77	126	0.67	120	0.69
Cl	415	0.61	98	0.68	231	0.62	74	0.72	77	0.73
Na	2662	0.61	802	0.53	3109	0.58	536	0.67	416	0.69
Cu	898	0.67	337	0.60	1022	0.61	279	0.62	224	0.68
Sp	484	0.75	164	0.67	312	0.65	137	0.78	135	0.77
HM	1726	0.73	522	0.69	1310	0.68	563	0.66	452	0.77
Covered(%)	75.55		87.19		71.44		81.04		70.73	

Table 5.9: Comparison of average affinity values of the different topics for global and language-dependent profiles.

We can compare the ranking of topic preferences for the global database with the rankings in each language according either to the number of users covered or the degree of affinity obtained for each of the 8 topics. If we compare the results with those obtained with the global sample, deviations can be found for both criteria.

Table 5.9 shows that for the global database the most popular topics (those with the most users) were, in descending order: Nature (Na), Historical Monuments (HM), Events (Ev) and Cuisine (Cu). The two most popular topics, Nature (Na) and Historical Monuments (HM), were the same for the users connecting in English and Spanish but were reversed (being almost equal) for users connecting in Basque or French. Moreover, the topics appearing in third and fourth positions, Events (Ev) and Cuisine (Cu), only matched the global sample in the case of users accessing in French, who also show a higher interest in Events (Ev). For users accessing in the other languages the order of these two topics is reversed compared to the global data.

Figure 5.15 shows the comparison of the interest profiles obtained for the whole database with those obtained for each language. In order to show the results we obtained a single value per topic by combining its affinity with its popularity. The combination was performed in each of the databases by normalizing coverage and affinity values so that the sum of all the coverage

values and the sum of all the affinity values were both 1, and then calculating the average of the two values obtained for each topic. To compare the global profiles to the language-dependent ones the differences between the values obtained were calculated and these are shown in Figure 5.15.

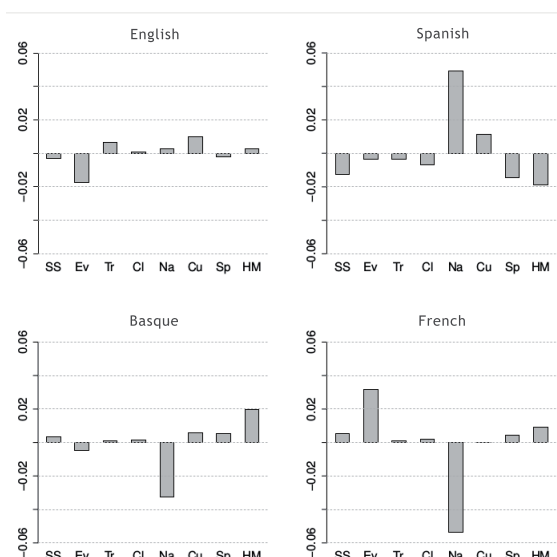


Figure 5.15: Profile differences for the language-dependent profiles compared with global profiles.

The first conclusion we can draw from Figure 5.15 is that for the topic preferences, there are differences between the users accessing in different languages. The figure shows that the results for tourists accessing the site in English are not very different from the global profile but that they seem to be more interested in Cuisine (Cu) and Tradition (Tr) and less interested in Events (Ev) than the generic tourists. However, the results for Spanish tourists are very different from the generic ones, as they seem to be more interested in Nature (Na) and Cuisine (Cu) and less interested in the other topics. In the case of Basque tourists, the main interest appears to be in Historical Monuments (HM) and Sea and Sports (SS and Sp) and they seem to have very little interest in Nature (Na). Finally, French tourists seem to focus on Events (Ev) when visiting the area although they also show an interest in Historical Monuments (HM) and Sea&Sports (SS). However, their interest in Nature (Na) is clearly lower than it is for the global profiles.

5.6.3.3.3 Monthly profiling

In this case we have repeated the global process for the database divided by months. The selection process for each clusters' topic as has been described at the beginning of Section 5.6.3.3. The results of this experiment are shown in the Table 5.10, which shows for each topic and each month the number of sessions that have shown greater interest in that topic, and the degree of affinity shown by these users. In order to make the results more visual we marked the 4 topics with most users interested as: 1 - (**bold + underlined**), 2- (underlined), 3- (**bold + italics**), 4- (*italics*) as it is done for language-dependent profiling.

		SS	Ev	Tr	Cl	Na	Cu	Sp	HM	Covered (%)
Global	<i>size</i>	899	<u>2381</u>	607	746	<u>5569</u>	<i>2300</i>	953	<u>3345</u>	73.73
	<i>affi.</i>	0.72	0.63	0.77	0.59	<u>0.60</u>	0.61	0.69	<u>0.70</u>	
Jan	<i>size</i>	49	<i>120</i>	91	75	<u>534</u>	<i>153</i>	108	<u>288</u>	90.09
	<i>affi.</i>	0.82	0.71	0.85	0.54	<u>0.55</u>	0.67	0.65	<u>0.68</u>	
Feb	<i>size</i>	81	<u>276</u>	49	62	<u>540</u>	<i>268</i>	151	<u>349</u>	83.54
	<i>affi.</i>	0.68	0.62	0.71	0.69	<u>0.61</u>	0.58	0.64	<u>0.70</u>	
Mar	<i>size</i>	115	<u>269</u>	103	69	<u>477</u>	<i>220</i>	129	<u>319</u>	69.63
	<i>affi.</i>	0.70	0.58	0.75	0.71	<u>0.66</u>	0.67	0.63	<u>0.77</u>	
Apr	<i>size</i>	92	<i>213</i>	69	42	<u>548</u>	<i>254</i>	75	<u>286</u>	76.58
	<i>affi.</i>	0.59	0.61	0.77	0.71	<u>0.56</u>	0.56	0.75	<u>0.70</u>	
May	<i>size</i>	138	<u>299</u>	40	54	<u>546</u>	<i>253</i>	97	<u>339</u>	75.31
	<i>affi.</i>	0.57	0.55	0.84	0.70	<u>0.58</u>	0.59	0.59	<u>0.64</u>	
Jun	<i>size</i>	140	<u>282</u>	68	51	<u>740</u>	<i>269</i>	84	<u>445</u>	78.48
	<i>affi.</i>	0.66	0.59	0.73	0.67	<u>0.56</u>	0.60	0.68	<u>0.65</u>	
Jul	<i>size</i>	238	<u>553</u>	133	161	<u>1041</u>	<i>345</i>	125	<u>605</u>	73.03
	<i>affi.</i>	0.69	0.60	0.69	0.53	<u>0.60</u>	0.63	0.75	<u>0.71</u>	
Aug	<i>size</i>	73	<i>202</i>	32	114	<u>860</u>	<u>285</u>	122	<u>394</u>	68.11
	<i>affi.</i>	0.86	0.75	0.85	0.49	<u>0.57</u>	0.59	0.64	<u>0.70</u>	
Sep	<i>size</i>	40	<u>190</u>	44	67	<u>304</u>	<i>111</i>	90	<u>308</u>	53.53
	<i>affi.</i>	0.78	0.63	0.66	0.61	<u>0.67</u>	0.69	0.54	<u>0.67</u>	

Table 5.10: Comparison of average affinity values of the different topics for global and time-dependent profiles.

Analysing the values of the table it can be concluded that, as in the global case, when the database is divided by months the system is able to group users in groups linked to a specific topics with an affinity that is on average between 0.65 and 0.7, and covering more than 70% of the cases. Thus, we can say that the profiles obtained for each month are robust, that is, that they represent a large portion of users with high affinity for specific topics.

Regarding the profiles found we can say that the months of February, March, May, June and July followed the same pattern as the global profile

on the four most popular topics. That is, in all of them it is emphasised the Nature (Na) topic, with certain distance to Historical Monuments (HM), followed by Events (Ev) and Cuisine (Cu). We can find four months in which the profile changed: January, April, August and September. In the first three Cuisine (Cu) becomes more important than during the rest of the year, this effect can come tied to the Christmas, Easter and summer holidays. The case of September is stranger since the two most popular topics are exchanged: Nature (Na) and Historical Monuments (HM).

Figure 5.16 shows the comparison of the interest profiles obtained for the global database with those obtained for each month. In order to show the results we obtained a single value per topic by combining its affinity with its popularity as in the language depending profiling. To compare the global profiles to the time-dependent ones the differences between the values obtained were calculated and these are shown in Figure 5.16.

Figure 5.16 shows the deviations from the overall average behaviour being the most notable ones: increase of interest in Tradition (Tr) in January, decrease of interest in Events (Ev) in January and March, increase of interest in Sports (Sp) in February and March, increase of interest in Cuisine (Cu) in Easter and August, decrease of interest in Cuisine (Cu) in July, increase of interest in Nature (Na) in August and finally the differences in trends between July and August are also notorious.

5.6.3.4 Expert validation

The results of the semantic profiling were presented to the BTw DMO staff, who confirmed that the semantic structure of the web obtained automatically with the STMT topic modelling tool resembles the actual structure of the website with minor variations.

For example, the BTw DMO staff consider boating to be a very important activity that is very much in demand in their environment. This is also reflected in our results, where specific topics for Sports (Sp), Sea&Sports (SS) and Nature (Na) appeared.

The DMO staff were aware that users from different origins, and thus accessing the site in different languages, are likely to have different interests, and this was confirmed by the results we obtained. In fact, in their opinion repeating the web structure for every access language is not the right decision, since the different interests shown by users of different origin (and therefore using different languages) would require different structures.

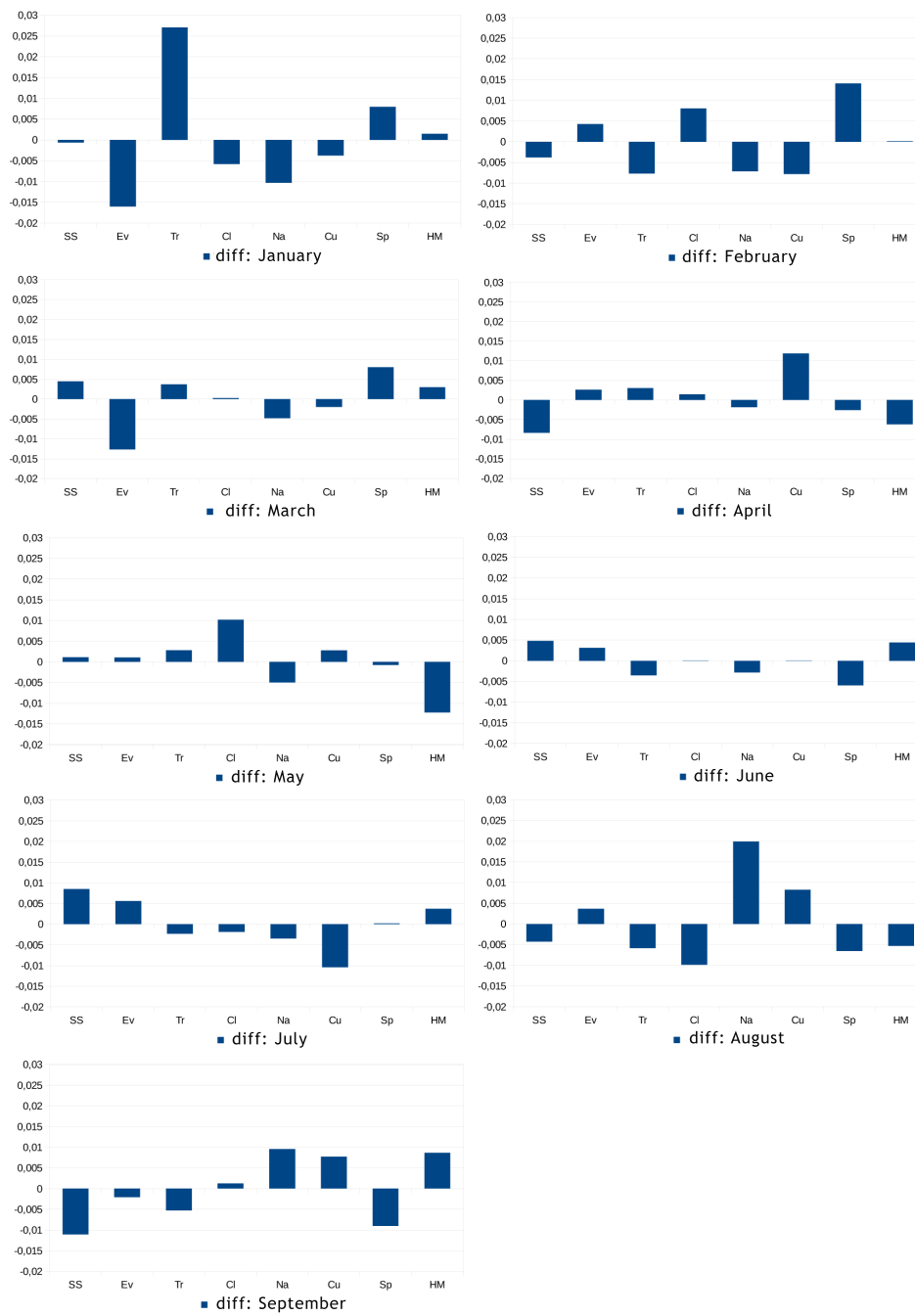


Figure 5.16: Profile differences for the monthly profiles compared with global profiles.

The profiles obtained for different languages are similar to what the experts perceive in the tourist offices. British tourists tend to be very interested in tradition (Tradition Tr). The Spanish tourist generally adheres to what is being promoted by the DMO: Nature and Cuisine (Na and Cu). Access by Basque tourists generally involves local people interested in their roots and also sport activities, which seems consistent with their interest in Historical Monuments (HM) and Sea and Sports (SS and Sp). Finally, the staff affirmed that the French tourist generally comes to "faire la fête". They are very interested in festivals and this is reflected in their specific interest in the Events topic (Ev).

The experts commented that the time period with more foreigner tourist is during Easter and summer holidays. They claimed that these tourists are specially interested in the gastronomy; literally they said "the gastronomy is the highlight of this land". This is reflected in the monthly profiling. During August as the weather is better they commented that people used to go more to the tourist offices asking for plans related to nature, according to experts the tourists used to ask for information about beaches, mountain hiking etc. About the difference between the profiles obtained in July and August, the experts affirmed that during July there is a famous blues festival and they said that "the majority of the tourists specially come for attending to the blues festival" and justify the differences obtained in these months.

In general, the experts validated the results and noted that some of the findings presented were useful for use in future web designs, specific marketing operations, etc.

5.7 Preliminars of a new system

The design and implementation of the global system described in previous sections has shown that the inclusion of semantic information in web usage mining systems enhances the results. Up to this point we have used only statistical methods for processing the content but ontology-based methods include deeper knowledge and could probably further benefit the system. In order to dissipate the doubts, the objective of the new system is to integrate a combination of statistical and ontology-based methods for computing semantically enhanced navigational patterns.

On the other hand, the experience acquired building the global system, made us classify the accessed URLs in 3 groups: the ones accessed very often (mainly sub-index pages), the ones accessed normally and finally, the hardly accessed ones. Therefore, we decided that the most interesting part to model was the one belonging to the normal access and we pre-processed the database accordingly.

As it is commented before, ontologies provide more sophisticated semantic information, in addition, they offer the opportunity to discover more types of relationships among documents and therefore, design accordingly different link proposal strategies. In this section as a preliminary trial link prediction will be selected as link proposal strategy and some options for it will be presented. So, we will add to the usage mining system previously used for link prediction (Section 5.6.1), the combination of statistical and ontology-based methods. Our goal is to compare and evaluate if the combination of statistical and ontology-based methods provide better recommendations for the user than only using usage information and which of these techniques works better from a link prediction point of view; the statistical enrichment or the combination of statistical and ontology-based enrichment. The main schema of this work is show in the Figure 5.17.

In order to implement these objectives, the navigation profiling was carried out following the schema described for previous systems. First of all the usage data was collected and pre-processed. The pre-processing phase was modified to center the analysis in the normal access and filters were added to remove the less frequent URLs and the most frequent ones.

After the data acquisition and pre-processing phases, the pattern discovery and analysis phase was carried out as described in Section 5.5.2.1.

Once the navigation profiles are extracted, the aim is to enrich the system using the combination between statistical and ontology-based system.

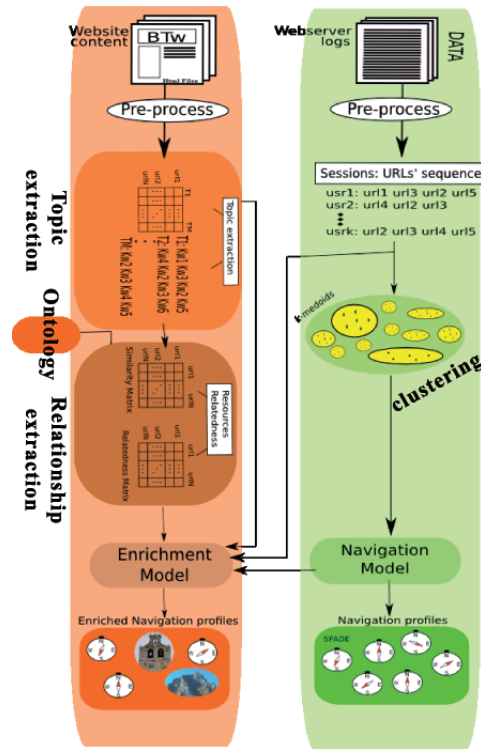


Figure 5.17: Schema of the new system architecture.

5.7.1 Enriched navigation profiling

In this section, some different options to enrich navigation profiles are going to be introduced. On the one hand, the combination of navigation profiles with content information, and on the other hand, the proposal of links based on the semantic of the URLs the users have visited in their navigation.

Before starting to present the different enrichment criteria, the combination methodology of statistical and ontology-based method is going to be introduced. After that, the different enrichment criteria are going to be explained.

5.7.1.1 Combination of a statical method with an ontology-based text comparison system

For the combination of the statistical method and the ontology-based method, we used the topic modelling technique for the former and a system used in Taddesse et al. [80] for the latter.

The first one has been analysed in Section 5.6.1.1.1. Concerning to the ontology-based techniques, we extended the system proposed in Taddesse et al. [80] for extracting the similarity and the relationships of a RSS collection to be used for text comparison.

The main goal of the Taddesse et al. system [80] was to calculate similarities and the relationship between RSSs using an ontology and to identify RSSs talking about the same issue, to merge them in a single one. We adapted the system for comparing URL content.

The process they followed is the one presented below:

Given two texts T_i and T_j , they represent each T_i as a vector V_i in an n -dimensional vector as: $V_i = [(Term_1, w_1), \dots, (Term_n, w_n)]$. The vector space dimensions represent distinct concepts or terms. The vector's length would be the size of the vocabulary of the RSS collection.

The weight w_i associated to a term in V_i is calculated using the semantic similarity value of two concepts in an ontology. If the term appears in both texts the weight of that term in the vector would be 1. In the case that the term or concept does not appear in both, they tried to find similarities between the term and the rest of terms in the other text using the ontology. If the term does not have any relationship with the terms of the other text its value in the vector will be 0.

Using this politic, they constructed the n -dimensional vector for each text. After that, they extracted the semantic similarity values between texts using cosine similarity.

Apart from that, the system created by Taddesse et al. [80] is capable of extracting text relationship as well. They identified *equality*, *intersection*, *inclusion* and *disjoint* relationships. Their method for identifying basic relationships is based on a fuzzy logic model to overcome the often imprecise descriptions of texts. They used the semantic similarity value for extracting the relationship between texts.

They addressed the fuzzy nature of textual content in identifying relationships by providing pre-defined/pre-computed similarity thresholds $T_{equality}$ and $T_{disjointness}$.

Moreover, they used a graph based agglomerative single-link clustering method to extract groups of similar texts. The clustering module puts to-

gether related (similar) items based on the relatedness results provided by the previous phase. Applying such algorithms in their RSS context would result in grouping highly related news in the same cluster. Using this clustering information they extracted a RSS-cluster list (representing the cluster each RSS belongs to).

In order to make possible the application of such method to compare URL contents we combined topic modelling and the system just explained. We collected 30 keywords per topic from the topic-keyword list obtained using the STMT tool (the output of the process explained in Section 5.6.1.1.1) forming a 300 word list (relevant-word list). In topic modelling each topic has a keyword list that is ordered per relevancy being those words the most important ones for each topic, that is the reason why we have assumed that having the 30 first keywords of each topic we are covering the most relevant words of the whole collection of data. As some keywords could be repeated in more than one topic we removed them. We used this list to represent the text of each web page; we removed from the content of each URL the words that do not appear in the relevant-word list.

This process was executed with two different content data. For the first one, we considered that the different parts of the texts should not have the same importance and for the second one, we gave the same importance to every part of the text.

Normally the information of the titles, the headings, the information that is in bold, underlined or in italics should have more relevance than the plain text. This kind of information is supposed to be the most representative in each URL and we decided to give them different importance. As the STMT does not include this option, we implemented it changing the frequency of the words that appear with this kind of tags. As a first approach we doubled the frequency of the information that appears within the previously enumerated tags, generating a weighted database (wDB). That is, we worked with two different databases one giving more relevance to some relevant tags (wDB) and the other one giving the same emphasis; called plain text database (ptDB).

After the summary of the texts, we applied an adaptation of the ontology-based system presented in [80], for comparing URLs. As we have explained the system is able to extract the semantic similarity between texts (URLs in our case), a matrix of relationships obtaining the relationship between URLs, *equal*, *disjoint*, *included in* and *intersection*. Furthermore, the adaptation of Taddesse et al. system is able as well to extract an URL-cluster list which represents the information to which clusters belong each URL to. The system was applied for each database (for wDB and ptDB).

The obtained information was used to design some enrichment strategies as described next.

5.7.1.2 New strategy to enrich navigation profiles based on distances relationship

The designed strategies make use of the URL relationships and the URL similarity matrix. Depending on the strategy a different similarity matrix was used, in the case of topic modelling we used a similarity matrix built using the URL-topic vectors and Hellinger distance (see Section 5.6.2.1) and in the combination case, the similarity matrix obtained from the adaptation of Taddesse et al. system.

The enrichment was carried out in two different ways. On the one hand enriching the navigational profiles (spade-URLs) and on the other hand, using a part of the navigation of the user for proposing some links related to the theme they used (accessed-URLs).

For enriching accessed-URLs the last 4 URLs of the user navigation sequence (in case the navigation is shorter than 4 clicks we adapted the rules of each option for implementing with less URLs) were used. Their relationship was analysed to conclude which type of navigation, as for example focused on one topic or disperse, the user did. Depending on the type of navigation the system proposes some links related with the last part of the navigation or a mix of every theme the user has navigated.

Table 5.11 contains the data extracted applying the described system to the accessed-URLs. Where URL_i represents the last 4 URLs of the user navigation in order, being URL_4 the last URL visited, Cl_i is the cluster URL_i belongs to and Rel_{ij} is the relationship between URL_i and URL_j according to the relationship matrix.

URL_1	URL_2	URL_3	URL_4
Cl_1	Cl_2	Cl_3	Cl_4
	Rel_{12}	Rel_{23}	Rel_{34}

Table 5.11: The navigation information we have.

The relationship can be *equal*, *intersection*, *included in* and *disjoint*. For this work we simplified the relationships to *equal* or *different*. Using the following condition.

```

If  $Rel_{ij}=\text{equal}$  or  $((Rel_{ij}=\text{intersection or included in}) \text{ and } Cl_i=Cl_j)$ 
    then  $Rel_{ij}=\text{equal}$ 
If  $Rel_{ij}=\text{disjoint}$  or  $((Rel_{ij}=\text{intersection or included in}) \text{ and } Cl_i \neq Cl_j)$ 
    then  $Rel_{ij}=\text{different}$ 

```

link proposal

The information obtained in the described process was combined with the previously designed link prediction system or integrated within it. Having different information sources the link proposal can be done within different hypothesis: URL proposals can be done based only on the navigation ($URL_1, URL_2, URL_3, URL_4$) (A) or using the output of the link prediction system based on usage, i.e. the SPADE proposals ($SPURL_1, SPURL_2, SPURL_3, SPURL_4$) (S). Moreover, The proposals can be done composed only by new proposals generated according to the semantic information (1) or the combination of the usage information proposal and the semantic proposal (2).

According to the presented options we implemented four different strategies to make proposals to the user $A1, S1, A2, S2$ summarized in Table 5.12.

We implemented as a first approach the strategy to help the users finding the URLs that they are interested in, i.e. a link prediction system so that the same objective pursued in the global system is pursued. However, having more information about the relationships of visited URLs more diverse strategies can be designed by the DMOs.

The first stage to carry out this strategy is to analyse the navigation of the user and decide whether the user is focused on a theme or she is wandering around and visiting different themes. This analysis was done counting how many *equal* and *different* relationships appear between the last 4 URLs of the navigation of the user. Considering that *equal* relationships mean that the user is focused whereas *different* mean not to be it. Our hypothesis is that when the user is focused on a theme, URLs on that theme will interest her whereas when she navigates wandering around different themes she will more likely be interested in different themes. Our strategy proposes links accordingly.

	accessed-urls	spade-URLs
only semantic similarity	A1	S1
semantic + usage	A2	S2

Table 5.12: Four different options for proposals

	accessed-urls
only semantic similarity	<p>A1</p> <p>If $\#(Rel_{ij}=\text{equal}) \geq \#(Rel_{ij}=\text{different})$ propose: two nearest URLs to URL4 two nearest URLs to URL3</p> <p>If $\#(Rel_{ij}=\text{equal}) < \#(Rel_{ij}=\text{different})$ propose: nearest URL to URL4 nearest URL to URL3 nearest URL to URL2 nearest URL to URL1</p>
	<p>A2</p> <p>propose: SPURL4 SPURL3</p> <p>If $\#(Rel_{ij}=\text{equal}) \geq \#(Rel_{ij}=\text{different})$ propose: nearest URL to URL4 nearest URL to URL3</p> <p>If $\#(Rel_{ij}=\text{equal}) < \#(Rel_{ij}=\text{different})$ propose: nearest URL to URL2 nearest URL to URL1</p>
	spade-URLs
only semantic similarity	<p>S1</p> <p>If $\#(Rel_{ij}=\text{equal}) \geq \#(Rel_{ij}=\text{different})$ propose: two nearest URLs to SPURL4 two nearest URLs to SPURL3</p> <p>If $\#(Rel_{ij}=\text{equal}) < \#(Rel_{ij}=\text{different})$ propose: nearest URL to SPURL4 nearest URL to SPURL3 nearest URL to SPURL2 nearest URL to SPURL1</p>
	<p>S2</p> <p>propose: SPURL4 SPURL3</p> <p>If $\#(Rel_{ij}=\text{equal}) \geq \#(Rel_{ij}=\text{different})$ propose: nearest URL to SPURL4 nearest URL to SPURL3</p> <p>If $\#(Rel_{ij}=\text{equal}) < \#(Rel_{ij}=\text{different})$ propose: nearest URL to SPURL2 nearest URL to SPURL1</p>

Table 5.13: The rules of the four different options for proposals

Table 5.13 shows the rules we implemented: where Rel_{ij} is the relationship between URL_i and URL_j according to the relationship matrix and

$\#(Rel_{ij} = equal)$ is the number of times that the relationship is *equal*.

The nearest URL is obtained selecting the URL with highest value in the similarity matrix.

In order to evaluate the improvement provided to the system by the introduction of an ontology-based method within the content analysis, we compared the achieved performance to the performance achieved with the usage based system and to the performance achieved with a semantically enriched system but based only on the statistical method.

In the case of enriching the usage proposals using only the information of the statistical method, we implemented the same 4 recommendation options explained before. As in this case we do not have relationship information (due to the fact that topic modelling does not provide it), we considered that two URLs are *equal* when they are catalogued to be of the same topic and *different* if their topic is not the same.

```
If Topic{URLi}=Topic{URLj} Relij=equal
If Topic{URLi}!=Topic{URLj} Relij=different
```

The 4 recommendation options explained before (*A1*, *A2*, *S1*, *S2*) can be directly applied using the statistical method. The relationship matrix needs to be calculated with the assumption previously explained and the similarity matrix is the one calculated in Section 5.6.2.1.

5.7.2 Experiments: results and analysis

5.7.2.1 Experimental setup

In order to evaluate the performance of the combined system, we applied the holdout method dividing the database into three parts. One for generating the clusters and extracting navigation profiles, the other one for validating the system and to fix the parameters and the last one for testing or using it in exploitation. To simulate a real situation we based the division of the database on temporal criteria as we did before: we used the oldest examples (70% of the database) for training, the next division (20%) for validation and the latest ones (10%) for testing.

Although navigation profiling was carried out using log information from January, 2012 until November, 2012, containing initially 3,850,086 requests, due to the new pre-process described in Section 5.7 the database was reduced to 107,394 request with 10,792 sessions.

For fixing the maximum number of clusters for *PAM* clustering algorithm, we explored a range of values for the parameter *K* (10, 20, 40, 60, 90,

100, 150, 200, 250 and 350) because the size of the database after the new pre-processing filters is reduced to the half, we reduced to the half as well the values of K for the analysis in comparison with previous system. Using the validation sample we selected as the best K value 150. The minimum support parameter for SPADE of 0.2, was also fixed based on our experience. As a result, we built a link prediction system applying the PAM clustering algorithm to the training data using as K value 150 and the SPADE algorithm with the minimum support of 0.2.

We validated the system in three different situation: when no content information is used (SP), when we enriched the web usage mining system with semantic information using only statistical methods (TM) and using the combination of statistical methods and the ontology-based method (oTM).

We evaluated the system using the test sample as described in Section 5.5.3.

We simulated the navigation of new users using 50% of the user navigation sequence in the validation and test examples for making decisions. That is to analyse the relationship information between URLs or to select the nearest cluster or profile. The decision of changing from 25% to 50% was conditioned by the new strategy; if the analysed navigation sequence is too short there are not enough relationships between URLs to decide if the user is focused or not. As we described in Section 5.7.1.1 we have used the last 4 URLs within the 50% of the user navigation sequence for calculating the type of navigation.

We computed statistics based on the results for each one of the new users. We calculated precision, recall and F-measure as we did in previous system to evaluate each option. We calculated the values from the link prediction point of view i.e., using only the clicks in the test sequence that have not been used to select the nearest profile.

5.7.2.2 Results and Analysis

As explained in Section 5.7 we decided to focus this system in the normally accessed URLs. An analysis of the most frequently accessed URLs showed that, the most frequent ones were the 5 main sub-index pages. So, we decided to propose them statically putting them always available for the user. Consequently, the performance of the link prediction system will be now the combination of: the results of the static proposals, i.e. the results the system would obtain proposing always the 5 sub-index pages, and the results of the dynamic proposals, i.e. the proposals based on link prediction. So, to extract the real results of the global system, the static proposal results

and the dynamic proposal results were added.

		Pr				Re				Fm				
Static proposal		30,35				32,09				30,52				
Validation dataset	Dynamic proposal	oTM		TM		oTM		TM		oTM		TM		
		ptDB	wDB	ptDB	wDB	ptDB	wDB	ptDB	wDB	ptDB	wDB	ptDB	wDB	
		A1	11,12	11,04	7,16	6,43	10,15	10,16	7,01	5,76	10,44	10,35	7,16	5,90
		A2	13,46	13,42	11,81	11,47	12,69	12,76	11,20	10,83	12,82	12,84	11,27	10,92
		S1	8,27	7,86	7,68	5,39	7,47	7,02	7,01	4,92	12,82	7,21	7,16	5,02
	S2	12,05	12,17	12,13	11,11	11,36	11,38	11,51	10,53	7,68	11,52	11,57	10,60	
SP		11,12				9,33				9,80				
Test dataset	Dynamic Proposal	oTM		TM		oTM		TM		oTM		TM		
		ptDB	wDB	ptDB	wDB	ptDB	wDB	ptDB	wDB	ptDB	wDB	ptDB	wDB	
		A1	13,02	12,95	7,57	7,25	12,14	12,09	6,99	6,57	12,33	12,26	7,07	6,72
		A2	15,04	14,82	13,52	13,11	14,63	14,46	13,15	12,79	14,57	14,39	13,10	12,74
		S1	8,12	7,57	6,57	5,66	7,41	6,88	6,04	5,18	7,60	7,08	6,17	5,29
	S2	12,82	12,75	12,52	12,22	12,32	12,27	12,15	11,93	12,35	12,30	12,12	11,86	
SP		11,99				10,59				10,96				

Table 5.14: Results of static and dynamic proposals

The first row of Table 5.14 shows the result the system would obtain proposing only statically the 5 more frequent sub-index links. Moreover Table 5.14 shows the results of oTM and TM systems for the plain text database (ptDB) and for the weighted database (wDB) respectively for validation (upper part) and test samples (lower part). The table contains the results of the enriched system for every recommendation option and the usage system proposals (SP) for both datasets.

Analysing test results, it showed that the use of the semantic knowledge extracted from the website content information improves the performance, precision, recall and F-measure values, of the usage system proposed. Using most recommendation options (A1, A2, S2) for oTM and (A2, S2) for TM we obtained better values than in SP.

From the methodology point of view, comparing the values obtained with the statistical method (TM) and the ones obtained with the combined method (oTM), the best one is oTM. It means that paying attention not only to the exact matching of the words but to the similarity between them using an ontology improves the performance of the content mining system and thus improves the performance of the complete link prediction system.

Regarding to the 4 recommendation options proposed A1, A2, S1 and S2 the best one according to the validation results is A2. We obtained the highest value achieving 43,34% (calculated adding 30.52 of the static proposal + 12.82 of dynamic proposal) of F-Measure value for oTM and 40,79 % for TM in the case of the ptDB. Moreover, for the wBD the F-measure values are 43,36% for oTM and 41,44 % for TM. Focusing in F-measure the values obtained with the wDB in the case of the oTM are slightly better than the ones obtained for ptDB. Although the difference between

the results of both databases is small, we consider that the relevance of the text within tags is important and, as a consequence, some reflection should be done about how to improve the wDB in the future.

According to the test results, the system achieved the highest values with A2 option for the combination of ontology-based plus statistical method (oTM) and the plain text database (plDB). It reached 45,39% precision, 46,72% recall and 45,09% F-Measure.

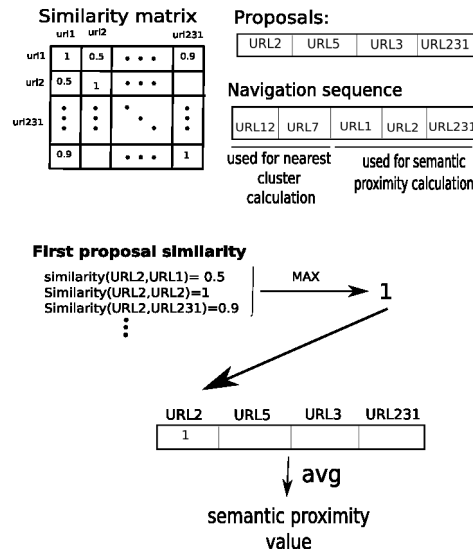


Figure 5.18: The process of the semantic proximity calculation.

We also implemented a non conventional performance metric to calculate how good the recommendations are from a semantic point of view (called semantic proximity). Figure 5.18 shows the calculation process of the semantic proximity. First of all we extracted from the similarity matrix the similarity value of each proposal with every click in the test sequence that was not used to select the nearest profile. In the example each of the proposals, *URL2*, *URL3*, *URL5* and *URL231* are compared to the URLs not used for nearest cluster calculation, *URL1*, *URL2* and *URL231*. The lower part of the figure describes the comparison process of the first proposal, *URL2*. *URL2* is compared to *URL1*, *URL2* and *URL231* and the maximum similarity value within them is chosen. This process is repeated for each of the proposals. Then, the average of the value extracted for each proposal is calculated. The outcome of this process will give us the chance to now

how similar our proposals are to the real navigation. Calculating the semantic proximity for every sequence in the test sample, we obtained a value of 65,47%.

5.7.3 Summary

The systems built using the standard web usage information and the website content are able to automatically extract the semantic structure of the site and combine both information sources to extract knowledge for many applications.

User navigation profiles obtained from usage information will be useful for link prediction. These profiles can be enriched with different URLs using semantic information, with the result that the set of proposed links will be diversified. This could have a direct application for the DMOs; they can introduce links that suit the taste of the users, according to their interests.

The user navigation profiles obtained were stable; the global system obtained good quality profiles that match in more than 60% of cases the real user navigation sequences. Since the previous evaluation was too rigid we evaluated the link prediction capacity of our system according to user interests instead of specific URLs. The precision and recall values soar up to 90.9% and 73%, which means that the generated navigation profiles were accurate in terms of interests, which will make the browsing experience of the new user more pleasant and faster. Moreover, link proposals diversified using semantic information obtained similar values for precision and recall.

The system improves the results using content information for enriching usage link prediction. Even if different methods were applied such as, search engines, keyword extractors and topic modelling as statistical methods and also an ontology-based method, in the system focused in the normally accessed URLs the system enriched with the ontology-based method shown to perform the best obtaining a precision over 45% in the link prediction context and a 35% of improvement in comparison with the usage based link prediction system.

The BTw DMO staff corroborated that the automatically extracted semantic website structure captured the ideas behind the website. Moreover, some of the information provided by the interest profiles obtained by combining content and usage information was also validated by the BTw DMO staff. They also found the knowledge provided to be very useful for future web design and marketing campaigns.

Chapter 6

Combination of web structure, content and usage mining for improving *discapnet* navigation

We used our previous experience to move into the area of disabled people. We concretely worked in *discapnet* website where added to implementing a link prediction system, we combined web usage, structure and content information to build a system able to detect navigation problems in the site.

6.1 Introduction

Nowadays, digital skills are considered fundamental. Therefore, it is important to familiarize people in general, and specifically people with disabilities and/or elderly people, with digital devices and applications and concretely to adapt websites to enable their use by these users. Within this context, website access is an important tool for information-seeking, communication and participation processes in our society.

Unfortunately, a theoretically accessible design might not be sufficient to enable efficient website access for people with disabilities. This makes the analysis of the interaction of users with a website crucial to assessing their behaviour, detecting possible problems and providing solutions.

In general web applications, it is very easy to fail to recognize the full range of users who might be interested in using or who might need to use

the application [171]. However, in the case of *discapnet* (www.discapnet.es financed by Technosite and the ONCE foundation), a website mainly aimed at visually impaired people, it is known that, at least, visually impaired users will be accessing it.

For visually impaired people the use of the audio web interfaces is essential, these tools are their eyes when they are navigating in the Internet. However, navigating through audio web interfaces is a challenging task mainly because content is rendered in serial. The sequential access of screen readers means that visually impaired users take up to five times longer than sighted users to explore a web page [172]. Besides, the screen reader itself requires an additional cognitive effort [173]. The most important negative implication of content serialization is that users cannot obtain an overview of the page, meaning that users can only obtain the information rendered sequentially by the screen reader as they scan through the document. Consequently, navigation across different web pages is a time-consuming task, and web page exploration is a resource-intensive activity that requires dedicated attention.

As a consequence, in the case of users with disabilities, system evaluation and problem detection become crucial to enhancing user experience and may speed up the navigation and contribute greatly to diminishing the existing technological gap. System evaluations can be conducted by different actors in different moments and in different manners; moreover, they can achieve different objectives [174]. Automated checking, evaluations conducted by experts and evaluations using models and simulations are very valuable when initial prototypes are available. In contrast, evaluations by users are required in at least the final stage of the development. However, this is difficult to perform because finding samples of disabled and elderly people willing and able to take part in evaluations is not easy. Another option is to collect in-use information while the user is accessing the Web, thereby building a non-invasive system capable of modelling the users in the wild.

As in any web environment, in *discapnet*, the contribution of the knowledge extracted from the information acquired from in-use observation, as it is explained in the previous chapter, can be used for web personalization and for extracting knowledge of a very diverse nature, such as the problems the users have while navigating, interests of the people browsing the website, or possible design mistakes, among others.

When the user is a person with physical, sensory or cognitive restrictions, data mining is the easiest and often the only way to obtain information about the uses of the website by the person.

The work presented in this chapter proposes and describes two non-invasive systems that analyse, in a completely anonymous manner, the interaction of users with *discapnet* website. The first one builds user navigation profiles that provide a tool to adapt the web (through link prediction) using the experience of previous systems. Being *discapnet* website addressed to people with disabilities, mainly to visually impaired people, link prediction will be specially important. This is corroborated somehow because a preliminary analysis of the web logs showed that the time spent in link type or hub type pages is considerably longer than it would be expected to; it is longer than the one spent in pages devoted to content (content pages) and dynamic pages which are mainly related to news. This makes us suspect that the implementation of an efficient link prediction system will definitely help to make navigation easier, and as a consequence, diminish the time spent in link type pages.

The second system is aimed at problem discovery in *discapnet* navigation and it can also be used for detecting problems when new users are navigating the site. The outcome of this second system, provides a very valuable knowledge about the difficulties the users are having while navigating *discapnet* and will be very useful not only for helping the users that are navigating through the site but also as a guide for future improvement of the website. Hence, it will contribute to improve the users' experience while navigating the *discapnet* website.

In addition to the benefits to the users, the system will also be very useful for the service providers because, as stated by Pucillo and Cascini [175], the most unhappy customers (in this case, those having the most problems with the website) could be your greatest source of learning.

6.2 Related work

Usability and accessibility are broadly used terms. Usability refers to the extent to which a product can be used by specific users to achieve specific goals with effectiveness, efficiency and satisfaction in a specific context of use [174].

On the other hand, accessibility usually refers to the use of e-Systems by people with special needs, particularly those with disabilities and elderly people. ISO 9241-171 (2008) defines accessibility as the usability of a product, service, environment or facility by people with the widest range of capabilities. This fits within the universal design or design-for-all philosophy [174].

The Web Accessibility Initiative (WAI), founded by the World Wide Web Consortium (W3C) to promote the accessibility of the Web, defines web accessibility as meaning that people with disabilities can use the Web. More specifically, web accessibility means that people with disabilities can perceive, understand, navigate, and interact with the Web [10].

It is now acknowledged that more than just usability is needed in the design and evaluation of e-Systems, and there has been an apparent shift in research on human/computer interaction from cognitive-task performance to user experience [176].

The automated checking of conformance to guidelines and standards can anticipate and explain many potential usability and accessibility problems and can be performed before a working system is available. However, the evaluation of detailed characteristics alone can never be sufficient because this does not provide sufficient information to accurately predict the eventual user behaviour. Therefore the study of the user interaction with such systems becomes important.

In the context of the interaction of users with disabilities with web systems, people with visual disabilities have been the focus of most of the studies. These studies mainly underline the fact that little is known about the navigation tactics employed by screen reader users when they face problematic situations on the Web. Modelling user navigation therefore becomes of the utmost importance because it allows us to not only predict interactive behaviour but also to assess the appropriateness of the text of a link, structure of a site and design of a web page [177].

Although predefined models can be used in the design processes, models built based on in-use information, where behaviours emerge from the obtained data, will provide more realistic information about the usage characteristics of the site. Web logs are the most simple in-use techniques and are thus applicable to a wider range of users.

As previously mentioned, a part of this work focuses on automatically discovering problematic navigation profiles to be used to improve the site and, as a consequence, the browsing experience of the user, thus making the website easier to use and more convenient. We consider that web mining techniques are adequate to achieve our objectives and that the combination of the knowledge extracted from web usage information with knowledge extracted from the content and structure of the site makes it possible to detect not only human problems but also content or structure-related problems.

The combination of usage and content information for knowledge extraction has shown to be effective outside of the area of users with disabilities such as the tourism area.

Based on the acquired experience, this work proposes two systems that combine usage, content and structure information from *discapnet* website (where many of the users are disabled). The first one builds user navigation profiles that provide a tool to adapt the web (through link prediction) and the second one is able to automatically detect navigation problems the users are having, which can contribute to improving the structure and automatically adapting the site to new users. The systems will therefore improve the quality of interaction on the site, which will certainly contribute to improving the user experience. As a result, although the main focus in this work is not to measure or improve the user experience, we believe that its outcome and use will contribute to improvements in the user experience.

6.3 *discapnet* website

discapnet is an initiative created to promote the social and work integration of people with disabilities and is financed jointly by the ONCE [178] foundation and Technosite. It contains two main lines of action:

- An information service for organizations, professionals, people with disabilities and their families.
- A platform to develop actions to promote the involvement of people with disabilities in economic, social and cultural life.

Technosite provided us with the logs of two servers that store the activity taking place in some areas of the *discapnet* website. The transferred data were anonymized server log data.

In addition to log data, we used information extracted from the structure and content of the site. Figure 6.1 shows the appearance of the front page of *discapnet*.

The site is divided into different areas, the main ones being *Areas Temáticas*, *Comunidad* and *Actualidad*. The treatment given to all these areas was not the same. Some of them, such as *Actualidad* (current affairs) and *Noticias* (news), are very dynamic and thus require specific approaches that are not within the scope of this work. Due to their dynamic nature, no content analysis was performed within them, and they were considered as a special type (we will refer to pages within this group as *other*-type pages). Index pages were also considered as a special type (of type *index*-type pages) because the number of times they are accessed can be useful for detecting problems. Moreover, these sections can hardly be used for link prediction because it is impossible to build the models according to news that will be



Figure 6.1: Appearance of the front page of the *discapnet* website.

generated in the future. From the rest of the zones in the website, the experts in Technosite considered that *Áreas Temáticas* (excluding *Salud*) and *Canal Senior* within *Comunidad* were the most interesting zones for modelling and introducing adaptation tools. And, as a consequence the provided data was limited to these zones. Therefore, the built user models and link prediction system will be mainly focused to *Áreas Temáticas* (see Figure 6.2).

Before starting to explain each of the systems, we carried out an analysis of the site: we evaluated the accessibility of each of the pages of *discapnet* and performed a simple structural analysis. We found accessibility to be an important starting point because, although it does not ensure facility of use, we considered it a requirement in this context. Therefore, the accessibility was evaluated using EvalAccess [179], the Automatic Accessibility Evaluator developed by EGOKITUZ (Human computer interaction laboratory for people with special requirements in the University of the Basque Country UPV/EHU) according to the design guidelines published by WAI [10] and devoted to helping designers to produce websites that are accessible. The study showed that the accessibility rate was approximately 90% on average, which means that each individual page in *discapnet* was designed considering the accessibility guidelines.

However, the structural analysis showed that in *discapnet*, the average number of links per page is 27, and the average depth of the *content*-type pages is 4. These values will affect to artifact complexity and task complexity [176] and, as a consequence, to task performance what is not independent

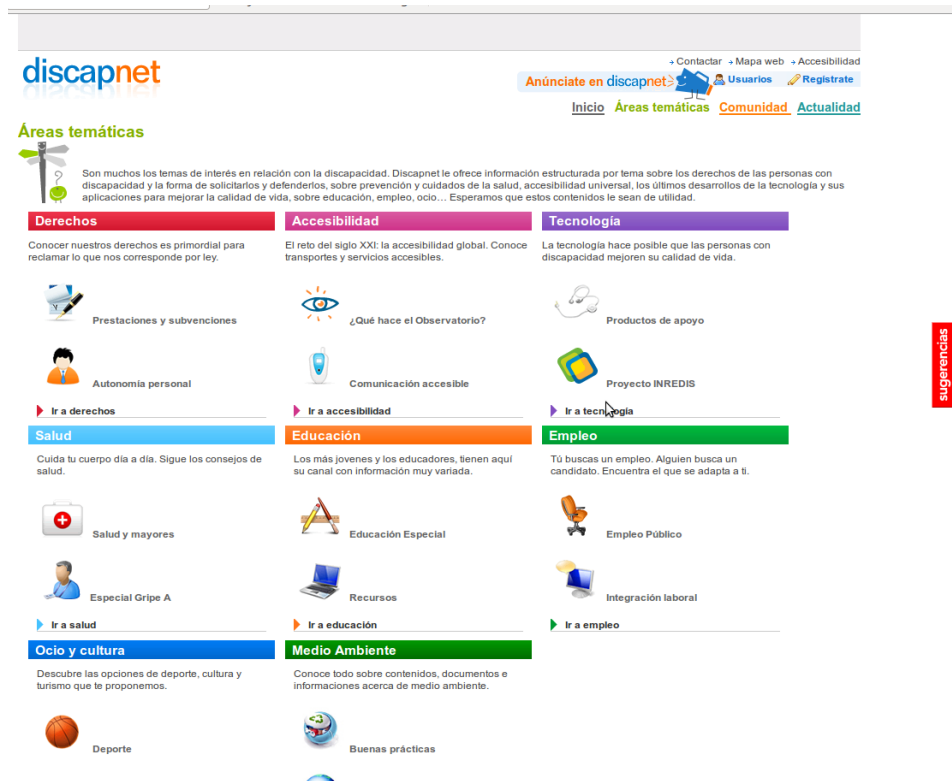


Figure 6.2: Appearance of *Áreas temáticas* within *discapnet* website.

from user experience.

According to Standen et al. [176], one source of artifact complexity is the complexity of the navigation structure of a website, which can be reflected in the number of options that are available for selection on each page (low artifact complexity is considered when the number of links ≤ 10 , and high artifact complexity ≥ 10). The greater the number of links per page, the more difficult the task of finding information will become, negatively affecting the balance of challenge and skill.

Task complexity reflects the path from the starting point (the home page of a website) to the destination (low complexity - minimum number of clicks from the index is 2, and high complexity - minimum number of clicks from the index is 4). The task performance will decrease with the path length. In addition, as the task complexity increases, the balance between challenge

and skill (i.e., performance) will be adversely affected.

Consequently, according to Standen et al. [176], in *discapnet*, the artifact complexity is very high, and the task complexity also tends to be high, which makes task achievement difficult, what makes any possible improvement very important.

6.4 Link prediction system

The aim of this system is to design a link prediction system similar to the one built for *bidasoa turismo* which contributes to make the navigation of users navigating in *discapnet* easier. As we did in the tourism environment, the proposed system is based on observation in-use; behaviours emerge applying a web mining process to the obtained data; web server log data. Due to the characteristics of the site, we developed two approaches for user profiling: a global approach built based on the complete website and a modular approach carried out discovering the navigation profiles within each zone. We considered that the inclusion of the described system in *discapnet* will contribute to improve navigation within the website and diminish the times spent in link type pages.

The phases followed to build the system were exactly the same ones used to build the link prediction system for *bisadoa turismo* and will be shortly described in the next section.

6.4.1 Data acquisition and pre-processing

The usage data used to build the link prediction system, contained all the requests processed by two of the servers hosting the *discapnet* website from the 2nd February, 2012 to the 31st December, 2012.

The pre-process was carried out following the steps described in Section 5.5.1. The log files used in this system contained 157,527,312 requests, which were reduced to 13,352,801 after removing the erroneous requests and the requests generated automatically by the server to compose the desired page.

The 13,352,801 requests were divided in 907,404 sessions after the sessioning stage. Finally we removed the outliers, obtaining a final database which contains 241,311 user sessions. The evolution of the database is summarized in Table 6.1.

Process	Requests	Sessions
In log files	157,527,312	
Valid After sessioning	13,352,801	907,404
After removing outliers	2,828,248	241,311

Table 6.1: Evolution of the database.

6.4.2 Pattern discovery and analysis

This section, taking as input the user click sequences, is in charge of modelling users and producing the navigational profiles.

6.4.2.1 User navigation profile discovery

We have used the process described in Section 5.5.2.1 for extracting user navigation profiles, the combination of PAM and SPADE. We fixed the value for the minimum support to 0.2 based on previous experience and limited the number of proposed URLs to 3 because proposing too many could disturb the user specially in this context where they might have some kind of disability.

In order to carry out the evaluation of the profiles, we used the holdout method based on the experience acquired in *bidasoa turismo* system where similar values were obtained using 10-fold cross-validation and holdout, for evaluation. We divided the database into a training set (70% of the examples), a validation set (20% of the examples) and a test set (10% of the examples). As in the previous chapter we used the validation set to select K and the test set to evaluate the performance of the system.

The same process was applied for the global approach and for the modular approach.

6.4.2.2 Global approach

The structure of each of the subtopics within *Areas Temáticas* is very different and this will probably affect to the navigation the users do within them. Moreover a preliminary analysis of the sequences showed that nearly 50% user navigations belonged to navigations in a single zone. We considered those sessions representative of the navigation within each zone and decided to build the global link prediction approach based on them (48,060 user sessions). The global approach consists on applying the clustering and profiling to the new database.

Figure 6.3 shows the process followed by the global approach. The patterns were grouped using PAM clustering algorithm and the profile for each of the clusters was discovered based on SPADE.

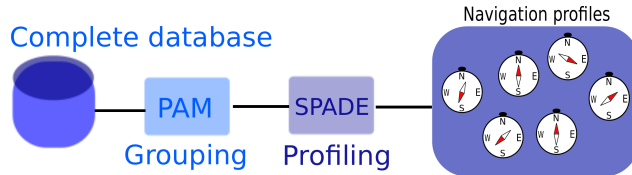


Figure 6.3: Global approach to user profile discovery.

The number of clusters for PAM must be selected. Based on the usual exploration limit for the number of clusters, \sqrt{n} , in this case, for the global approach we tried the following 5 values: $\sqrt{n}/4$, $\sqrt{n}/3$, $\sqrt{n}/2$, $2 * \sqrt{n}/3$ and \sqrt{n} . Using the validation sample, this value was fixed to 130.

6.4.2.3 Modular approach

Being the structure of each zone within *Areas Temáticas* (explained in section 6.3) different, we decided to build a modular approach to the user navigation profiling within *discapnet*. This means to build the profiles focusing on each of the possible analysis zones for user navigation profile discovery. With this aim, instead of working with the whole database, we worked with the user sessions located in a single web zone. That is we divided the database according to navigation zones and we worked with 8 different subsets; one for each of the zones where user navigation profile discovery will be carried out. Table 6.2 summarizes the sizes of each subset and Figure 6.4 shows the schema of the system where it can be observed that the profile discovery process within each zone was carried out as described in Section 6.4.2.1.

The set of profiles in the modular approach will be composed by the set of profiles generated for each one of the 8 zones.

Obviously, being the sizes of the subsets very different, the selection of the number of clusters or user profiles generated in each of them was different; Table 6.2 shows in column *K* the number of profiles generated in each of the modules according to the evaluation carried out using the validation set. This number was selected using the validation set to evaluate results in the same way that the test set was used to evaluate the final system. The procedure is explained in the following lines.

Website zone	User Sessions	Average length	K
<i>Accesibilidad</i>	10,259	5.08	90
<i>Derechos</i>	22,561	4.78	80
<i>Educación</i>	1,773	4.27	27
<i>Empleo</i>	4,720	3.89	60
<i>Medioambiente</i>	852	5.05	20
<i>Ocio y cultura</i>	3,603	4.86	50
<i>Tecnología</i>	3,954	4.54	50
<i>canal senior</i>	338	4.6	13

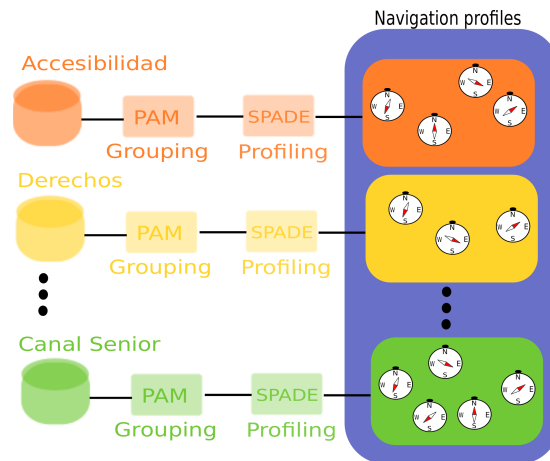
Table 6.2: Size of each zone within *Areas Temáticas*

Figure 6.4: Modular approach to user profile discovery.

6.4.3 Exploitation

The generated profiles were evaluated by comparing them to new users navigating the website (test set). As a result, the system will propose to the new user the set of links that models the user's nearest clusters.

We simulated this real situation using 25% of the test sequences to select the profile for the new user according to the built model (between 1 and 2 links, because, as it is shown in Table 6.2, the click sequences have in average near 5 links).

Based on previous experience, we used 2- NN to select the nearest clusters and combined the profiles of the two nearest clusters with defined pro-

files, weighting URL selection probabilities according to their distance. We combined these to propose profiles containing at most 3 URLs; those with the highest support values. If there are not enough URLs exceeding the minimum support value the profiles could have less than 3 URLs.

We computed performance metrics based on the results obtained for each of the new users of the test set. We calculated precision, recall and $F\text{-measure}_{0.5}$. We consider that in the concrete environment we are working it is really important to propose links that the user finds interesting because other proposed links would probably disturb the user. As a consequence, it is more important for the proposed links to be adequate according to user interests (precision) than guessing more of the used links (recall). This is why we used $F\text{-measure}_{0.5}$.

Table 6.3 shows the average results (precision, recall and $F\text{-measure}_{0.5}$) obtained for the test and validation sets in both cases: with the global approach, and the modular approach. The numbers show that the modular approach achieves better results than the global one, obtaining improvements of around 9% in recall and around 6.5% in precision and $F\text{-measure}_{0.5}$. Furthermore, results are similar for both, the validation set and the test set what means that the concrete data used to evaluate the system does not severely affect to the obtained performance.

	k	Validation			Test		
		Pr	Re	F05	Pr	Re	F05
global approach	130.00	0.55	0.40	0.51	0.55	0.40	0.51
modular approach	48.75	0.58	0.44	0.54	0.58	0.44	0.54
% improvement		5.65	9.06	6.10	6.49	8.97	6.70

Table 6.3: Summary of the results.

The values obtained for the modular approach show that if we would use the profiles for link prediction, nearly 60% (precision=0.58) of the proposed links (tending to 2 out of 3) would be used by the new user. This could make the user navigation easier. Moreover, taking into account that the preliminary analysis showed that the time spent in hub pages is longer than usual we could assert that using those profiles for link prediction would save a big part of the time spent by users in their navigations.

The designed system seems to obtain near balanced values for precision and recall. Therefore analysing the recall we could state that nearly 45% of the links used by the new users would be among the ones proposed by the system (recall=0.44).

6.5 Problem detection system

The main objective of the second system is to automatically detect navigation problems that the users are having while using the *discapnet* website. The proposed system as the one before is divided into data acquisition and pre-processing, pattern discovery and analysis and exploitation phase.

The architecture of this second system is showed in the Figure 6.5

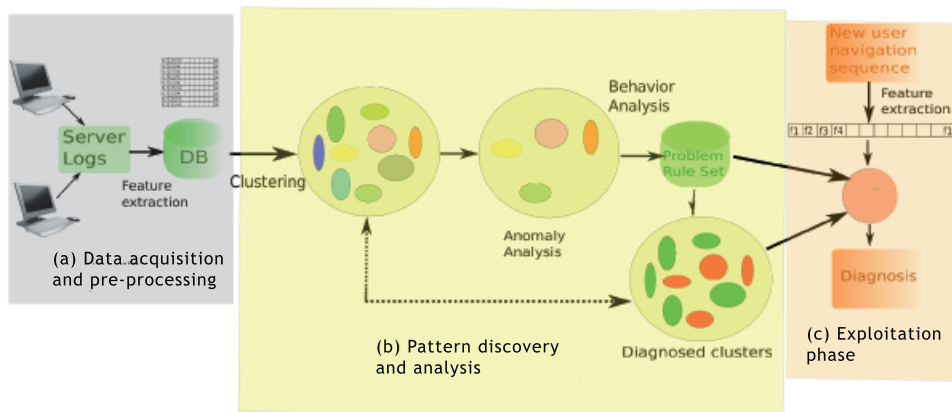


Figure 6.5: Architecture of the problem detection system

6.5.1 Data acquisition and pre-processing

This phase was already carried out to build the link prediction system (Section 6.4.1) which obtained at the end a 241,311 sessions database. However, the sequential representation of the data is not adequate for problem detection and a feature extraction phase was required.

6.5.1.1 Feature extraction

Once the database to be used for the process was selected via the filtering and pre-processing stages, we needed to decide how to represent the information to be used in the machine learning algorithms. We extracted a set of features from the user sessions that might be indicators of user problems and combined them in a vector representation.

The selection of the features used for the process was a difficult task due to the fact that we have to supply the machine learning algorithms with

sufficient and adequate information to be able to automatically detect users with navigation problems.

We expected to find different types of difficulties, and the ideal would be to be able to extract features to detect all of them. The difficulties in navigation could be related to physical problems (for instance if the person is visually impaired or if she has any cognitive problem, etc.), the structure of the site, the structure of the pages and so on. Situations associated with problems in one web page might be not problematic in another; for example, spending a long time in a *content*-type page with a large amount of text might seem normal, whereas spending a long time in a *link*-type page, where the main information is to link to other pages, indicates a problem. Other problems are difficulties finding the required information or difficulties navigating through the structure, which could appear in the form of, for example, frequent changes in navigation zone (calculated using usage + content information) or reaching pages through external links (extracted using usage + structure information). Therefore, to be able to extract meaningful features to detect these problems in *discapnet*, we need to collect information on the nature of the web pages using usage, content and structure information, which has been obtained through a previous analysis of the site and its pages as follows:

- We defined an index value *LCIndex* (Link-Content index) to determine if the main aim of a web page is to provide links to contents or to other parts of the site (*link*-type pages) or if the page is devoted to the content (*content*-type pages). We defined this index as the ratio between the amount of text and the number of links.

$LCIndex = \frac{(Nwords - NwordsLinks)}{Nlinks}$ where *Nwords* represents the number of words in the page, *NwordsLinks* represents the number of words in the links appearing within the page, and finally, *Nlinks* represents the number of links appearing within the page. According to this index, the type of the page is determined as

$$Type = \begin{cases} link & LCIndex < LCIThreshold \\ content & LCIndex \geq LCIThreshold \end{cases} \quad (6.1)$$

A range of values were used to tune the *LCIThreshold* parameter for the *discapnet* website. We analysed the characteristics of *discapnet* website, the text/link ratios (*LCIndex* values) of some sample pages and the rank of values obtained for all the pages in the website and

determined 10 to be the adequate value for $LCIThreshold$ parameter for *discapnet* website.

All the URLs studied of the website were analysed to determine whether they were of *link*- or of *content*-type.

- A content analysis of the website was performed with the aim of assigning a topic to every URL. To obtain a topic structure of the site, based on previous experience, we applied a topic modelling process [180, 181] based on the Stanford Topic Modelling Toolbox (STMT)[169] to the set of URLs. Unfortunately, we discovered that the outcome was no more informative than the page path given by the URL name; the information in the paths is as semantically meaningful as the information we were able to obtain through analysing the texts. As a consequence, we could consider the website to have been previously annotated by the page paths, which were used as the semantic information of the website; we specifically used depth 3 of the page path to determine the topic. We have selected 9 different topics.
- The structure of the site was also used to obtain some features. An adjacency matrix (*Adj.Mat.*) was built to determine if direct links exist between every pair of URLs on the site.

The three points explained above are combined in order to extract informative features that would provide an insight into physical access problems, the intensive use of index pages, looseness and the use of the structure of the site.

As it has been said all the presented features are extracted from the sequences of navigation of the users (usage data), and depending on the case, they also use content and structure information (topics, adjacency matrix and so on).

Features are extracted for each user session composed of L URLs where L is the sequence length. The characteristics of the k -th URL in the session will be indicated by the following features:

- $U_k^{Top,Type}$ where $Top \in (topic_1, topic_2...)$ and $Type \in (link(L), content(C), other(O), index(I))$.

$$U_k^{Top,Type} = \begin{cases} 1 & \text{when the URL is of the topic} \\ & \text{and type given in } Top \text{ and } Type \\ 0 & \text{otherwise} \end{cases} \quad (6.2)$$

For each URL we will have two vectors: $(U_k^{1,L}, U_k^{1,C}, U_k^{1,O}, U_k^{1,I})$ a binary one, which contains the type of URL using previous equation, and an integer one $(t_{U_k})_1^L$ where t_{U_k} indicates the time spent in the k th URL of the session.

According to these features, the total time of a user session can be defined as $T_s = \sum_{k=1}^L U_k^{-,-} * t_{U_k}$ where $-,-$ means any value for *Top* and *Type*. Giving to *Top* and *Type* every possible value they can take, the time spent in every URL in the sequence will be accumulated in variable T_s . On the contrary, giving concrete values for example to *Type*, will make possible to calculate the time spent by the user in a concrete type of URLs.

Table 6.4 summarizes the main features extracted for each user session. We extracted a set of 14 features per user session to analyse the time spent in different URL types, types of URLs visited, navigation within zones or between zones, accesses to index pages, etc.

Additional features, such as average accessibility of the URLs visited, type of browser and so on, were extracted, but they were considered to be uninformative after analyzing their distribution in the database.

Table 6.5 shows the statistics for all the extracted features and the database. It includes average values, standard deviation and values for different percentiles, from 10 to 90 (from P_{10} to P_{90}).

A pure statistical analysis can give us some insights into the specific features of the website and its users. As it has been commented before, they show, for example, that the average time spent on *link*-type pages is greater than would be expected; it is greater than the time spent on *content*-type pages and on *other*-type pages that are mainly related to news. This is something that would not be expected in a normal situation and, therefore, might indicate a problem in the structure of the site or, at least, that the users are experiencing problems while navigating through *discapnet*.

Another feature to highlight is the small proportion of URL transitions made using the links within the website structure. This denotes an unusual navigation within the site, which could be due to the use of screen reader software such as Jaws or due to difficulty in finding items within the site and, as a consequence, in navigation through search engines such as Google.

Moreover, the values in the table show that the insights commented on the previous paragraph are not isolated but occur to a large number of the users. If we analyze the median values (P_{50}) of the elapsed time on a page, we can see that the problems are quite generalized because in 50% of the sequences, the average time spent on *link*-type pages (22.54 seconds) was nearly six times longer than the average time spent on *content*-type pages

Name	Description
Nreqs	Number of requests in the user session L
t_avg	Average time spent in each link $T_s \div L$
t_link	Average time spent in <i>link</i> -type URLs $\frac{\sum_{k=1}^L U_k^{-,link} * t_{U_k}}{\sum_{k=1}^L U_k^{-,link}}$
t_cont	Average time spent in <i>content</i> -type URLs $\frac{\sum_{k=1}^L U_k^{-,content} * t_{U_k}}{\sum_{k=1}^L U_k^{-,content}}$
t_oth	Average time spent in <i>other</i> -type URLs (they belong to the dynamic part of the website) $\frac{\sum_{k=1}^L U_k^{-,other} * t_{U_k}}{\sum_{k=1}^L U_k^{-,other}}$
%link	Proportion of <i>link</i> -type URLs $\frac{\sum_{k=1}^{L-1} U_k^{-,link}}{L}$
%cont	Proportion of <i>content</i> -type URLs $\frac{\sum_{k=1}^{L-1} U_k^{-,content}}{L}$
%other	Proportion of <i>other</i> -type URLs $\frac{\sum_{k=1}^L U_k^{-,other}}{L}$
%index	Proportion of <i>index</i> -type URLs $\frac{\sum_{k=1}^L U_k^{-,index}}{L}$
%directLink	Between all the transitions proportion of them done using direct links from the site $\frac{\sum_{k=1}^L D(U_k, U_{k+1})}{L}$ where $D(U_k, U_{k+1}) = \begin{cases} 1 & \exists \text{link } (U_k \rightarrow U_{k+1}) \text{ in } Adj.Mat. \\ 0 & \text{otherwise} \end{cases}$
%change	Proportion of times the access of a new URL involved a change of topic or zone $\frac{\sum_{k=1}^L C(U_k^{a,-}, U_{k+1}^{b,-})}{L}$ where $C(U_k^{a,-}, U_{k+1}^{b,-}) = \begin{cases} 1 & a \neq b \\ 0 & a = b \end{cases}$
%topic	Number of different topics or zones visited normalized with the length of the sequence $\frac{\sum_{k=1}^L TP_{U_k^a}}{L}$ where $TP_{U_k^a} = \begin{cases} 1 & \leftrightarrow \forall m < k: \nexists U_m^{b,-} b = a \\ 0 & \text{otherwise} \end{cases}$
change_topic	Relation between number of topic or zone changes and different topics visited $\frac{\%change}{\%topic}$
index_topic	Relation between the number of times the <i>index</i> page is visited and amount of different topic or zones visited $\frac{\%index}{\%topic}$

Table 6.4: Features extracted for problem detection and their description.

	Nreqs	t_avg	t_link	t_cont	t_oth	%link	%cont	%other	%index	%directLink	%change	%topic	change_topic	index_topic
Average	11.72	70.31	59.65	48.27	53.35	0.34	0.23	0.44	0.11	0.25	0.31	0.37	1.40	0.64
stdv	14.35	79.41	93.27	89.23	86.35	0.26	0.26	0.29	0.19	0.24	0.26	0.26	0.75	2.06
P_{10}	3.00	2.50	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.09	1.00	0.00
P_{25}	4.00	12.50	2.00	0.00	1.00	0.14	0.00	0.20	0.00	0.00	0.00	0.17	1.00	0.00
P_{50}	6.00	45.40	22.54	4.00	14.00	0.33	0.15	0.50	0.00	0.25	0.29	0.33	1.00	0.00
P_{75}	12.00	95.80	75.00	58.39	69.00	0.50	0.33	0.67	0.20	0.43	0.52	0.50	1.50	0.67
P_{90}	26.00	179.00	167.00	152.79	156.09	0.67	0.67	0.80	0.40	0.60	0.67	0.75	2.00	2.00

Table 6.5: Statistics in the database for the extracted features.

(4.00 seconds), which seems illogical and is twice as long as the time spent on *other*-type pages (14.00 seconds).

Moreover, if we compare the proportions of *link*- and *content*-type pages, we can see that half of the users seem to visit twice as many *link*-type pages as *content*-type pages, which should not be the aim of the site. Finally, with regard to navigation within the site, in P_{50} , only a quarter of the navigation is performed using links that exist in the website structure, indicating that the structure does not seem to be good for navigation.

The average values, although most likely too simplistic, show the overall behaviour of users navigating the *discapnet* website. We believe that the skewed values in some of these features or their combinations show anomalous behaviour, and, as a consequence, they might be problem indicators within the database.

6.5.2 Pattern discovery and analysis

The second phase of the system is the pattern discovery and analysis phase. Taking into account the fact that the provided database contains 241,311 user sessions, a manual analysis of the complete database to identify problems that the users of *discapnet* are having using the extracted features is not feasible; a reduction in the search volume is essential. We therefore designed a system capable of detecting these problems. The complete process is summarized in Figure 6.6.

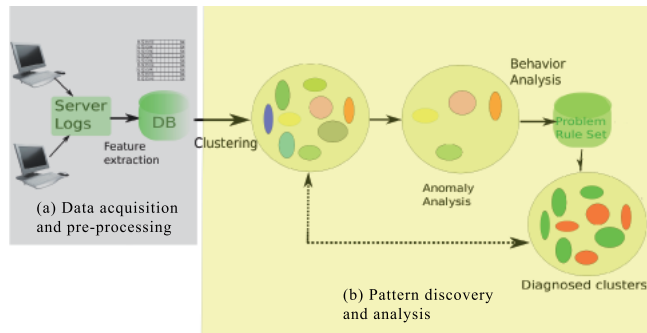


Figure 6.6: Problem discovery process and generation of the structure to be used in the automatic problem detection process.

6.5.2.1 Identification of problematic sessions

The aim of this section is to automatically select the sets of sequences showing the highest probability of being problematic and to focus the problem discovery process on these sequences.

The process consisted of grouping together users with similar behaviour patterns according to the new features. Based on our experience in previous works, we used a clustering algorithm (K-means [170]) to group together users with similar navigation patterns according to the extracted features, and we used the euclidean distance as a metric. Using these techniques, we grouped users behaving in a similar way into the same group. K-means requires the K parameter to be estimated. Bearing in mind that our aim at this point is to reduce the search space, a balance must be found between the uniformity of the patterns within a cluster and the number of different groups obtained. After several experiments, 1000 was the value of K selected to this end.

To reduce the pattern analysis effort required, we proposed to rank the clusters according to their degree of anomaly or decided which were the clusters more likely to group users with problems accessing the *discapnet* website. To this end, we defined an anomaly index to estimate the degree of anomaly of each cluster and only selected for analysis the clusters with the highest anomaly indices. The anomaly index was denoted by the number of anomalous features or by activated flags. The value of a variable was considered anomalous (or a flag was activated) if it was extreme compared to the average value: very small (being within the 10th percentile of the complete database, $feature_{value} < P_{10}$) or very large (not being within the 90th percentile of the complete database, $feature_{value} > P_{90}$). Table 6.6 shows an example of the 10 sessions with more flag activated. Being marked as red flags (the ones in bold) the large feature values and marked as green ones (the ones in italic) the low values according to P_{90} and P_{10} respectively. To find a trade-off between the cost of the analysis and the recall in the analysed clusters, we specifically selected for analysis the clusters whose centroids contained more than 30% anomalous features (4 or more flags activated), which comprised a set of 254 clusters.

6.5.2.2 Problem discovery and detection

The set of selected user groups and their representative patterns (the centroid in each of the clusters) were used for problem discovery. The focus during the problem discovery phase was on features with anomalous values,

cl.	Nreqs	t_avg	t_link	t_cont	t_oth	%link	%cont	%other	%index	%directLink	%change	%topic	change_topic	index_topic	# flags
50	3,05	271,09	271,09	0,00	0,00	1,00	0,00	0,00	0,00	0,00	0,67	0,99	1,01	0,00	12
779	3,19	246,22	246,22	0,00	0,00	1,00	0,00	0,00	0,00	0,00	0,40	0,66	1,11	0,00	10
325	3,31	26,13	0,00	0,00	26,13	0,00	0,00	0,01	0,00	0,33	0,68	0,98	1,00	0,00	9
652	3,22	246,37	0,00	246,37	0,00	0,00	1,00	0,00	0,00	0,02	0,51	0,67	1,25	0,00	9
720	3,38	26,52	26,53	0,00	0,00	1,00	0,00	0,00	0,34	0,00	0,00	0,00	1,00	1,14	9
979	3,22	24,23	24,27	0,00	0,06	1,00	0,00	0,00	0,00	0,00	0,68	0,99	1,00	0,00	9
66	3,56	82,98	0,00	83,00	0,02	0,00	1,00	0,00	0,00	0,00	0,01	0,30	1,00	0,00	8
82	3,23	7,77	9,11	0,00	5,05	0,67	0,00	0,00	0,00	0,00	0,69	1,00	1,00	0,00	8
205	5,10	24,31	29,25	16,44	1,30	0,68	0,24	0,07	0,74	0,00	0,00	0,00	1,00	3,76	8
284	3,28	18,70	27,02	2,44	0,00	0,66	0,34	0,00	0,66	0,00	0,00	0,00	1,00	2,14	8

Table 6.6: Top 10 of the cluster with activated flags

or activated flags, because we considered these to be the most informative indicators of problems.

As expected, the analysis of the features of these clusters led us to identify several symptoms and problems arising in the navigation logs of *discapnet*. The activation of some of the flags was interpretable for us, and we were able to diagnose the type of problem underlying the flags. We further diagnosed whether the origin of the problems seemed to be mainly related to the user or to the website structure, organization and content. Both types of problems were considered to be important: user problems, because they allow the identification of users having problems during navigation, and web-related problems, because, in addition to detecting users experiencing problems, they are useful from the service provider point of view and can help to improve the design of the site.

The profiles found, together with their features and diagnoses, are described in Table 6.7. The table includes a description of the main features and conditions (flags) used to define each of the profiles (some softer conditions were also used but are not shown for the sake of clarity) and some clues about the diagnosis made.

For instance, Profile1 where the user visits many *link*-type pages and spends long time in each of them could mean that the user has high indecision or problems finding her objective, i.e. we could state that the user is lost.

In Profile2 where the user visits many times the index pages for continuing in the same zone, could mean that the users have difficulties in the navigation or that the users are visually impaired. According to web accessibility experts, blind users used to get an image of the default page and try to navigate using that page over and over not to get lost in the depth of the website.

	Indicators/Symptoms of the problem	Diagnosis	Main Source
Profile1	- Many <i>link</i> -type pages - Long time in each <i>link</i> -type page $t_link > P_{90} \wedge (t_cont < P_{10} \vee t_oth < P_{10})$ $\wedge \%link > P_{90} \wedge (\%cont < P_{10} \vee \%oth < P_{10})$	- Lost - High indecision - Problems finding the objective	User
Profile2	- Index visited very often - A single topic visited $index_topic > P_{90} \wedge (\%topic < P_{10} \wedge change_topic < P_{10})$	- Difficulties in navigation - Users seem to start again from the memorized page (Blind)	User
Profile3	- Long time in pages $\{t_link > P_{90}, t_cont > P_{90}, t_oth > P_{90}\}$ two out of three	- Slow (Blind)	User
Profile4	- Variability of topics $\%topic > P_{90}$	- Lost - Looking for ambiguous concepts	Web
Profile5	- Long time in <i>content</i> or <i>other</i> -type pages. - Short time in <i>link</i> -type pages. - Few direct links. $(t_cont > P_{90} \vee t_oth > P_{90}) \wedge \%directLink < P_{10}$	- Seems to control navigation but slow. - Navigation via Google (structure problem) - Or blind person using Jaws	Web
Profile6	- Many changes and few topics. $change_topic > P_{90}$	- Lost in few topics (circular ABABABA). - It is not clear where to find.	Web
Profile7	- Long time in <i>link</i> -type pages. - Many direct links. $t_link > P_{90} \wedge \%directLink > P_{90}$	- High indecision. - Uses the site structure but not clear where to find	Web
Profile8	- Many direct links. - Many <i>link</i> -type pages. $\%link > P_{90} \wedge \%directLink > P_{90}$	- High indecision. - Does not arrive to content	Web
Profile9	- Many <i>index</i> pages. $\%index > P_{90}$	- Seem to start again from the memorized page (Blind)	Web

Table 6.7: Description of the extracted problematic profiles. Variables and conditions used to detect them, possible diagnosis and main source of the problems

In summary, the diagnosed users seem to have difficulties finding what they want. Some of them rarely reached *content*-type pages or spent a very long time on *link*-type pages; others went back and forth between a couple of topics; many were obliged to use external tools to navigate within the site instead of using the structure of the site; and many others were very slow, which could be due to their physical features because, as stated in the introduction, blind people tend to navigate less efficiently compared to able-bodied people using Jaws because of the sequential access.

Using the system described, we were able to discover some types of problems appearing in the navigation of the *discapnet* website and were even able to interpret them. The interpretation of the detected profiles was based only on the indicators detected because no insight was given into the type of users navigating the site. Therefore, using the process described, we generated a problem rule set (PRS) and a set of diagnosed clusters or user groups (see Figure 6.6) to be used in the exploitation phase.

6.5.3 Exploitation

The outcome of the pattern discovery phase can be used with different aims. It can be used to improve the structure of the site so that navigation becomes more comfortable for every user. According to the problems and definitions discovered with the proposed system, among the user sessions considered

for analysis in the site (those seeming to belong to real user navigation in *discapnet*), 33.5% of them (80,787 out of 241,311), i.e., a considerable number of users, appear to have had navigation problems, and any improvement in the site will especially benefit this group. Moreover, the site could include a system to automatically detect users having problems or to make a diagnosis for new users navigating the site (see Figure 6.7) while they are navigating and automatically introduce specific adaptations of the site according to the type of user detected.

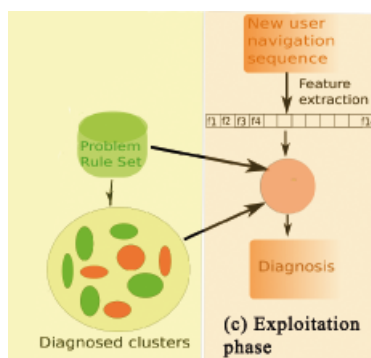


Figure 6.7: Problem detection process for new users navigating the site.

In the new users' diagnosis process, a new user is considered to be having problems if their navigation session is classified as problematic by the PRS or if its nearest cluster is labelled as problematic. Therefore, the problem detection system will use both the output of the PRS and the label of the nearest cluster (the first one detects diagnosed problems and the other is an early anomaly detector). If at least one of these indicates that the user is having a problem, the system will diagnose it.

6.5.4 Evaluation of the system

6.5.4.1 Evaluating the performance of the anomaly index

The process described in the previous section helped us to identify several problems based on the values of the features of a user session. Once the problem rule set (PRS) has been defined, it can be used to easily identify such problems within each user session. Although only part of the database was used for problem discovery, we analysed the database to look for users flagged as having problems and calculated the detection rate for the 254

clusters analysed.

Table 6.8 shows the number of users in the database seeming to have each of the profiles described and the detection percentage if only the 254 clusters classified as anomalous were analysed. The analysis showed that a high number of users flagged as having problems, especially those with the most important type of problems (user-related problems), were within the clusters with a high anomaly index. Moreover, the problems with a smaller detection rate, P6, P7, P8 and P9, belonged to the “Web” class, and two of them (P6 and P9) had the highest population, which was very high.

	User related			Web related						Total
	p1	p2	p3	p4	p5	p6	p7	p8	p9	
# sessions in complete DB	1,680	2,525	17,298	20,019	376	25,125	1,130	202	21,985	80,787
% in clusters labelled anomalous	85.34	90.7	58.5	82.3	100.0	30.3	15.1	42.1	24.9	43.4

Table 6.8: Number of users with problematic profiles in the database and % detected analyzing only 25% of the clusters.

Based on the problem definitions obtained, and as shown by the values in Table 6.8, 33.5% (80,787 out of 241,311) of the user sessions in the database indicate problems while navigating *discapnet*. Moreover, when analysing only 25% of the user sessions, we observed almost 45% of the users experiencing problems.

6.5.4.2 Experiment-based evaluation

For further evaluation and confirmation of our hypothesis, it was important to obtain logs for some user sessions where we had information about the users’ real characteristics: type of disability in the case that they have a disability or, otherwise, identifying that they do not have special requirements.

During recent months, we carried out some experiments in our laboratory with selected users involving navigation under observation in *discapnet*. The experiments allowed us to evaluate the problem detecting system with new but known users. We would expect to encounter some of the problems found for users with disabilities or to find users with web-type problems.

The logs for each of the users participating in the experiment were retrieved from the server logs and used to determine if their navigation was problematic based on the problem detection process (combination of PRS and labeled clusters) described in the previous section.

Two types of experiments were recorded: free navigation of users during a short period of time and searching for a target on the website. The second experiment is most likely nearer to reality because most users connect to a

	Free Navigation sessions	Target sessions	Total sessions
Visually impaired	13	11	24
Physically impaired	11	11	22
Able-bodied	5	5	10
Problem Rule Set	14	15	29
Diagnosed Cluster Set	14	8	22
Visually impaired	8	10	18
Physically impaired	10	10	20
Total	18	20	38
Percentage	75%	90.9%	82.6%

Table 6.9: Summary of information of the experiment-based evaluation. Indicating the analyzed sessions (upper part) and problem detection rates for disabled users (lower part).

website searching for specific information. Two main groups of users participated in the experiment: users with different types of disabilities, such as blind users and users with physical disabilities, and users without known disabilities or able-bodied users. We collected 46 user sessions for the first group (22 for physically disabled people, and 24 for visually impaired people), and 10 for the second group. The first two rows in Table 6.9 summarize the distribution of the acquired user sessions.

The log data obtained for each user session was pre-processed to extract the features described in Section 6.5.1.1, and the diagnosis module was used to determine if the users were having problems during navigation. We considered a user to be experiencing problems if they were automatically classified as it using the problem rule set or if their nearest cluster was considered to be experiencing problems because this could mean that although the problem detection flags had not yet been activated for the user, the users with more similar behaviour had problems, and, as a consequence, they might be expected to have problems soon. The results of this process for users with disabilities are shown in Table 6.9.

The numbers in the table show that most disabled users were diagnosed by our system as having experienced problems during their navigation, reaching a 90.9% detection rate in the experiment with the target, which is, from our point of view, the most realistic situation.

As the values in the table show, the fact that *discapnet* is mainly aimed at visually impaired people seems to affect to the problems the users find while navigating. Physically impaired users tend to have navigation problems with higher probability than visually impaired people: 10 users with physical

disabilities out of 11 had problems for navigating freely in the site whereas that was the case for 8 out of 13 visually impaired users. In the case of navigation with target, 10 out of 11 users found problems in both cases physically and visually impaired people.

We carried out a further analysis of the hypothetical source of the problems the users with disabilities are having according to our system. The analysis showed that they were distributed almost homogeneously. Among the found problems the source of 20 of them was hypothetically the user, whereas for the other 18, the source seemed to be the website.

In the case of the 10 user sessions for users without known disabilities, or able-bodied users, the system detected problems in five sessions, but all of these were related to the website.

6.6 Summary

Web personalization becomes essential in industries and specially for the case of users with disabilities such as visually impaired people. Adaptation may very much speed up the navigation of visually impaired people and contribute to diminish the existing technological gap. In this context, the proposed systems will probably contribute to improve the disabled user's navigation experience.

On the one hand, concerning the link prediction system, we concluded that both approaches the global and the modular approaches, are effective for helping the users finding what they are looking for. However, the modular approach achieves better results than the global one. The modular approach outperforms obtaining values of nearly 60% for precision and 45% for recall. This means that when 25% of the navigation of the new user has happened the designed system is able to propose a set of links where nearly 60% of them (2 out of 3) is among the ones the new user will be using in the future and this will definitely make the navigation of the user easier.

On the other hand, in the problem detection system, the system models users in the wild and discovers navigation problems in *discapnet*. The pattern discovery phase was performed using clustering and is based on 14 features extracted from each user session. The outcome of this phase, the problem rule set (PRS) and the set of labelled clusters, can be used to efficiently detect users experiencing problems during navigation. This was demonstrated by a controlled experiment carried out with mainly disabled people, where 81.5% of users with disabilities was automatically flagged as having experienced problems. In comparison, in the original logs used to

build the system, where a more diverse range of people were accessing the system, only 33.5% of the users experienced problems.

Part V

Conclusions

Chapter 7

Conclusions and Further work

7.1 Conclusions

The dramatic growth of internet and the easy availability of the information on the Web, have brought a new information age. In the era of the Web, the information overload problem is continuously expanding. The users are very often overwhelmed by a huge amount of information that is available online when they are browsing the Web. Undoubtedly, the ever more complex structure of sites and the heterogeneous nature of the Web, make extremely difficult the web navigation for users, who often are faced with the challenging problem of finding the desired information in the right time.

In this context it is necessary to provide tools to the user to access easier to the desired information. This can be achieved introducing adequate adaptations and requires normally user profiling from different points of view: their interests, their navigation habits, their abilities, among others. The obtained profiles will support the design of the required adaptations or personalization strategies (web personalization). In order to build these profiles user information must be acquired. This can be done intrusively, that is, inquiring information explicitly to the user, or non intrusively, i.e. extracting information directly from the user activity (user navigation logs, geographical information...). Obviously, from the user point of view, a non intrusive solution will be preferred. In the context of web navigation, the context of this work, web miming techniques have shown to be adequate as non intrusive solutions.

The aim of this work is the **application of web mining techniques based on different information sources: usage, content and structure, to build valuable systems for users and service providers**. The systems have been developed in two different contexts: a tourism web site (*bidasoa turismo*, www.bidasoaturismo.com), and a website addressed to people with disabilities (www.discapnet.es, website supported by ONCE-Technosite).

The system developed in **the tourism context** and presented in this dissertation is a **combination of web usage mining and web content mining techniques**. Based on the experience acquired from previous works carried out in ALDAPA research group, we built navigation profiles for a link prediction system. Starting from the server logs of the *bidasoa turismo* website, provided by the *bidasoa turismo* DMO (which is a tourism organization responsible for management the promotion of the destination of Bidasoa Txingudi bay) we followed the typical phases of a web mining system: data acquisition and pre-processing phase, pattern discovery and analysis phase and exploitation phase.

In the first phase, data acquisition and pre-processing phase, we first selected the entries directly related with the user requests erasing the ones automatically generated. Then we applied a sessioning phase and obtained a sequence representation of the user sessions.

In the second phase, pattern discovery and analysis phase, the machine learning techniques were applied combining a clustering algorithm (PAM was used in order to grouping into the same segment users that show similar navigation patterns) with a sequential pattern mining technique (SPADE was used for extracting the most common click sequences of each cluster) for creating navigation models. The model created was used in the exploitation phase (using kNN algorithm) for proposing links to new users when they start the navigation.

The system was evaluated based on precision, recall and F-measure. **According to the results, the system obtained good quality profiles that match in more than 60% the real user navigation sequences (precision), achieving 26.7% in a link prediction system**. The obtained values could be seen as a lower bound because, although not appearing in the user navigation sequence, the proposed links could be useful and interesting for the user. Therefore, **we analysed the navigation profiles according to interests and, the results showed that a high percentage of the links proposed by our system is of interest to the new users (over 90% of precision for profilers and over 70% for link prediction systems)**.

Taking into account that the link prediction system provided good results and considering that URLs with similar or related content to the ones appearing in the navigation profile will also be interesting for the user **we enriched the previous system introducing semantic information** (web content information) into usage information based profiles.

Different content analysis tools were tested for the enrichment, three based on **statistical methods** (keyword extractor, search engines and topic modelling), and one ontology based method. Although all the explored statistical content analysis tools **showed to be useful** for our aims (the inclusion of any of the statistical methods outperformed the results of usage based link prediction systems), **topic modelling seemed to be the most appropriate** one since it also provides information about the semantic structure of the site.

The introduction of ontology based text comparison methods provides richer relationships among texts and this allows the design of more sophisticated enrichment strategies. The implemented strategies improve **the link prediction systems carried out** using only usage information or using statistical methods.

The best results were obtained with the ontology based system achieving precision values for the link prediction system, which are over 45% with a 35% of performance improvement taking into account the links proposed based on profiles compared to the SPADE based system (without semantic data).

The third branch of the created system is the **generation of semantic or interest profiles**. Interest profiles have been generated **combining web usage information and content information**. Using the topical structure of the site, we obtained an interest based representation of the user sessions and we used clustering (K-means algorithm) to obtain interest profiles.

We carried out **three analysis** of the users' interests: a **global** analysis (we extracted global profiles), **language-dependent** analysis (considering that the access language is an indicator of the origin of the user) and **time-dependent** analysis (taking into account the time period of the navigations).

The comparison of the profiles showed that **there are differences between the users** accessing in different languages, i.e. the users from different nationalities look for different things and are attracted by different information. Moreover, there are differences as well between the users accessing in different time periods of the year, what means that people depending on the period of the year change their interests. The outcome of this analysis

will be useful for service providers. They can redesign the sites accordingly or use it in future marketing campaigns.

The information extracted from the interest profiling was corroborated by the service provider.

In **the context of disabled people**, two different systems were proposed, a **link prediction** system similar to the one implemented for *bidasoa turismo*; and an **automatic problem detection system** which uses three sources of information for web mining: web structure, content and usage information.

In the link prediction system, the phases followed to build the system were exactly the same as in the system built in the tourism context, but adapted to the specific site. A thorough analysis of the site was carried out to select adequate zones for analysis. Due to the characteristics of the site **two approaches** were implemented: **a global approach and a modular approach**.

In the global approach the clustering and profiling process were done for the whole site and in the modular case the website was split into 8 different zones. Both approaches showed to be effective for link prediction but **the modular approach outperforms the global one**.

The second system proposed, **the problem detection system** models users in the wild and **discovers navigation problems** appearing in *discapnet* **and can also be used for problem detection when new users are navigating** the site. In this system **the three information sources were used to extract informative features**. Fourteen different features were selected for being able to detect navigation problems. A system based on anomaly flags was implemented. The process consists of grouping together users with similar behaviour patterns according to the new features (based on K-means algorithm). Then the clusters with the highest anomaly indices and the activated flags were selected to diagnose the type of problems underlying the users. A problem rule set was generated and used together with the set of diagnosed clusters to detect problems when new users are navigating the site. An hypothesis about the source of the detected problems was done concluding that some are due to website problems and some others due to physical problems.

A controlled experiment carried out demonstrated that the designed system identifies automatically %85 of the users with disabilities as users having problems. This is very useful information for service provider in order to have the chance to modify the site making it more comfortable and accessible for users.

In summary the presented work shows that as other works addressed, machine learning techniques are adequate to build user navigation profiles in real websites of diverse characteristics and areas which can be used to support their navigation suggesting interesting links. The combination of clustering and SPADE has shown to be effective to first find the structure and then obtain the profiles. Furthermore, this work has shown that the inclusion of semantic information and also some structure information in the mining process, has many advantages such as improving the quality on the navigation profiles, providing the possibility of including new link proposal strategies, discovering different types of profiles, as for example interest profiles, and discovering users having problems to navigate in a website. We therefore claim that the combination of usage, content and structure information in web mining processes should be the trend to make the most of the stored information.

7.2 Further Work

There are several lines of work that remain open in relation to the outcomes presented in this dissertation.

In general, in the systems proposed in this thesis, re-learning is used to modify the profiles as new data is generated. Although being a batch process this is not very critical, a further work could be to use incremental learning not to repeat the complete modelling process when the input data changes. Using incremental learning the system will save on computational time.

Another possible line of work is to combine web content, usage and structure mining for the link prediction system. The structure of the site could be used in order to increase the precision, recall and F-measures values.

Furthermore, more lines for each specific domain remain open.

7.2.1 Tourism context: *bidasoia turismo*

Although we presented a complete work, the system is not closed and many new ideas to be implemented in the future appeared during its development. In this work we used a deterministic approach to assign interest profiles to clusters; we selected a single topic as representative. However, a fuzzy approach, where more than one topic is extracted as interest profile for each cluster, would probably be more realistic.

In addition, more realistic evaluations of the system could be done using controlled experiments, collecting the real tourists' feedback in real time.

Regarding to the system using ontologies for content analysis it is a system that is open to many more strategies for enriching the link prediction system. The enrichment and strategies could be defined using different criteria, such as marketing criteria and helping people with their browsing criteria, among others. Moreover, as we think that the relevance of the text within the tags (bold, italics and so on) is important we consider that some reflection should be done about how to use this information to improve the performance of the system in the future.

7.2.2 Disabled people context: *discapnet*

Two systems were presented in this context: a link prediction system and a problem detection system for helping disabled people, however we consider that a single system merging both systems would be interesting. In addition we contemplate the inclusion of specific on-line adaptations of the site according to the problem detected in the user navigation, so that it can be either reduced or removed in the future.

Apart from the lines mentioned, we consider as well the option of expanding the zones of the website to explore as could be (news, canal senior...).

7.3 Related Publications

The majority of the work presented in this dissertation has already been published. The complete list of publications appears below.

- International Journals:
 - O. Arbelaitz, I. Gurrutxaga, A. Lojo, J. Muguerza, J.M. Pérez, I. Perona. (2013). “Web usage and content mining to extract knowledge for modelling the users of the Bidasoa Turismo website and to adapt it ”. *Expert Systems with Application*.
 - O. Arbelaitz, A. Lojo, J. Muguerza, I. Perona. (2015). “Web mining for navigation problem detection and diagnosis in Discapnet: a website aimed at disabled people”. *Journal of the Association for Information Science and Technology: JASIST*.
- International Conferences:
 - O. Arbelaitz, I. Gurrutxaga, A. Lojo, J. Muguerza, J.M. Pérez, I. Perona. (2012). “Enhancing a Web Usage Mining based Tourism Website Adaptation with Content Information”. *International*

Conference on Knowledge Discovery and Information Retrieval (KDIR).

- O. Arbelaitz, I. Gurrutxaga, A. Lojo, J. Muguerza, J.M. Pères, I. Perona. (2013). “A Navigation-log based Web Mining Application to Profile Users from Different Origins Accessing the Web of Bidasoa Turismo”. Information and communication Technologies in Tourism: Enter.
- O. Arbelaitz, A. Lojo, J. Muguerza, I. Perona. (2013). “Datuetatik ezagutzara. Web orrietan nabigatzean utzitako azterna abiapuntu”. EKAIA.
- O. Arbelaitz, A. Lojo, J. Muguerza, I. Perona. (2014). “Global versus modular link prediction approach for discapnet: website focused to visually impaired people”. International Symposium Advances in Artificial Intelligence and Applications: AAIA.

- Internal Reports:

- O. Arbelaitz, A. Lojo, J. Muguerza, I. Perona. (2012). "Aplicación de Técnicas de Minería de Datos para la Extracción de Conocimiento de la Web de Bidasoa Turismo". Internal report EHU-KAT-IK-13-12, Universidad del País Vasco Euskal Herriko Unibertsitatea.

There is still a work in progress to be submitted to WWW journal related to the introduction of ontology based methods to analyse the semantic structure.

Bibliography

- [1] World-Stats, “Internet world stats usage and population statistics.” <http://www.internetworldstats.com/stats.htm>, 2015.
- [2] S. Chakrabarti, “Data mining for hypertext: A tutorial survey,” *ACM SIGKDD Exploration Newsletter*, vol. 1, no. 2, pp. 1–11, 2000.
- [3] Usa-Today, “Usa today: Latest world and us news.” <http://www.usatoday.com>, 2003.
- [4] P. Brusilovsky, A. Kobsa, and W. Nejdl, eds., *The Adaptive Web: Methods and Strategies of Web Personalization*, vol. 4321. Springer, 2007.
- [5] E. García, C. Romero, S. Ventura, and C. D. Castro, “An architecture for making recommendations to courseware authors using association rule mining and collaborative filtering,” *User Modeling and User-Adapted Interaction*, vol. 19, no. 1-2, pp. 99–132, 2009.
- [6] ETC, “New media trend watch - online travel market.” The European Travel Commission (ETC): <http://www.newmediatrendwatch.com/world-overview/91-online-travel-market?showall=1>, 2012.
- [7] U. Gretzel, Y. Yuan, and D. Fesenmaier, “Preparing for the new economy: Advertising strategies and change in destination marketing organizations,” *Journal of Travel Research*, vol. 39, no. 2, pp. 146–156, 2000.
- [8] A. Steinbauer and H. Werthner, “Consumer behaviour in e-tourism,” in *Information and Communication Technologies in Tourism (ENTER)*, pp. 65–76, 2007.
- [9] UNWTO, *Technology in Tourism*, vol. 1. World Tourism Organization (UNWTO) Affiliate Members AM-reports, 2011.
- [10] WAI, “Wai guidelines and techniques.” <http://www.w3.org/WAI/guid-tech.html>, 2006.
- [11] A. Kobsa, “A component architecture for dynamically managing privacy constraints in personalized web-based systems,” *Privacy Enhancing Technologies*, pp. 177–188, 2003.

-
- [12] J. B. Schafer, J. Konstan, and J. Riedl, "Recommender systems in e-commerce," in *Proceedings of the 1st ACM Conference on Electronic Commerce*, pp. 158–166, 1999.
- [13] G. Semeraro, V. Andersen, H. Andersen, M. de Gemmis, and P. Lops, "User profiling and virtual agents: a case study on e-commerce services," *Universal Access in the Information Society*, vol. 7, no. 3, pp. 179–194, 2008.
- [14] M. Vigo, B. Leporini, and F. Paternò, "Enriching web information scent for blind users," in *Proceedings of the 11th International ACM SIGACCESS Conference on Computers and Accessibility*, pp. 123–130, 2009.
- [15] A. Kobsa, J. Koenemann, and W. Pohl, "Personalized hypermedia presentation techniques for improving online customer relationships," *The Knowledge Engineering Review*, vol. 16, no. 2, pp. 111–155, 2001.
- [16] G. Castellano, L. C. Jain, and A. M. Fanelli, *Web Personalization in Intelligent Environments*, vol. 229. Springer, 2009.
- [17] S. Schiaffino and A. Amandi, "Artificial intelligence," in *Lecture Notes in Artificial Intelligence (LNAI 5640)*, pp. 193–216, 2009.
- [18] M. Claypool, P. Le, M. Wased, and D. Brown, "Implicit interest indicators," in *Proceedings of the 6th international conference on Intelligent user interfaces*, pp. 33–40, 2001.
- [19] M. Zanker, M. Fuchs, W. Höpken, M. Tuta, and N. Müller, "Evaluating recommender systems in tourism - a case study from austria," *Information and Communication Technologies in Tourism (ENTER)*, pp. 24–34, 2008.
- [20] M. K. Sarkaleh, M. Mahdavi, and M. Baniardalan, "Designin a tourism recomender system based on location, mobile device and user features in museum," *International Journal of Managing Information Technology*, vol. 4, no. 2, pp. 13–21, 2012.
- [21] U. Leimstoll and H. Stormer, "Collaborative recommender systems for online shops.," in *Proceedings of the Eighth Americas Conference on Information System (AMCIS)*, p. 156, 2007.
- [22] A. H. N. Rafsanjani, N. Salim, A. R. Aghdam, and K. B. Frad, "Recommendation systems: a review," *International Journal of Computational Engineering Reseach*, vol. 3, no. 5, pp. 47–52, 2013.
- [23] P. Melville and V. Sindhvani, "Recommender systems," in *Encyclopedia of Machine Learning*, pp. 829–838, 2010.
- [24] J. S. Breese, D. Heckerman, and C. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," in *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, pp. 43–52, 1998.

-
- [25] H. Lieberman, "Letizia: An agent that assists web browsing," in *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, pp. 924–929, 1995.
- [26] M. Pazzani, J. Muramatsu, and D. Billsus, "Syskill & webert: Identifying interesting web sites," in *Proceedings of the 13th National Conference on Artificial Intelligence*, pp. 54–61, 1998.
- [27] M. Pazzani and D. Billsus, "Learning and revising user profiles: The identification of interesting web sites," *Machine Learning*, vol. 27, no. 3, pp. 313–331, 1997.
- [28] L. Chen and K. Sycara, "Webmate: A personal agent for browsing and searching," in *Proceedings of the Second International Conference on Autonomous Agents*, pp. 132–139, 1998.
- [29] D. Mladenic, "Machine learning used by personal webWatcher," in *Proceedings of ACAI-99 Workshop on Machine Learning and Intelligent Agents*, pp. n/a–n/a, 1999.
- [30] B. Sheth and P. Maes, "Evolving agents for personalized information filtering," in *Artificial Intelligence for Applications Proceedings Ninth Conference*, pp. 345–352, 1993.
- [31] D. Billsus and M. J. Pazzani, "A hybrid user model for news story classification," in *Proceedings of the Seventh International Conference on User Modeling*, pp. 99–108, 1999.
- [32] B. Magnini and C. Strapparava, "Improving user modelling with content-based techniques," in *User Modeling*, vol. 2109, pp. 74–83, Springer Berlin Heidelberg, 2001.
- [33] M. Degemmis, P. Lops, and G. Semeraro, "A content-collaborative recommender that exploits wordnet-based user profiles for neighborhood formation," *User Modeling and User-Adapted Interaction*, vol. 17, no. 3, pp. 217–255, 2007.
- [34] I. Cantador, A. Bellogín, and P. Castells, "News@hand: A semantic web approach to recommending news," in *Adaptive Hypermedia and Adaptive Web-Based Systems*, vol. 5149, pp. 279–283, Springer Berlin Heidelberg, 2008.
- [35] R. R. Sinha and K. Swearingen, "Comparing Recommendations Made by Online Systems and Friends," in *DELOS Workshop: Personalisation and Recommender Systems in Digital Libraries*, pp. n/a–n/a, 2001.
- [36] D. Goldberg, D. Nichols, B. M. Oki, and D. Terry, "Using collaborative filtering to weave an information tapestry," *Communications of the ACM*, vol. 35, no. 12, pp. 61–70, 1992.
- [37] J. B. Schafer, D. Frankowski, J. Herlocker, and S. Sen, "Collaborative filtering recommender systems," *Foundations and Trends in Human-Computer Interaction*, vol. 4, no. 2, pp. 81–173, 2007.

- [38] B. Marlin, "Modeling user rating profiles for collaborative filtering," in *Neural Information Processing Systems*, pp. 627–634, 2003.
- [39] D. Y. Pavlov and D. M. Pennock, "A maximum entropy approach to collaborative filtering in dynamic, sparse, high-dimensional domains," in *Proceedings of Neural Information Processing Systems*, pp. 1441–1448, 2002.
- [40] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, "GroupLens: An open architecture for collaborative filtering of netnews," in *Proceedings of ACM Conference on Computer Supported Cooperative Work*, pp. 175–186, 1994.
- [41] U. Shardanand and P. Maes, "Social information filtering: Algorithms for automating word of mouth," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 210–217, 1995.
- [42] W. Hill, L. Stead, M. Rosenstein, and G. Furnas, "Recommending and evaluating choices in a virtual community of use," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 194–201, 1995.
- [43] Y. Ge, H. Xiong, A. Tuzhilin, and Q. Liu, "Collaborative filtering with collective training.," in *ACM Recommender Systems (RecSys)*, pp. 281–284, 2011.
- [44] M. Gori and A. Pucci, "Itemrank: A random-walk based scoring algorithm for recommender engines," in *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, pp. 2766–2771, 2007.
- [45] D. Kelly and J. Teevan, "Implicit feedback for inferring user preference: A bibliography," *Special Interest Group on Information Retrieval forum (ACM SIGIR forum)*, vol. 37, no. 2, pp. 18–28, 2003.
- [46] Y. S. Kim, B. J. Yum, J. Song, and S. M. Kim, "Development of a recommender system based on navigational and behavioral patterns of customers in e-commerce sites," *Expert System with Application*, vol. 28, no. 2, pp. 381–393, 2005.
- [47] Y. S. Kim and B.-J. Yum, "Recommender system based on click stream data using association rule mining," *Expert System with Application*, vol. 38, no. 10, pp. 13320–13327, 2011.
- [48] J. Lee, M. Podlaseck, E. Schonberg, and R. Hoch, "Visualization and analysis of clickstream data of online stores for understanding web merchandising," *Data Mining and Knowledge Discovery*, vol. 5, no. 1-2, pp. 59–84, 2001.
- [49] R. Burke, "Hybrid recommender systems: Survey and experiments," *User Modeling and User-Adapted Interaction*, vol. 12, no. 4, pp. 331–370, 2002.
- [50] M. Balabanović and Y. Shoham, "Fab: Content-based, collaborative recommendation," *Communications of the ACM*, vol. 40, no. 3, pp. 66–72, 1997.

-
- [51] M. Claypool, A. Gokhale, T. Miranda, P. Murnikov, D. Netes, and M. Sartin, "Combining content-based and collaborative filters in an online newspaper," *In Proceedings ACM-SIGIR Workshop on Recommender Systems*, pp. n/a–n/a, 1999.
- [52] P. Symeonidis, "Content-based dimensionality reduction for recommender systems," in *Data Analysis, Machine Learning and Applications*, pp. 619–626, Springer Berlin Heidelberg, 2008.
- [53] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl, "Evaluating collaborative filtering recommender systems," *ACM Transactions on Information Systems*, vol. 22, no. 1, pp. 5–53, 2004.
- [54] T. Miranda, M. Claypool, M. Claypool, A. Gokhale, A. Gokhale, T. Mir, P. Murnikov, P. Murnikov, D. Netes, D. Netes, M. Sartin, and M. Sartin, "Combining content-based and collaborative filters in an online newspaper," in *Proceedings of ACM SIGIR Workshop on Recommender Systems*, pp. n/a–n/a, 1999.
- [55] R. Kosala and H. Blockeel, "Web mining research: A survey," *ACM SIGKDD Exploration Newsletter*, vol. 2, no. 1, pp. 1–15, 2000.
- [56] T. Srivastava, P. Desikan, and V. Kumar, "Web mining – concepts, applications and research directions," in *Foundations and Advances in Data Mining*, vol. 180, pp. 275–307, Springer Berlin Heidelberg, 2005.
- [57] J. Fürnkranz, "Web structure mining exploiting the graph structure of the world-wide web," *ÖGAI Journal*, vol. 21, no. 2, pp. 17–26, 2002.
- [58] S. Jeyalatha and B. Vijayakumar, "Design and implementation of a web structure mining algorithm using breadth first search strategy for academic search application," in *International Conference for Internet Technology and Secured Transactions (ICITST)*, pp. 648–654, 2011.
- [59] C. hue Moh, E. peng Lim, and W. keong Ng, "Dtd-miner: A tool for mining dtd from xml documents," in *Proceedings of the 2nd International Workshop on Advanced Issues of E-Commerce and Web-Based Information Systems*, pp. 144–151, 2000.
- [60] L. Getoor, "Link mining: A new data mining challenge," *ACM SIGKDD Exploration Newsletter*, pp. 1–6, 2003.
- [61] J. M. Kleinberg, "Authoritative sources in a hyperlinked environment," *Journal of the ACM*, vol. 46, no. 5, pp. 604–632, 1999.
- [62] L. Page, S. Brin, R. Motwani, and T. Winograd, "The pagerank citation ranking: Bringing order to the web," Technical Report 1999-66, Stanford InfoLab, 1999.
- [63] L. Vaughan and J. You, "Keyword enhanced web structure mining for business intelligence," in *Interactive Topic Information System (SITIS)*, vol. 4879, pp. 161–168, 2006.

- [64] C. h. Li and C. c. Kit, "Web structure mining for usability analysis," *ACM International Conference on Web Intelligence*, pp. 309–312, 2005.
- [65] J. Choi and G. Lee, "New techniques for data preprocessing based on usage logs for efficient web user profiling at client side," in *Web Intelligence/IAT Workshops*, pp. 54–57, 2009.
- [66] Z. Pabarskaite and A. Raudys, "A process of knowledge discovery from web log data: Systematization and critical review," *Journal of Intelligent Information Systems*, vol. 28, no. 1, pp. 79–104, 2007.
- [67] B. Berendt and M. Spiliopoulou, "Analysis of navigation behaviour in web sites integrating multiple information systems," *The Very Large Data Bases Journal*, vol. 9, no. 1, pp. 56–75, 2000.
- [68] B. Mobasher, R. Cooley, and J. Srivastava, "Automatic personalization based on web usage mining," *Communications of the ACM*, vol. 43, no. 8, pp. 142–151, 2000.
- [69] S. Gunduz and M. T. Ozsü, "A web page prediction model based on click-stream tree representation of user," in *Proceedings of the 9th ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, pp. 535–540, 2003.
- [70] R. Srikant and Y. Yang, "Mining web logs to improve website organization," in *Proceedings of the 10th International Conference on World Wide Web*, pp. 430–437, 2001.
- [71] P. Makkar, P. Gulati, and A. Sharma, "A novel approach for predicting user behavior for improving web performance," *International Journal on Computer Science and Engineering (IJCSSE)*, vol. 2, no. 4, pp. 1233–1236, 2010.
- [72] S. Bhawsar, K. Pathak, and V. Patidar, "New framework for web access prediction," *International Journal of Computer Technology and Electronics Engineering (IJCTEE)*, vol. 2, no. 1, pp. 48–53, 2012.
- [73] T. M. Kroegeer, D. D. E. Long, and J. C. Mogul, "Exploring the bounds of web latency reduction from caching and prefetching," in *Proceedings of the USENIX Symposium on Internet Technologies and Systems*, p. 2, 1997.
- [74] Chen, Xin, Zhang, and Xiaodong, "A popularity-based prediction model for web prefetching," *Computer*, vol. 36, no. 3, pp. 63–70, 2003.
- [75] A. Anitha, "A new web usage mining approach for next page access prediction," *International Journal of Computer Applications*, vol. 8, no. 11, pp. 7–9, 2010.
- [76] I. Zukerman, D. W. Albrecht, and A. E. Nicholson, "Predicting users' requests on the WWW," in *Proceedings of the seventh international conference on User modeling*, pp. 275–284, 1999.

- [77] B. S. Chordia and K. P. Adhiya, "Grouping web access sequences using sequence alignment method," *Indian Journal of Computer Science and Engineering (IJCSE)*, vol. 2, no. 3, pp. 308–314, 2011.
- [78] H. Kaur and S. Chawla, "Web data mining: Exploring hidden patterns, its types and web content mining techniques and tools," *International Journal of Innovative Science and Modern Engineering (IJISME)*, vol. 3, pp. 34–36, 2014.
- [79] R. Malarvizhi and K. Saraswath, "Web content mining techniques tools & algorithms – a comprehensive study," *International Journal of Computer Trends and Technology (IJCTT)*, vol. 4, pp. 2940–2945, 2013.
- [80] F. G. Taddesse, J. Tekli, R. Chbeir, M. Viviani, and K. Yetongnon, "Semantic-based merging of rss items," *World Wide Web*, vol. 13, no. 1-2, pp. 169–207, 2010.
- [81] Z. S. Zubi, "Using some web content mining techniques for arabic text classification," in *Proceedings of the 8th WSEAS International Conference on Data Networks, Communications, Computers*, pp. 73–84, 2009.
- [82] P. Chaovalit and L. Zhou, "Movie review mining: a comparison between supervised and unsupervised classification approaches," in *Proceedings of the 38th Annual Hawaii International Conference*, pp. 112c–112c, 2005.
- [83] M. Schedl and G. Widmer, "Automatically detecting members and instrumentation of music bands via web content mining," in *Adaptive Multimedia Retrieval: Retrieval, User, and Semantics*, vol. 4918, pp. 122–133, Springer Berlin Heidelberg, 2008.
- [84] R. Campos, G. Dias, and C. Nunes, "Wise: Hierarchical soft clustering of web page search results based on web content mining techniques," in *Proceedings of the ACM International Conference on Web Intelligence*, pp. 301–304, 2006.
- [85] I. Pollach, A. Scharl, and A. Weichselbraun, "Web content mining for comparing corporate and third-party online reporting: a case study on solid waste management," *Business Strategy and the Environment*, vol. 18, no. 3, pp. 137–148, 2009.
- [86] M. Agyemang, K. Barker, and R. S. Alhaji, "Mining web content outliers using structure oriented weighting techniques and n-grams," in *Proceedings of the 2005 ACM Symposium on Applied Computing*, pp. 482–487, 2005.
- [87] J. Pokorný and J. Smizanský, "Page content rank: an approach to the web content mining," in *International Association for Development of Information Society (IADIS)*, pp. 289–296, 2005.
- [88] S. Taherizadeh and N. Moghadam, "Integrating web content mining into web usage mining for finding patterns and predicting users' behaviors," *International Journal of Information Science and Management*, vol. 7, no. 1, pp. 51–66, 2009.

- [89] B. Mobasher, H. Dai, T. Luo, Y. Sun, and J. Zhu, "Combining web usage and content mining for more effective personalization," in *Proceedings of the International Conference on E-Commerce and Web Technologies (ECWeb)*, pp. n/a–n/a, 2000.
- [90] L. Chen and W. L. Chue, "Using web structure and summarisation techniques for web content mining," *Information Processing & Management*, vol. 41, no. 5, pp. 1225–1242, 2005.
- [91] D. S. Babu, P. Sathish, and J. Ashok, "Fusion of web structure mining and web usage mining," *International Journal of Computer Science and Information Technologies (IJCSIT)*, vol. 2, no. 3, pp. 965–967, 2011.
- [92] P. Senkul and S. Salin, "Improving pattern quality in web usage mining by using semantic information," *Knowledge and Information Systems*, vol. 30, no. 30, pp. 527–541, 2012.
- [93] T. M. Mitchell, *Machine Learning*. McGraw-Hill, Incorporation, 1997.
- [94] R. Kohavi and F. Provost, "Glossary of terms," *Machine Learning*, vol. 30, no. 2-3, pp. 271–274, 1998.
- [95] P. Cunningham, M. Cord, and S. Delany, "Supervised learning," in *Machine Learning Techniques for Multimedia*, pp. 21–49, Springer Berlin Heidelberg, 2008.
- [96] Q. Liu and Y. Wu, "Supervised learning," in *Encyclopedia of the Sciences of Learning*, pp. 3243–3245, Springer US, 2012.
- [97] J. R. Quinlan, "Induction of decision trees," *Machine Learning*, vol. 1, no. 1, pp. 81–106, 1986.
- [98] J. Pearl, "Bayesian networks: A model of self-activated memory for evidential reasoning," in *Proceedings of the 7th Conference of the Cognitive Science Society, University of California, Irvine*, pp. 329–334, 1985.
- [99] J. A. Anderson and E. Rosenfeld, eds., *Neurocomputing: Foundations of Research*. MIT Press, 1988.
- [100] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.
- [101] P.-N. Tan, M. Steinbach, and V. Kumar, *Introduction to data mining*. Pearson Addison Wesley, 2006.
- [102] J. Macqueen, "Some methods for classification and analysis of multivariate observations," in *5-th Berkeley Symposium on Mathematical Statistics and Probability*, pp. 281–297, 1967.
- [103] L. Kaufman and P. J. Rousseeuw, *Finding Groups in Data: An Introduction to Cluster Analysis*. Wiley-Interscience, 1990.

- [104] S. Vijayarani and S. Deepa, "Sequential pattern mining: A study," *IJCA Proceedings on International Conference on Research Trends in Computer Technologies*, pp. 14–18, 2013.
- [105] C. Antunes and A. Oliveira, "Generalization of pattern-growth methods for sequential pattern mining with gap constraints," in *Machine Learning and Data Mining in Pattern Recognition*, vol. 2734, pp. 239–251, Springer, 2003.
- [106] T. Slimani and A. Lazzez, "Efficient analysis of pattern and association rule mining approaches," *International Journal of Information Technology and Computer Science (IJITCS)*, vol. 6, no. 3, pp. 70–81, 2014.
- [107] V. S. Motegaonkar and M. V. Vaidya, "A survey on sequential pattern mining algorithms," *International Journal of Computer Science and Information Technologies (IJCSIT)*, vol. 5, no. 2, pp. 2486–2492, 2014.
- [108] C. Chetna, T. Amit, and G. Amit, "Sequential pattern mining: Survey and current research challenges," *International Journal of Soft Computing and Engineering (IJSCE)*, vol. 2, no. 1, pp. 185–193, 2012.
- [109] R. Srikant and R. Agrawal, "Mining quantitative association rules in large relational tables," in *Proceedings of ACM SIGMOD International Conference on Management of Data*, pp. 1–12, 1996.
- [110] M. J. Zaki, "Spade: An efficient algorithm for mining frequent sequences," *Machine Learning*, vol. 42, no. 1-2, pp. 31–60, 2001.
- [111] J. Ayres, J. Flannick, J. Gehrke, and T. Yiu, "Sequential pattern mining using a bitmap representation," in *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 429–435, 2002.
- [112] J. Just, "A short survey of web data mining," in *Proceedings of Contributed Papers*, pp. 59–62, 2013.
- [113] P.-E. Danielsson, "Euclidean distance mapping," *Computer Graphics and Image Processing*, vol. 14, no. 3, pp. 227 – 248, 1980.
- [114] S. Tata and J. M. Patel, "Estimating the selectivity of tf-idf based cosine similarity predicates," *ACM SIGMOD Record*, vol. 36, no. 2, pp. 7–12, 2007.
- [115] E. Hellinger, *Neue Begründung der Theorie quadratischer Formen von unendlichvielen Veränderlichen*. Walter De Gruyter Incorporated, 1909.
- [116] S. Kullback and R. A. Leibler, "On information and sufficiency," *Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79–86, 1951.
- [117] D. Gusfield, *Algorithms on Strings, Trees, and Sequences - Computer Science and Computational Biology*. Cambridge University Press, 1997.
- [118] L. Birnbaum and G. Collins, *Machine Learning Proceedings*. Morgan Kaufmann, 1991.

- [119] Mozenda, “Mozenda.” <http://www.mozenda.com/default>.
- [120] Web-Info-Extractor, “Web info extractor.” <http://www.WebinfoExtractor.com>.
- [121] Web-Text-Extractor, “Web text extractor.” <http://www.webinfoextractor.com/index.htm>.
- [122] Web-Data-Extractor, “Web data extractor.” <http://www.WebExtractor.com/>.
- [123] Web-Content-Extractor, “Web content extractor.” <http://www.newprosoft.com/Web-content-Extractor.htm>.
- [124] Wget, “Gnu wget.” <http://www.gnu.org/software/wget/>, 1996.
- [125] S. Deepti and C. Sonal, “Web content mining techniques: a study,” *International Journal of Innovative Research in Technology and Science (IJIRTS)*, vol. 2, no. 3, pp. 67–72, 2014.
- [126] A. G. Jivani, “A comparative study of stemming algorithms,” *International Journal of Computer Technology and Applications*, vol. 2, no. 6, pp. 1930–1938, 2011.
- [127] A. Schenker, H. Bunke, M. Last, and A. Kandel, *Graph-Theoretic Techniques for Web Content Mining*. World Scientific, 2005.
- [128] I. Santos, J. De-la Peña-Sordo, I. Pastor-López, P. Galán-García, and P. G. Bringas, “Automatic categorisation of comments in social news websites,” *Expert System with Application*, vol. 39, pp. 13417–13425, 2012.
- [129] T. Xia, Y. Chai, H. Lu, and T. Wang, “Vector space model based internet contextual advertising,” in *International Conference on Management of e-Commerce and e-Government (ICMeCG)*, pp. 301–304, 2012.
- [130] M. Christian and B.-Y. Ricardo, “A comparison of open source search engines,” technical report, 2007.
- [131] S. E. Robertson and K. S. Jones, “Relevance weighting of search terms,” *Journal of American Society for Information Science (JASIS)*, vol. 27, no. 3, pp. 129–146, 1976.
- [132] KEA, “Key phrase extraction algorithm (kea).” <http://www.nzdl.org/Kea/index.html>.
- [133] Keyword-Analysis-Tool, “Keyword analysis tool.” <http://seokeywordanalysis.com/>.
- [134] Keyword/Terminology-Extractor, “Text api: Keyword / terminology extraction.” <http://www.alchemyapi.com/api/keyword/>.
- [135] Maui, “Maui.” <http://maui-indexer.appspot.com/>.

- [136] Topia-Term-Extractor, “Topia term extractor.” <https://pypi.python.org/pypi/topia.termextract/>.
- [137] Yahoo-Term-extractor, “Yahoo term extractor.” <http://www.programmableweb.com/api/yahoo-term-extraction>.
- [138] D. M. Blei, “Probabilistic topic models,” *Communications of the ACM*, vol. 55, no. 4, pp. 77–84, 2012.
- [139] J. Grimmer, “A bayesian hierarchical topic model for political texts: Measuring expressed agendas in senate press releases,” *Political Analysis*, vol. 18, no. 1, pp. 1–35, 2010.
- [140] S. Gerrish and D. M. Blei, “A language-based approach to measuring scholarly impact,” in *International Conference on Machine Learning*, pp. 375–382, 2010.
- [141] R. Socher, S. Gershman, A. Perotte, P. Sederberg, D. Blei, and K. Norman, “A bayesian analysis of dynamics in free recall,” in *Advances in Neural Information Processing Systems*, pp. 1714–1722, 2009.
- [142] K. Pargfrieder, *Interorganizational Workflow Management - Concepts, Requirements and Approaches*. GRIN Verlag, 2013.
- [143] D. Sánchez, M. Batet, D. Isern, and A. Valls, “Ontology-based semantic similarity: A new feature-based approach,” *Expert System with Application*, vol. 39, no. 9, pp. 7718–7728, 2012.
- [144] R. Thiagarajan, G. Manjunath, and M. Stumptner, “Computing semantic similarity using ontologies,” *the International Semantic Web Conference (ISWC)*, pp. n/a–n/a, 2008.
- [145] L. Ding, T. Finin, A. Joshi, R. Pan, R. S. Cost, Y. Peng, P. Reddivari, V. Doshi, and J. Sachs, “Swoogle: a search and metadata engine for the semantic web,” in *Proceedings of the thirteenth ACM international conference on Information and knowledge management*, pp. 652–659, 2004.
- [146] A. Valls, K. Gibert, D. Sánchez, and M. Batet, “Using ontologies for structuring organizational knowledge in home care assistance,” *I. J. Medical Informatics*, vol. 79, no. 5, pp. 370–387, 2010.
- [147] C. C. Aggarwal, *Data Classification: Algorithms and Applications*. CRC Press/Taylor & Francis Group, 2014.
- [148] S. De Ascaniis, N. Bischof, and L. Cantoni, “Building destination image through online opinionated discourses. the case of swiss mountain destinations,” *Information and Communication Technologies in Tourism (ENTER)*, pp. 94–106, 2013.
- [149] R. Ballantyne, K. R. Hughes, and W. Brent, “Meeting the needs of tourists: The role and function of australian visitor information centers,” *Journal of Travel and Tourism Marketing*, vol. 26, pp. 778–794, 2009.

- [150] E. Marchiori, P. Milwood, and F. Zach, “Drivers and benefits of analysing dmos’ e wom activities,” *Information and Communication Technologies in Tourism (ENTER)*, vol. 1, pp. 107–118, 2013.
- [151] B. Pan and D. R. Fesenmaier, “Travel information search on the internet: a preliminary analysis,” *In Proceedings of the International Conference in Helsinki*, pp. 242–251, 2003.
- [152] C.-I. Hsu, M.-L. Shih, B.-W. Huang, B.-Y. Lin, and C.-N. Lin, “Predicting tourism loyalty using an integrated bayesian network mechanism,” *Expert Systems with Applications*, vol. 36, no. 9, pp. 11760 – 11763, 2009.
- [153] E. Turban and D. Gehrke, “Determinants of e-commerce website,” *Human Systems Management*, vol. 19, pp. 111–120, 2000.
- [154] D. Chaffey, F. Ellis-Chadwick, K. Johnston, and R. Mayer, *Internet Marketing*. Prentice Hall/Financial Times, 2006.
- [155] U. Gretzel, “Intelligent systems in tourism: A social science perspective,” *Annals of Tourism Research*, vol. 38, no. 3, pp. 757 – 779, 2011.
- [156] M. Abou-Shouk, W. M. Lim, and P. Megicks, “Internet adoption by travel agents: a case of egypt,” *International Journal of Tourism Research*, pp. n/a–n/a, 2012.
- [157] A. Luberg, P. Järv, and T. Tammet, “Information extraction for a tourist recommender system,” *Information and Communication Technologies in Tourism (ENTER)*, pp. n/a–n/a, 2012.
- [158] L. Cao, J. Luo, A. C. Gallagher, X. Jin, J. Han, and T. S. Huang, “A worldwide tourism recommendation system based on geotagged web photos,” *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 2274–2277, 2010.
- [159] H. Berger, M. Denk, M. Dittenbach, A. Pesenhofer, and D. Merkl, “Photo-based user profiling for tourism recommender systems,” *In Proceedings of the 8th International Conference on Electronic Commerce and Web Technologies*, pp. 46–55, 2007.
- [160] P. Brejla and D. Gilbert, “An exploratory use of web content analysis to understand cruise tourism services,” *International Journal of Tourism Research*, pp. n/a–n/a, 2012.
- [161] CLF, “Common log format (clf).” The World Wide Web Consortium (W3C): <http://www.w3.org/Daemon/User/Config/Logging.html#common-logfile-format>, 1995.
- [162] O. Arbelaitz, I. Gurrutxaga, A. Lojo, J. Muguerza, J. M. Pérez, and I. Perona, “Adaptation of the user navigation scheme using clustering and frequent pattern mining techniques for profiling,” in *Proceedings of the International Conference on Knowledge Discovery and Information Retrieval (KDIR)*, pp. 187–192, 2012.

- [163] H. Daqing and G. Ayse, “Detecting session boundaries from web user logs,” in *Proceedings of the BCS-IRSG 22nd Annual Colloquium on Information Retrieval Research*, pp. 57–66, 2000.
- [164] D. Pierrakos, G. Paliouras, C. Papatheodorou, and C. D. Spyropoulos, “Web usage mining as a tool for personalization: A survey,” *User Modeling and User-Adapted Interaction*, vol. 13, no. 4, pp. 311–372, 2003.
- [165] P. Boldi and S. Vigna, “Mg4j at trec 2006,” in *TREC* (E. M. Voorhees and L. P. Buckland, eds.), National Institute of Standards and Technology (NIST), 2006.
- [166] S. Dasarathy, *Nearest neighbor (NN) norms : NN pattern classification techniques*. IEEE Computer Society Press, 1991.
- [167] O. Arbelaitz, I. Gurrutxaga, J. Muguerza, J. M. Pérez, and I. Perona, “An extensive comparative study of cluster validity indices,” *Pattern Recognition*, vol. 46, no. 1, pp. 243–256, 2013.
- [168] O. Arbelaitz, I. Gurrutxaga, A. Lojo, J. Muguerza, J. Pérez, and I. Perona, “Adaptation of the user navigation scheme using clustering and frequent pattern mining techniques for profiling,” *4th International Conference on Knowledge Discovery and Information Retrieval (KDIR)*, pp. 187–192, 2012.
- [169] D. Ramage and E. Rosen, “Stanford topic modeling toolbox(stmt).” <http://nlp.stanford.edu/software/tmt/tmt-0.2/>, 2009.
- [170] S. P. Lloyd, “Least squares quantization in pcm,” *IEEE Transactions on Information Theory*, vol. 28, pp. 129–137, 1982.
- [171] A. Dillon, “Beyond usability: process, outcome and affect in human computer interactions.,” *Canadian Journal of Information Science*, vol. 26, no. 4, pp. 57–69, 2001.
- [172] P. B. J. Craven, “Nonvisual access to the digital library: The use of digital library interfaces by blind and visually impaired people,” Technical report 145, Centre for Research in Library and Information Management, Manchester, United Kingdom, Manchester, UK, 2013.
- [173] S. Chandrashekar, T. Stockman, D. Fels, and R. Bedyk, “Using think aloud protocol with blind users: A case for inclusive usability evaluation methods,” in *Proceedings of the 8th International ACM SIGACCESS Conference on Computers and Accessibility*, pp. 251–252, 2006.
- [174] H. Petrie and N. Bevan, “The evaluation of accessibility, usability and user experience,” in *The Universal Access Handbook*,, 2009.
- [175] F. Pucillo and G. Cascini, “A framework for user experience, needs and affordances,” *Design Studies*, vol. 35, no. 2, pp. 160 – 179, 2014.

-
- [176] P. J. Standen, R. B. Karsandas, N. Anderton, S. Battersby, and D. J. Brown, “An evaluation of the use of a computer game in improving the choice reaction time of adults with intellectual disabilities,” *Journal of Assistive Technologies*, vol. 3, no. 4, pp. 4–11, 2009.
- [177] M. Vigo and S. Harper, “Challenging information foraging theory: Screen reader users are not always driven by information scent,” in *Proceedings of the 24th ACM Conference on Hypertext and Social Media*, pp. 60–68, 2013.
- [178] ONCE, “Fundación once for cooperation and social inclusion of people with disabilities.” <http://www.fundaciononce.es/EN/Pages/Portada.aspx>, 2014.
- [179] EGOKITUZ, “Evalaccess 2.0: Web service tool for evaluating web accessibility.” <http://s ipt07.si.ehu.es/evalaccess2/>, 2010.
- [180] D. Blei, A. Ng, and M. Jordan, “Latent dirichlet allocation,” *The Journal of Machine Learning Research*, vol. 3, pp. 993–1022, 2003.
- [181] D. M. Blei, “Introduction to probabilistic topic models,” *Communications of the ACM*, pp. n/a–n/a, 2011.
- [182] Lemur-project, “Lemur toolkit homepage.” <http://www.lemurproject.org/>.

Part VI
Appendices

Appendix A

Web content extractor study

A.1 Web content extractor tools analysis

In this appendix, we will present the main features of the analysed web content extractor tools that are mentioned in the Chapter 4 in the Section 4.2.1.1. Concretely we have analysed the following systems: Mozenda, Web Info Extractor, Web Text Extractor, Web Data Extractor, Web Content Extractor and Wget crawler.

The analysis was performed using the same example web page in all tools, so that input/output options and offered results can be compared. We used concretely the web page of the University of the Basque Country (www.ehu.es/p200-home/eu/).

A.1.1 Mozenda

URL: <http://www.mozenda.com/default>

Mozenda is a software that enables users of all types to easily extract and manage web data. With Mozenda, users can set up agents that routinely extract data, store data, and publish data to multiple destinations. Once information is in the Mozenda systems users can format, re-purpose, and mash-up the data to be used in other online/offline applications or as intelligence. All data in the Mozenda system is secure and is hosted in class A data warehouses but can be accessed over the web securely via the Mozenda web console.

Mozenda works only under Windows operating system and it can only be used with Windows XP, Windows Vista and Windows 7. Furthermore, it accepts an URL as input.

Apart from that, these are the main characteristics of the system:

- It does not perform a recursive analysis of the pages linked to the provided URL.
- It requires to select online (with the mouse) the kind of information the user wants to be provided with: figures, e-mails... That is not an automatic tool.
- The output of the system is a list containing all the information of the selected type in the provided URL (see Figure A.1).
- The user can select the format of the output information between: CSV, TSV, XML or in an Excel file. Therefore, Mozenda is capable to send the result by email.

Figure A.1 shows the appearance of the tool. Mozenda is not a free system but it offers a demo of 14 days to try the product. The only restriction is that users can analyse only 500 pages and 100 images. There are many different products offered by Mozenda and the price depends on the selected option: if we choose the standard the price is \$99 per month, whereas if we choose the professional option the price is \$199 per month and the last option is the enterprise option and the price is \$ 799 per year.



Figure A.1: Appearance of the Mozenda tool.

A.1.2 Web Info Extractor

URL: www.WebinfoExtractor.com

Web Info Extractor is a powerful tool for content extraction and content analysis. It can extract structured or unstructured data from web pages, reformat it into local files or save it to databases, as well as post it to a web server.

Web Info Extractor works only under Windows operating system. It does not matter what version of Windows it is. Furthermore, it accepts an URL as input. This application performs a recursive analysis of the pages linked to the URL given. In addition the output of the system is a list containing the information of selected type (text, image, links ... see Figure A.2)

These are the main characteristics of the system:

- It is able to show the output in different formats such as: CSV, in a text file or in a database (access, MySQL, SQLServer).
- It extracts structured and unstructured data to different types of files.
- It is able to monitor web pages and extract new content when updated.
- It can deal with text, image and link files.
- It can deal with web pages in all languages.
- It runs multiple tasks at the same time. If the users give to the program more than one URL it can support to run all of them at the same time.

There are more than one price depending on the limitations of the system, the basic licence's price is 99.95 and full likeness's price is 499.95.

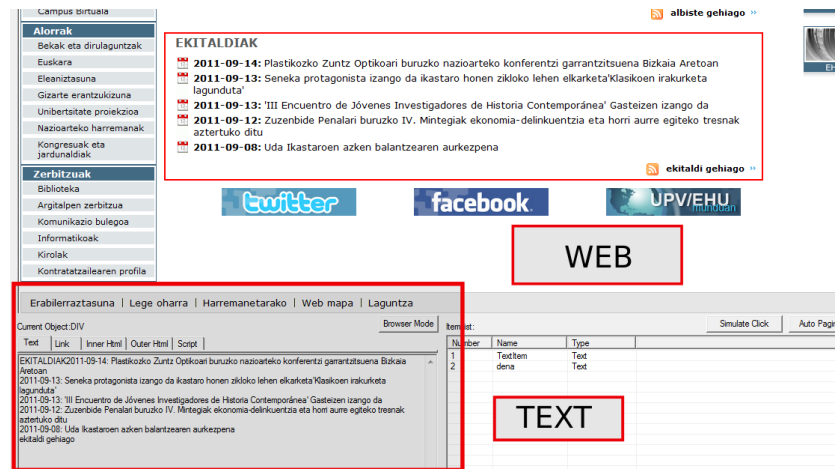


Figure A.2: Appearance of the Web Info Extractor tool.

A.1.3 Web Text Extractor

URL: <http://www.webinfoextractor.com/index.htm>

Web Text Extractor is designed for extracting text from web. The user can extract and copy these texts without selecting them. This tool can be the simplest one in the area of web content extraction. It has a simple and intuitive interface, but it is used only to extract little information. This tool works only under Windows operating system. Moreover the output is the textual content of the browse page. It offers the output in a text file (see Figure A.3) These are the main characteristics:

- It only needs Internet connection to browse the page that the user is interested in.
- It does not provide a recursive analysis.
- It requires moving the cursor of the mouse from the system to the text the user wants to be provided with.

Nevertheless Web Text Extractor is not a free system. However, it has a demo in order to try the system. The price for the program is very cheap \$19.95.

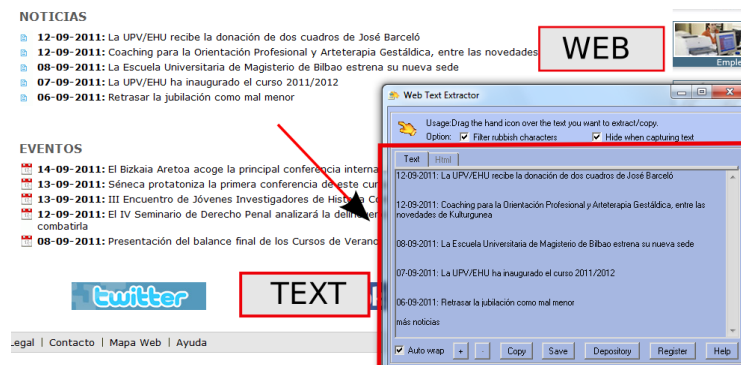


Figure A.3: Appearance of the Web Text Extractor tool.

A.1.4 Web Data Extractor

URL: <http://www.WebExtractor.com/>

This tool is a powerful web data and link extractor utility. It extracts URL, meta tag (title, keyword etc.), body text, email, phone, fax from websites. Web Data Extractor has numerous filters to restrict sessions, such as URL filter, date modified, file size, etc. It allows user-selectable recursion levels, retrieval threads, time-out, proxy support and many other options.

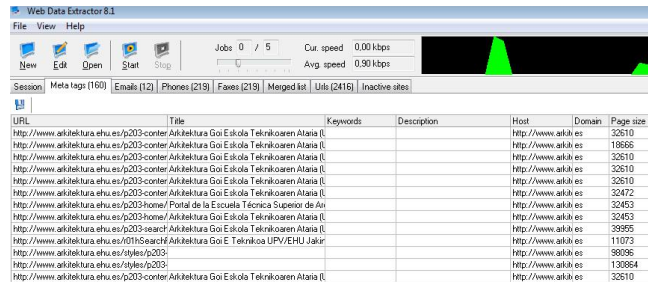
This application works only in Windows operating system. Furthermore, It needs a URL or a list of URLs as an input. Apart from that it performs a recursive analysis. The system is able to extract the information about the pages linked to the provided URL and URL list.

Web Data Extractor characteristics are the following ones:

- It offers the option to select the information that the users need. For instance, the body of a web page, emails, fax numbers...
- The output of the Web Data Extractor is a list containing all the information previously selected. The results can be all the textual content of the provided URL and the URLs linked with it (if this is what the users have selected previously) (see Figure A.4).
- It offers the information extracted in different formats such as: extracted in CSV format, in TXT format, in HTML format and in Excel format.

- It has a negative point, sometimes when the system tries to extract telephone numbers; it confuses them with dates and returns a mixture of both.
- There are abundant opinions in the Internet about this system and all of them are positive.

The price depends on the type of the extraction program the user wants, these are the different options: URL, meta tag and body extractor, \$99.00; URL, email extractor, \$89.00; URL, phone/fax extractor, \$89.00; URL, meta tag, body extractor, email, phone and fax extractor, \$149.00; The update of an old version, \$49.00.



URL	Title	Keywords	Description	Host	Domain	Page size
http://www.arkitekтура.edu.es/p203-conter/Arkitekтура Goi Eskola Teknikoaren Ataria (http://www.arkitekтура.edu.es	arkitekтура.edu.es	32610
http://www.arkitekтура.edu.es/p203-conter/Arkitekтура Goi Eskola Teknikoaren Ataria (http://www.arkitekтура.edu.es	arkitekтура.edu.es	18666
http://www.arkitekтура.edu.es/p203-conter/Arkitekтура Goi Eskola Teknikoaren Ataria (http://www.arkitekтура.edu.es	arkitekтура.edu.es	32610
http://www.arkitekтура.edu.es/p203-conter/Arkitekтура Goi Eskola Teknikoaren Ataria (http://www.arkitekтура.edu.es	arkitekтура.edu.es	32610
http://www.arkitekтура.edu.es/p203-conter/Arkitekтура Goi Eskola Teknikoaren Ataria (http://www.arkitekтура.edu.es	arkitekтура.edu.es	32472
http://www.arkitekтура.edu.es/p203-home/Portal de la Escuela Técnica Superior de Ar				http://www.arkitekтура.edu.es	arkitekтура.edu.es	32463
http://www.arkitekтура.edu.es/p203-home/Arkitekтура Goi Eskola Teknikoaren Ataria (http://www.arkitekтура.edu.es	arkitekтура.edu.es	32463
http://www.arkitekтура.edu.es/p203-search/Arkitekтура Goi Eskola Teknikoaren Ataria (http://www.arkitekтура.edu.es	arkitekтура.edu.es	39995
http://www.arkitekтура.edu.es/071hsearch/Arkitekтура Goi Eskola Teknikoa UPV/EHU Jakar				http://www.arkitekтура.edu.es	arkitekтура.edu.es	11073
http://www.arkitekтура.edu.es/style/p203				http://www.arkitekтура.edu.es	arkitekтура.edu.es	38096
http://www.arkitekтура.edu.es/p203-conter/Arkitekтура Goi Eskola Teknikoaren Ataria (http://www.arkitekтура.edu.es	arkitekтура.edu.es	13084
http://www.arkitekтура.edu.es/p203-conter/Arkitekтура Goi Eskola Teknikoaren Ataria (http://www.arkitekтура.edu.es	arkitekтура.edu.es	32610

Figure A.4: Web Data Extractor tool's output layout.

A.1.5 Web Content Extractor

URL: <http://www.newprosoft.com/Web-content-Extractor.htm>

Web Content Extractor is the most powerful and easy-to-use data extraction software for web scraping, web harvesting and data extraction from the Internet. Web Content Extractor offers the users a friendly, wizard-driven interface that will walk the user through the process of building a data extraction pattern and creating crawling rules in a simple point-and-click manner. Not a single string of code is required, web data extraction is completely automatic.

Web Content Extractor can be used only with Windows operating system. In addition it accepts an URL or more as an input. Moreover, it is a recursive system.

These are the characteristics of the Web Content Extractor:

- It extracts data from multiple pages very fast, thanks to the multi-threaded crawling technology that downloads up to 20 threads simultaneously.
- The output is a list containing the selected information.
- The extracted data can be exported to a variety of formats, including Microsoft Excel (CSV), Access, TXT, HTML, XML, SQL script, MySQL script and to any ODBC data source. This variety of export formats allows users to process and analyse data in their customary format. (See Figure A.5).
- It can return a URL structure of the particular website.

Web Content Extractor is not a free system. But it has a 14 days trial in order to try the system. The price of this product is \$159.

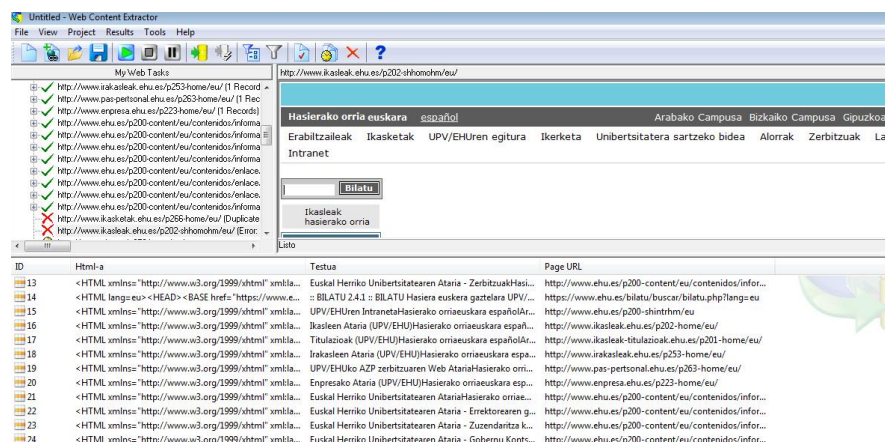


Figure A.5: Web Content Extractor tool's output layout.

A.1.6 Wget

GNU Wget (or just Wget, formerly Geturl) is a computer program that retrieves content from Web servers, and is part of the GNU Project. Its name is derived from World Wide Web and get. GNU Wget is a free software package for retrieving files using the most widely-used Internet protocols. It

is a non-interactive command line tool, so it may easily be called from scripts, cron jobs, terminals without X-Windows support, etc.

Written in portable C, Wget can be easily installed on any Unix-like system and has been ported to many environments, including Microsoft Windows, Mac OS X, OpenVMS, HP-UX, MorphOS and AmigaOS.

Its features include:

- Wget is non-interactive, meaning that it can work in the background, while the user is not logged on. This allows the user to start a retrieval and disconnect from the system, letting Wget finish the work.
- Wget can follow links in HTML, XHTML and CSS pages, to create local versions of remote sites, fully recreating the directory structure of the original site. So, it is a recursive downloading tool.
- Robustness. Wget has been designed for robustness over slow or unstable network connections. If a download does not complete due to a network problem, Wget will automatically try to continue the download from where it left off, and repeat this until the whole file has been retrieved.
- File name wild-card matching and recursive mirroring of directories are available. Wget can read the time-stamp information given by servers, and store it locally. Thus Wget can see if the remote file has changed since last retrieval, and automatically retrieve the new version if it has.
- Wget supports proxy servers.
- It can resume aborted downloads.
- Most of the features are fully configurable, either through the command line options, or via the initialization file.

Wget is a free software. This means that everyone could use it, redistribute it and/or modify it under the terms of the GNU General Public License, as published by the Free Software Foundation.

Appendix B

Search engine study

B.1 Search engine tools analysis

In this appendix, we will present the main features of the analysed search engines that are mentioned in the Chapter 4 in the Section 4.4.1.1. The search engines that are going to be presented are MG4J and Indri.

B.1.1 MG4J (Managing Gigabytes for Java)

MG4J is a full text indexer for large collection of documents written in Java, developed at the University of Milano, Italy. The main points of MG4J are:

- Powerful indexing. Support for document collections and factories makes it possible to analyse, index and query consistently large document collections, providing easy-to-understand snippets that highlight relevant passages in the retrieved documents.
- Efficiency. MG4J can index and scales to hundreds of millions of documents.
- Multi-index interval semantics. When the user submit a query, MG4J returns, for each index, a list of intervals satisfying the query. This provides the base for several high-precision scorers and for very efficient implementation of sophisticated operators. The intervals are built in linear time using new research algorithms.
- Expressive operators. MG4J goes far beyond the bag-of-words model, providing efficient implementation of phrase queries, proximity restrictions, ordered conjunction, and combined multiple-index queries.

- Flexibility. The user can build much smaller indices by dropping term positions, or even term counts. Several different types of codes can be chosen to balance efficiency and index size. Documents coming from a collection can be renumbered (e.g., to match a static rank or experiment with indexing techniques).
- Openness. The document collection/factory interfaces provide an easy way to present the users' data representation to MG4J, making it a breeze to set up a web-based search engine accessing directly your data. Every element along the path of query resolution (parsers, document-iterator builders, query engines, etc.) can be substituted with the users version.
- Distributed processing. Indices can be built for a collection split in several parts, and combined later. Combination of indices allows non-contiguous indices and even the same document can be split across different collections (e.g., when indexing anchor text).
- Multithreading. Indices can be queried and scored concurrently.
- Clustering. Indices can be clustered both lexically and documentally (possibly after a partitioning). The clustering system is completely open, and user-defined strategies decide how to combine documents from different sources. This architecture makes it possible, for instance, to load in RAM the part of an index that contains terms appearing more frequently in user queries.

MG4J has some negative points; firstly the format of the queries has this appearance: "word OR word OR word". Therefore, the text need to be processed to be converted to that format. Secondly, this system does not have the option to make multiple queries at a time, so to use it for comparing a set of texts (which can become more than 3000) a java program needs to be designed to make queries automatically. The positive point is that it returns similarity values from 0 to 514, so it would be very easy for normalizing.

B.1.2 Indri

Indri is a search engine built on top of the Lemur [182] project, which is a tool-kit designed for research in language modelling and information retrieval. This project was developed by a cooperative work between the University of Massachusetts and Carnegie Mellon University, in the USA.

Below the main points of Indri are going to be presented:

- **Flexibility.** It supports popular structured query operators from IN-QUERY. Moreover, it is an open source tool. In addition, it can parse PDF, HTML and XML, Word and PowerPoint (Windows only) documents.
- **Usability.** It supports UTF-8 encoded text and language independent tokenization of UTF-8 encoded documents. It includes both command line tools and a Java user interface. Furthermore, the API can be used from Java, PHP, or C++ and it works on Windows, Linux, Solaris and Mac OS X.
- **Powerful indexing.** It can be used on a cluster of machines for faster indexing and retrieval. It scales to terabyte sized collections.

To use this search engine, it is necessary to follow only two instructions, so it is a very easy search engine. It can receive as input more than one query, so it is not need to write any java program as in MG4J. The negative point of Indri is that the similarity values go from 0 to negative infinite. So it would be very difficult to normalize the values.

Appendix C

Keyword extractor study

C.1 Automatic keyword extractor tools

In this appendix, we will present the main features of the analysed keyword extractor tools. We have analysed the following systems: Kea (Key phrase extraction algorithm), Keyword Analysis Tool, Keyword/Terminology Extractor, Maui, Topia Term Extractor and Yahoo Term Extractor.

C.1.1 Kea (Key phrase extraction algorithm)

URL: <http://www.nzdl.org/Kea/index.html>

KEA is an algorithm for extracting key phrases from text documents. It is implemented in Java and it is open-source software distributed under the GNU (General Public License).

It works in Windows, Linux and Mac Os operating systems. To make it work, first a KEA key phrase extraction model needs to be built from a set of documents (preferably from the same domain) and it needs as well the keywords related to texts. Once we have this model structured we are able to try the application.

C.1.2 Keyword Analysis Tool

URL: <http://seokeywordanalysis.com/>

The purpose of this tool is to assist webmasters in performing keyword analysis. This tool will, when given a URL, display the most frequently used keywords and key phrases present in the document in question. This

application is used online.

To use this tool, the users simply need to type in the URL to analyse, and press the "Analyse Page" button. Optionally, they can filter their analysis by entering specific keywords, or add or delete words from the provided stop word list. The default stop word list used by this tool is the list of most commonly used English language stop words. But this list can be changed to another language. Once keywords and key phrases are displayed, they can click on a keyword/phrase to see text passages in the document containing the keyword(s) in question. This is helpful in showing the context in which the word or phrase is used. They can also generate a custom research report on the term to see how it is used elsewhere on the Internet. The following layout is the appearance of the software (Figure C.1):

Hoskinson.net > Keyword Analysis Tool: Advanced Keyword and Keyphrase Extraction Technology for Content Analysis and Search Engine Optimization (SEO)

[Try our [Keyword Library](#) | [Add the SEO Keyword Analysis Tool](#) to your [Google Tool Bar](#)]

URL:

Keywords (Optional):

Stop Words (Optional):

Figure C.1: Interface of Keyword Analysis Tool.

Although this tool has the advantages of including a stop word list useful for automatically reject the most common words from the texts, it is not an automatic software due to the fact that all texts need to be typed one by one.

C.1.3 Keyword/Terminology Extractor

URL: <http://www.alchemyapi.com/api/keyword/>

This software is capable of extracting topic keywords from HTML, text, or web-based content. It employs sophisticated statistical algorithms and Natural Language Processing technology to analyse the data for extracting keywords. This application has an online demo.

Keyword/Terminology Extractor is supported in over a half-dozen different languages, enabling even foreign- language content to be categorized

and tagged.

To use this tool, there are two different options: to type the URL to analyse and to copy the text in the scroll bar. Once this has been done the submit button needs to be pressed and it returns the keywords of the text or URL given.

The following layout is the appearance of the software (Figure C.2):

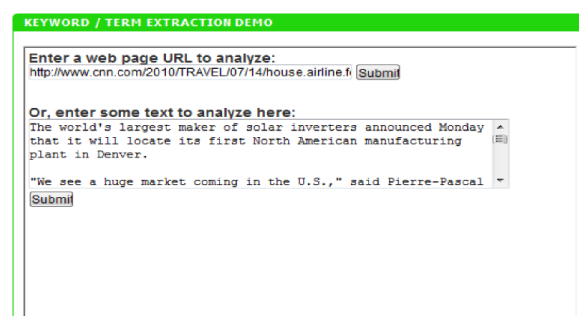


Figure C.2: Interface of Keyword/Terminology Extractor.

Even if it has positive points, the application is not prepared to work with document collections.

C.1.4 Maui

URL: <http://maui-indexer.appspot.com/>

Maui automatically identifies main topics in text documents. Depending on the task, topics are tags, keywords, key phrases, vocabulary terms, descriptors, index terms or titles of wikipedia articles. A demo can be extracted for testing the usability of this tool. The demo shows how a vocabulary can be used to derive the topics, e.g. High Energy Physics thesaurus or the agricultural thesaurus Agrovoc. It also shows how key phrases can be extracted from document text. To use this tool, first a text needs to be typed (or copy it from other location) or uploaded (in a pdf or word format). Then, a vocabulary needs to be selected. Figure C.3 shows the appearance of the tool analysed.

This application is not prepared to work with document collections and it requires to introduce a vocabulary.

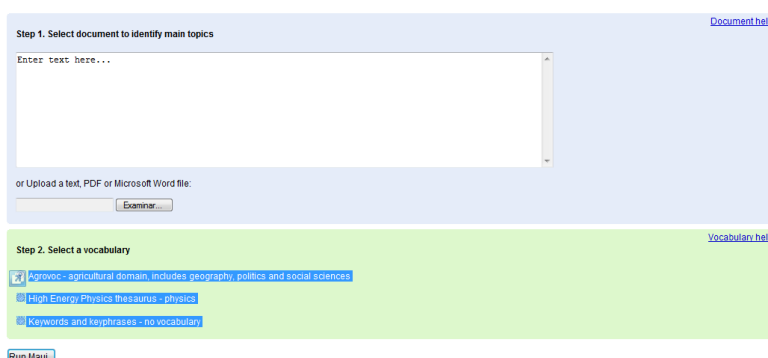


Figure C.3: Interface of Maui.

C.1.5 Topia Term Extractor

This package determines important terms within a given piece of content. It uses linguistic tools such as Parts-Of-Speech (POS) and some simple statistical analysis to determine the terms and their strength. Topia Term Extractor is a python term extractor.

Although in the documentation it says that this is a very valuable tool, it is not intuitive enough to be used and it is not prepared for letting user modify the tool neither to use it for a document collection.

C.1.6 Yahoo Term Extractor

The Yahoo Term Extractor allows users to perform content analysis by providing a list of significant words or phrases extracted from a larger content. It is one of the technologies used in Yahoo search engine. It uses an API and responses are formatted in XML.

Even if Yahoo Term Extractor is prepared to extract the keywords or keyphrases of a single text, it is easy to use it for a collection or set of documents due to the fact that it is a free source C# program. Consequently, the users can modify the program according to their necessity.

As a result, the input can be the text of a single document or the content of a set of documents (in case the user modifies the program) and as output it returns a set of relevant words for each processed text.

