Using Data Mapping and Facebook Data to Enrich Web Browsing

By

Yanjun Liu

Dissertation

Presented to the

University of Dublin, Trinity College

in fulfilment

of the requirements

for the Degree of

Master of Science in Computer Science

Supervisor: Declan O'Sullivan

Assistant Supervisor: Rob Brennan

Declaration

I, the undersigned, declare that this work has not been previously submitted as an exercise for a degree at this or any other University, and that, unless otherwise stated, it is entirely my own work.

Yanjun Liu

August 2013

Permission to Lend or Copy

I, the undersigned, agree to deposit this dissertation in the University's open access institutional repository or allow the library to do so on my behalf, subject to Irish Copyright Legislation and Trinity College Library conditions of use and acknowledgement.

Yanjun Liu

August 2013

Acknowledgements

I would like to express my special thanks and gratitude to my supervisor Dr. Declan O'Sullivan and my assistant supervisor Dr. Rob Brennan for giving me the golden opportunity to do this research, for their support, guidance and help to the dissertation. Without their help, the research would not have been possible. I also would like to thank Dr. Dave Lewis for his precious suggestions. Additionally I would like to express my appreciation to all the participants for helping me to do the experiment for the dissertation. Finally, I would like to thank my husband Dr. Hui Song for his support and help throughout this research process.

Abstract

Social data is playing a more and more important role in recommender systems, to improve the quality of suggestions and to enrich the recommended information within the social network systems. However, as people spend more and more time on the web, there is a requirement that bringing this data out of those social systems and letting it serve users when they are browsing on external websites. The difficulty is how to establish the relations between social data and the web content a user is browsing, which are annotated by different schema languages. The dissertation proposes a two phase mapping approach to tackle the data matching problem. In the schema mapping phase, the Alignment API and the Wordnet library are used to calculate the similarity between the data types defined by different schema languages, as well as the matched properties between the similar types. In the instance mapping phase, instances extracted from original websites of the shared social links are mapped to the web content based on the results computed in phase one. In order to adjust this approach to multiple websites including those with self-defined subtypes, a component called strengthen model is designed to enrich the schema information of instances. The evaluation is carried out on the prototype called the Suggestion Tool which utilises Facebook data for the web enrichment. Evaluation results shows that the approach provides suggesting content effectively and with user satisfaction.

Table of Contents

List of Figures

Chapter 1. Introduction

This section firstly presents the motivation of the research project. Then, the research question and objectives are described. Finally, the contribution of the research is presented.

1.1 Motivation

Recommender systems assist the natural social process of people getting suggestions from other people through various mediums, such as book reviews, letters, newspapers or the word of mouth [1]. Those suggestions aim to present users with products such as books, music and movies according to their tastes. With the help of recommender systems, users can survive the huge and ever-expanding amount of web data, and get what they need efficiently.

Social networks (SN) are a promising source of data to enhance recommendations [2]. With the development of SN services, more and more social circles are created and maintained in SN platforms. Those circles are either images of the relations between people in the real world, or virtual relations between users with some kind of commonality. Information about individuals, their preferences, experiences and relationships are embedded in those circles and gathered by SN services like Facebook[1]. People in such social circles are influenced by each other, and also are likely to pay attention to what people in their relations are interested in [3]. Suggestions from those people are more important and credible than that from strangers [4]. This reflects a relationship between trust and recommendations that have been shown in Golbeck's research [5]. Recommender systems are also attracted by this relation information. Some of the recommender systems already utilize social relations to enhance the effectiveness of results filtering, for examples: FilmTrust [5] and MovieLens [6].

People spend several hours per day surfing the Internet, but only a small percentage is spent on social network. For example, in the U.S., people spent 98 hours per month online [7], and 7.6 hours per month on social networking on average [8]. Thus it would be useful to use SN data to recommend relevant

[1] www.facebook.com

information when users are browsing data outside the SN service. Imagine this situation: Bob is skimming the introduction of a book from an online book store, and he is not sure if the book really suits him. At this time, he notices the recommender system that shows that his friend Alice shares the same book, and he often agrees with Alice’s tastes. Now, he does not need to hesitate to click the button ‘Buy’.

In recent years, driven by the usage of search engines, more than 16.8% of webpages have been annotated by structural markup languages such as schema.org[2] [9] and Open Graph Protocol[3]. Relying on this markup, the semantics of webpages can be interpreted by machines. This gives an opportunity that when users are browsing these schema-annotated websites, a recommender system can analyse the web content automatically to use its semantic information, such as what kind of object it is about, and the key features of this object.

A key challenge is how to filter the social data and give appropriate suggestions whilst a user is browsing. The main difficulty in addressing this challenge is caused by the mismatch between data structures and semantics of webpage content and social data. In particular, as data from different sources (e.g., inside and outside a SN platform) are usually annotated by different markup languages, even if they are semantically equivalent, their formats are different. Moreover, different social data, e.g., Facebook graph data, often links to multiple websites, which adds complexity to the structure of social data. Can we find a way to solve the mismatching problem? Fortunately, data mapping techniques like ontology mapping are developed for years in order to create relations between two data sets [10]. This provides a promising foundation to establish the relations between social data and the browsing web content. However, how to design the proper mapping method for this specific domain is still a challenge.

In summary, it is the argument in this dissertation that social network services have very rich data, and those data can be helpful for providing recommendations when users are browsing outside social networking platforms.

[2] http://schema.org/
[3] http://ogp.me/

How to effectively filter social data for offering suggestions forms the need to create relations between social data and the browsing web content.

1.2 Research Question

The research question is:

To what extent is it possible to efficiently and effectively leverage social network data to provide browsing recommendations based on webpage meta-data, in a manner that users are satisfied with?

Social network data refers to web content that a user shares to their friends. Webpage metadata refers to web content that is annotated by structural languages. If two sets of data are *defined under different type systems*, the data types are specified by different schemas. In this dissertation, *Mapping* means creating relations between social network data and the web content a user is browsing, or in other words, identifying to what extent two data are similar or even equivalent. *Efficiency* in this dissertation refers to users being satisfied of the time interval from when a new webpage is opened to the displaying recommendations that are relevant to the web content. *Effectiveness* in this dissertation refers to users agreeing that suggestions are relevant to the web content. User *satisfaction* means users think the system could provide useful information and they like using the system when they are browsing.

1.3 Research Objectives

The following objectives for the project were derived from the research question:

RO1: Study the background knowledge and survey the state of the art for the research area. This includes the knowledge of data mapping related techniques, schema.org, Open Graph Protocol, Facebook data extraction, recommender systems, personalised user interface and evaluation methods.

RO2: Design a method for using data mapping technique to tackle the problem of matching social data to user browsing web content.

RO3: Design a method that can analyse social data shared from multiple websites.

RO4: Design and implement a recommendation tool that utilizes the methods in **RO2** and **RO3**.

RO5: Evaluate effectiveness and user satisfaction of the tool. An experiment will be carried out to collect user study results.

1.4 Contribution

The major contribution of the project is using data mapping techniques, specifically ontology mapping, to solve the mismatch between the web content a user is browsing and the relevant social data. In particular, the author adopts a two-phase mapping approach, a schema mapping phase and an instance similarity phase. In schema mapping, the author leverages the alignment ontology mapping tool [11] to correlate similar data types with different ontology definitions, and enhance it by introducing WordNet-based synonym analysis [12]. From the schema mapping results, the instance similarity is calculated by weighted comparison of relevant fields. The approach is implemented as a recommender system called the Suggestion Tool. A set of user studies on effectiveness, efficiency and user satisfaction are performed to evaluate the Suggestion Tool.

1.5 Dissertation Overview

The rest of the dissertation is organized as follows:

Chapter two presents the background knowledge of the research.

Chapter three provides an overview and analysis of the state of the art in the research area.

Chapter four describes the design of the approach to providing recommendations based on the web content a user is browsing.

Chapter five describes the implementation of the Suggestion Tool.

Chapter six presents the project evaluation process, including the detailed experiment introduction, results and results analysis.

Chapter seven summarise the whole project and presents the conclusion and discusses the future work.

Chapter 2. Background

In this section the following topics are discussed: First of all, a brief overview of linked data concept is shown including its description vocabularies and the querying language. Then, Facebook open graph related information is introduced such as its content, application, format and a relation to the semantic web. Next, schema.org concept is introduced together with the presentation of its features and an introduction of an application sample. Following on from this, an overview of data mapping technique is shown focusing on the concept of data mapping and an introduction of some mapping tools. After that, the on-going research and sample applications for web personalization and recommender systems are presented.

2.1 Linked Data and Related Techniques

The term linked data refers to the process of publishing and linking structured web resources in different domains [13]. As the flourishing of the semantic web, more and more web data are published on the Internet. They cover diverse domains, such as countries, people and films. It uses Resource Description Framework [4] (RDF) vocabularies such as Friend of a Friend [5] (FOAF) and GoodRelations to represent domain information as object instances such as a person or a country. Related classes, properties for the instances are defined in RDF, RDF Schema (RDFS) or Web Ontology Language (OWL).

In order to meet the growing needs of storage of RDF resources, many repositories emerge, also known as triple stores [14]. SPARQL Protocol and RDF Query Language (SPARQL) is a querying language used to access data in those stores. With the help of SPARQL, data could be shared between different stores. The shared data on the web could be represented as a global unique id called Universal Resource Identifier (URI). It is usually embedded in the RDF files for giving a reference of other resources.

[4] http://www.w3.org/RDF/
[5] http://www.foaf-project.org/

Linked data techniques provide the technical foundation for semantic data publishing and sharing [13], and play an important role in the process of adding Facebook information to schema.org websites that will be discussed in section 2.

2.2 Facebook Open Graph

Social networks provide rich user information on the web, which covers several aspects such as user's experiences, preferences, family and friends. They also provide a way to combine this information together. As a result, from a start point, you can get the whole world around the user. With the help of this information, a complete user model could be constructed, which can be very useful for web personalization [15].

Facebook is one of the most popular social network services. It is estimated to have nearly 901 million users [16], which gives the opportunity to collect huge volume of user data from it. Facebook open graph is a new set of programming tool that let you put information in Facebook and get information out of Facebook. It helps the developer to create deep relation between their application and Facebook. The Open Graph Protocol [6] is used to structure information of web pages, which can be understood by Facebook search engine. It is a simple schema set that defines common objects, e.g., image, audio and music.

The graph API provides the entry for retrieving user data, and the retrieved data is structured in JSON format. For using this information in the semantic web, they have to be restructured to the semantic web format, as the author mentioned above, the RDF format.

2.3 Schema.org

Schema.org[7], as its name suggests, defines the schemas of real world objects. The term schema here refers to the definitions of object types. The definitions can be used in RDF vocabulary in the form of URIs and also can be used to extend object types from some ontology such as GoodRelations[8]. In order to

[6] http://opengraphprotocol.org/
[7] http://www.schema.org/
[8] http://wiki.goodrelations-vocabulary.org/Microdata

make web pages be easily understood by machines, web developers could use schema.org terms to markup web pages[9].

Traditional web pages are mainly built up in HTML format. HTML tags describe the display of information, but are silent regarding the semantics of the information. On the other hand, pure text itself is also hard to be understood by the search engines. As a result, data mining and NLP techniques have to deal with lots of difficulties. So, more and more companies are in favour of using schema sets to build up web sites, especially some e-business companies. They showcase their products on the web sites, which are easier to be understood by search engines, hence are easier to be found by customers. Now, billions of pages are using schema.org and many of them are commercial web sites to which search sensitivity can bring more benefits.

Compared to the Open Graph Protocol schema, the schemas defined by schema.org give more complete and detailed information, which is more meaningful to machines. Some web sites like IMDB[10] mix Open Graph Protocol and schema.org to cater to both Facebook and Search engines.

2.4 Data Mapping

The term data mapping refers to the work of relating two data sets. In practice, there are many domains, such as database schema mapping, XML schema mapping, ontology mapping and heterogeneous cases. Here the author emphasis on the ontology mapping, which is widely used in the semantic web. In particular, when Facebook open graph information is converted to RDF format, one can generate a map between the RDF and schema.org vocabulary.

As the application of ontologies is expanding, a single ontology is not enough to meet all possible needs. The requirement of using ontologies from different domains encourages the development of ontology mapping techniques. Ontology mapping related techniques could be categorised into alignment, articulation and merging of ontologies [10]. Ontology alignment of the ontologies O_1 and O_2 is

[9] www.imdb.com

similar to creating a common intermediate source O_0 from which both O_1 and O_2 maps. The combination of O_0 and the mapping is called articulation. For the term merging, as described in [10] "merging can be relaxed by taking the articulation of two sub-ontologies of O1 and O2 respectively, and defining the merged ontology O according to their articulation". From now on, the term data mapping or ontology mapping in this thesis means the term ontology articulation.

Data mapping techniques have been used in several areas. For instance, [17] build a knowledge searching system on the Simple Knowledge Organisation System[11] (SKOS), which maps learning resources in different domain, provides more classification parameter and more relevant searching results. The MAFRA Toolkit [18], which is being applied in the European project Harmonise, provides a graphical GUI and supports a semi-automate integration between resources.

Ontology mapping is difficult and labour intensive work. In order to improve the accuracy, efficiency and reduce human effort, a lot of mapping tools have been created. For example, the COMA++ system provides a graphical mapping interface, which assist users in completing data mappings through drawing lines between two ontologies [16]. The PROMPT system provides a useful log function which could replay the mappings in the log when users need to do updates. The COGZ system improved the PROMPT system by adding some convenient features such as filtering and navigation to help users to do the mapping. Finally, the tag based mapping tool, based on description words, is designed to meet the requirement of doing mapping by ordinary users. The author will discuss these tools in detail in section 3.

2.5 Web Personalization

As there are a growing number of web resources, it has been harder and harder to get information that fits users' needs. Consequently, the techniques for web personalization gains more attention, which aim to improve the user experience when they are searching or browsing. For instance, some researchers put their efforts on helping search engines to filter and order their results according to personal user model [19]. After filtering, the volume of results is

[11] http://www.w3.org/2004/02/skos/

narrowed and the relevance is improved. After ordering, searching results with higher priority will be presented at the top. With this improvement, users could save a lot of time and effort to deal with the searching result. The core idea for personalization is user modelling and adaptation. Modelling means the process of creating a user model, which presents the users' information and their preferences. In the most common case, a user model needs to be maintained and constantly improved [15]. Adaptation refers to the process of adjusting the UI content or layout according to the user model.

Meanwhile, web personalization has been applied in several areas. For instance, many adaptive systems emerge such as the open corpus adaptive educational system [15] and the e-commerce recommender system [20]. The former helps users to get materials conformity to their will. The later could recommend products that may get more satisfaction from users. In this dissertation, personalisation is used to provide external information from Facebook that is related to the current user browsing content on e-commerce websites.

2.6 Recommender Systems

As the growing size of volume of web content, it has become harder and harder for users to get their needs at the right time. Recommender systems are considered a response to the information overload problem [21]. Recommender systems filter the web content and provide users with personalized information.

Many industries have started applying recommender systems, such as social media [5] [22] and e-commerce [23]. The history of purchases, rating, reviews or user profile and social relations are used to provide recommendations. In general, recommender systems are categorised based on the type of filtering methods they use, which are so called content-based filtering and collaborative filtering. As its name suggests, content-based filtering gets relevant content according to the content information itself, such as its genre, actors etc. Collected user interests information is used to match those content using similarity computation algorithms such as cosine similarity based on pre-defined vectors [21] [24]. In collaborative filtering, item knowledge is not required. It utilizes social relation to enhance the filtering results, and user ratings histories are usually used. This dissertation utilises a content-based mechanism but suggestions are based on social relations and social activities. The suggestions are required to be relevant to the web content a user is browsing. The challenge is how to establish the relations between social data and the browsing web content.

Chapter 3. State of the Art

This chapter presents the state of the art of this research. It first presents an overview of the recommender systems that utilise social information for providing suggestions. Then, the research about Facebook information extraction and transformation is shown. Following on from this, mapping systems working manually or automatically are presented. Finally, interfaces of personalised systems are compared and analysed.

3.1 Recommender Systems based on Social Information

Recommender systems provide users with suggestions about products or services that fit their interests. These recommended items are usually filtered from a huge volume of data, and many filtering methods are designed to meet this requirement. Traditionally recommender systems are based on user profiling, information filtering and machine learning [2]. However, in some recent methods, social information begins to play an important role to enhance the relevancy of recommendations. This section presents three recommender systems that provide suggestions with the help of social information: Amazon.com utilises customer purchase records. Facebook utilises user profile information and information of activities on Facebook and websites of third parties. The approach based on Epinions.com uses user relations generated by trust path. Another approach based on Flickr.com utilises user group and contacts information. In these approaches, social information is used in different ways, from the course grained way that regards all customers as a whole, to fine grained ones that differentiate the contributions of different customers based on the ranking or clustering of them.

3.1.1 All Customers as a Whole: Amazon

Amazon[12] is one of the most famous e-commerce systems in the world. It provides many kinds of products to millions of customers. In order to help users to get their needs (and therefore to create the opportunities to sell more products), Amazon deploys a recommender system.

[12] www.amazon.com

The recommendations are provided based on an item-to-item algorithm [23]. They build a product-to-product metric and iterate through each product pairs to calculate the similarity between each pair. There are two kinds of recommendations: One is provided according to the products in the customer's purchase history. It matches the current product to the product repository and finds similar items that have been bought by other customers. These products are shown on the home page as "Recommendations for You". The other is provided according to the item that the customer is currently browsing. Similar to the first method, it is also product-based. All the recommendations are shown as items that "Customers Who Bought This Item Also Bought". These products are shown in the same page as the current one.

To calculate the recommendation list, Amazon utilizes a filtering algorithm to build the similar items table. Two items are defined as similar, if there are many customers who have bought, or tend to buy the two items together. In the calculation, people are equally weighted, and there is no particular customer or customer group that should be considered to be more similar to the purchaser. The suggestions are firstly matched by their product attributes such as colour and size and finally ordered by the ratings provided by consumers.

3.1.2 User Profile Features & User Activities: Facebook

Facebook cooperates with third parties to provide users recommendations [25]. In order to suggest the right things to the right people, three parts of information are used to enhance the relevancy of recommendations. Firstly, user profile information such as location and gender is used. Secondly, user activities on Facebook are also helpful, for example, user shares and likes. Thirdly, user activities on websites of third parties are utilised, for example, user visit history, click history or purchase history.

Besides providing recommendations for third parties, some researches also focus on improving internal services of Facebook. Baatarjav et al. [26] designed an approach to providing group suggestions for users on Facebook. They used user profile information such as time zone, age, relationship status and number of friends as factors to categorise group members and utilised a classification method called decision tree to find an appropriate group for users. See Figure 3-1.

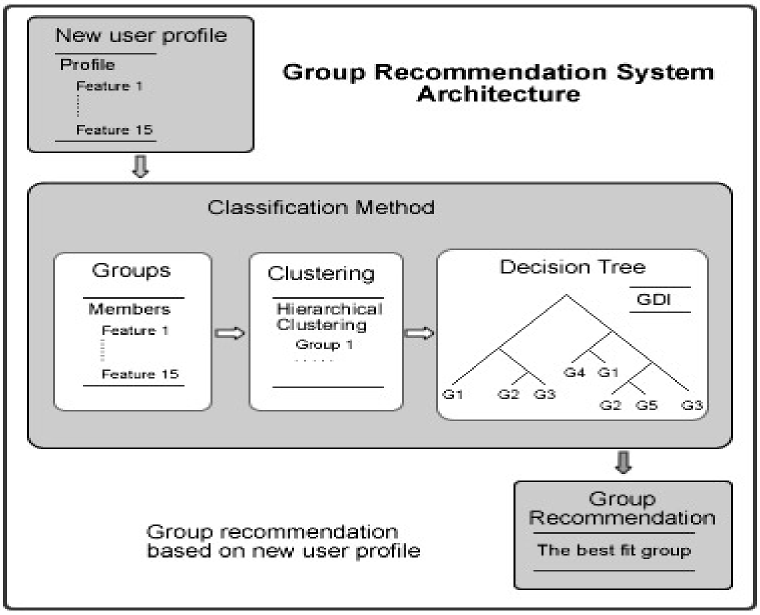

Figure 3-1 Group Recommendation Architecture described in [26]

3.1.3 Trust Propagation: An Approach Based on Epinions

For item-based recommender systems like Amazon, suggestions from 'unreliable' users cannot be filtered. In order to improve the reliability of suggestions, a trust propagation method, which is based on Epinions[13] collects user weight through a trust network [27].

Epinions.com is a consumer opinion website. It helps consumers to choose an appropriate product by comparing ratings, reviews and other information provided by other consumers. In the system, users can issue trust statements for other people. Figure 3-2 shows a small trust network. For example, user A issues a statement values 0.4 for B and 0.7 for C respectively. According to B, C is considered more credible. As a result, suggestions from C are more credible than that from B. Then, B and C also issue statements for D, which create a bridge between A and D. Through the trust chain, the scale of trust people is expanded and trustworthiness can be predicted for unknown users like D.

[13] www.epinions.com

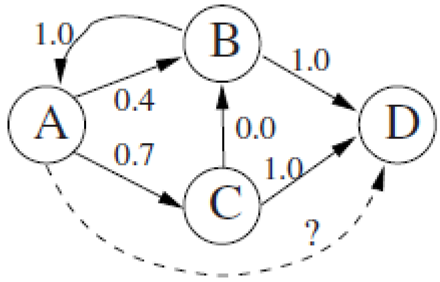

Figure 3-2 Trust Network Presents in [27]

Suggestions from users with trustworthiness are regarded as more credible and valuable for users. The computed trust metrics together with item rating metrics provides a more comprehensive rating for items, which improves the usefulness of suggestions.

3.1.4 User Community & Object-based Relation: An Approach Based on Flickr

Instead of using rated relations to provide recommendations, an approach based on Flickr[14] utilises real user community together with an object-based relation [28] to construct a particular recommendation system with multiple relation layers.

Flickr is a photo sharing system, in which users upload, share and comment photos. Users have their own contracts which are people they are familiar to. Also, they can join communities with people that have the same interests. When different people share or comment the same photo, a relation is also established between those people and also the photo author. They are considered to have the same tastes. Upon those activities, the proposed approach extracts relations between user contacts, user created tags for photos, user communities, favourites and opinions.

In the approach, both conscious (communities and groups) and unconscious (people bridged by photos) relations are utilised to provide recommendations which is shown to be satisfied by users.

[14] www.flickr.com

3.1.5 Analysis

In the presented recommender systems, social relations play an important role. Amazon gets the similar products by collecting items that people are tend to buy together. Facebook itself is both a social network and a giant recommender system. The trust propagation approach utilises a trust net to calculate user trustworthiness level. The value of trustworthiness level is used to compute the final rating value of an item. The approach based on Flickr calculates multiple relations between people, including both direct (group, contacts) and indirect (relations generated by rating and comments) relations. The effectiveness of recommendations is enhanced by the social relations. There are some limitations emerge in those systems:

- There is a cold start (new user) problem in the systems utilises Amazon similar recommendation methods [29].
- Filtering methods are designed to suit to a particular system and recommendations are provided only for that system.
- High computational cost in the Flickr based approach and the Epionions based approach.

3.2 Mapping Between Facebook Open Graph and Schema.org

This section introduces in detail the process of adding Facebook open graph data to schema.org websites, which aims to improve the user experience when users are browsing. It starts from presenting Facebook data extraction, and then, the way that the extracted data can be transformed into the semantic web data format named RDF. Following on from this, the author illustrates some methods for adding Facebook data to web sites that are structured using schema.org vocabularies. Finally, the author will discuss the difficulties of this work.

3.2.1 Facebook Open Graph Information Extraction

The huge volume of social network data is constructed by information about individuals and their relationships. This forms a big information network. As the saying goes, “Any two people in the world are in a maximum of six degrees of separation”. It seems that obtaining the information of the whole world is not too hard. You just need to start from one person, and then walk through the social network graph.

However, for obtaining personal information, we need to consider the privacy issue. Some people may not allow you to access their information, and when you were rejected, your searching would stop there. This is the way the social network works. As a result, if you want to collect data of individuals legally, you will need users' authorisation. To obtain permission from Facebook users, we need to build an application on Facebook[15]. When people join the application, they can choose which permission they want to give to the application. There are many kinds of permissions, such as email permission, without which your application could not get the users' email address, and extended permission, which enables your application to send comments to the user's posts.

After getting the users' authorisation, we can now collect useful data through Facebook graph API. Upon the graph API, Facebook provides the Facebook Query Language (FQL), and some SDK such as JavaScript SDK, PHP SDK and IOS SDK. With the help of FQL, we can query data using a SQL style through HTTP. The following code shows the query command format. `me()` returns the current user's ID, which is a unique ID in Facebook. The `access_token` is required in every query which includes session and permission information. In order to get the object definition, `metadata=1` needs to be included in the URL.

```
GET/fql?q=SELECT+uid2+FROM+friend+WHERE+uid1=me()&access_token
=...
```

The returned data is structured in JSON format, and the code below gives an example of the result for querying a user. The data describes a user called Jane, female, and her Facebook unique ID is `"100003473318947"`. `"connections"` gives links that are references of other information, e.g., the link of `"likes"` refers to the items that Jane likes. `"fields"` gives the definitions of each field of the data structure.

```
{
  "id": "100003473318947",
```

15 https://developers.facebook.com/apps

```
      "type": "user",
      "name": "Jane",
      "gender": "female",
      "metadata": {
        "connections": {
          "likes":
"https://graph.facebook.com/100003473318947/likes",
          ...},
        "fields": [{
            "name": "id",
            "description": "The user's Facebook ID. No
`access_token`
            required. `string`."
          },{
            "name": "name",
          ...
```


3.2.2 Transformation of Extracted Information

The data with JSON format could not be used in the semantic web directly. As the author mentioned before, it has to be transformed to RDF format. Several approaches are used to deal with the transformation, e.g., a semi-automatic approach [30] uses a model-driven engineering technique to transform data to ECORE style. Weaver et al. [16] gave a method for converting data to Turtle RDF. Rowe et al. [31] designed a tool named Facebook FOAF Generator to do the transition from JSON to RDF format which expresses data in FOAF ontology.

The semi-automatic approach divides its process into four steps, data extraction, schema extraction, merging and clearance and refactoring. In the schema extraction stage, they provided detailed methods for extracting types, properties and links. In the merging and clearance stage, they remove the duplicated types and properties. The strong point of this method is that they analysed the strategy for deriving object types and dealing with specific types like enumeration. For instance, they introduce the ways to derive the name of different kinds of object types such as that of a property and a nested type. The problem of the semi-automatic approach is that when they export the ECORE data to XML schema format, they did not mention how to deal with the URI. Unfortunately, without defining appropriate URI, the resources could not be published on the web.

Weaver and Tarjan's method considers dealing with URIs. They gave five URI patterns: an instance with a primitive identifier, an instance with a composite identifier, a type, a type specific property and a generic property for a key. Those methods are quite helpful. A limitation of their method is a lack of dealing with types, which is very challenging because as the type ambiguity of JSON data, there are many scenarios to deal with. Meanwhile, the internal limitations of Turtle such as presentation of language property and prioritization of nodes should be considered. This is because the Turtle cannot present the language property and prioritize the nodes.

The Facebook FOAF Generator has proved to be highly accurate [31], but it only deals with very limited information such as user name, email, websites, interests, location and friends. In addition to this, some core issues like URI and types should be considered because they will influence the accuracy of the RDF.

3.2.3 Adding Open Graph Data to Schema.org Websites

After converting the open graph data from JSON format to RDF format, a mapping from the RDF and schema.org vocabularies could be generated. Once the connection is established, the related open graph data could be added to schema.org websites when users are browsing.

For the data presentation, the author of thesis [32] gives a way to add external information to web pages. He generates a Firefox side bar and displays that information in it. The data is shown in normal text format.

3.2.4 Analysis

Researchers have proposed at least three approaches on Facebook data extraction and transformation. For data extraction, besides the specifications that Facebook has already provided, some methods are analysed. For instance, how to use multiple requests for getting further information (e.g., first get a user's information including his friends list, then use the user id of a friend to get detailed information) are analysed. For data transformation, some challenging problems are solved, e.g., deriving correct type in various circumstances and creating appropriate URI for objects, predicates and properties.

However, there are still some problems to be solved. The author lists them as follows:

- Ambiguous identity of users: Rowe, M. [33] gives a method for identifying the same person in different domains by analysing his relationships in social network. But it is still based on guessing, and the absence of those relationships in any domain or two people with the similar information in the same social circle will cause difficulty for this method.

- Mapping between Facebook open graph and Schema.org vocabularies: As far as the author known, there is no previous work based on this scenario. The lack of references will cause many difficulties.
- Version incompatibility: Information in Facebook keeps changing, how to guarantee the generated event publishes up to date resources.

3.3 Data Mapping Systems

This section presents an overview of data mapping systems and evaluation methods that are commonly used for these systems.

3.3.1 Systems

Data mapping is a labour intensive task because of its inherent complexity and the huge volume of web resources required to be mapped. In a traditional data mapping environment, professionals use a tool to assist with the mapping, and normally, they are required to have specialised domain knowledge. A series of mapping tools have been designed to help people to do the work such as COMA++, PROMT and COGZ, which are among the most famous and popular ones [32].

In recent years, researchers have put more effort into reducing the cost of mapping. The complexity of the mapping interface is demonstrated to affect the efficiency and accuracy of the work. Conroy [32] suggests a tag-based mapping tool using a novel way to use plain text for explaining the relationship between source and target ontologies. It is evaluated to be of high efficiency and accuracy.

Meanwhile, some researchers also focus on the automatic ontology mapping, which runs the mapping process robotically. For example, the Alignment API [11] uses two ontologies as input, and outputs the align results such as the similarity confidence level of the two ontologies and detailed aligned information about class and properties etc.

3.3.2 Evaluation Methods for Mapping Systems

In the aforementioned approaches, the authors mainly use lab experiments to do evaluation. The major process is described as follows: The first step is to carry out a brief investigation on backgrounds of participants. Normally, interviews or questionnaires are used. Then, participants are divided into several groups according to their backgrounds. This is followed by a guide for using the system and an introduction about the experiments. Furthermore, they start completing designed tasks, and long period tasks may go along with intermediate interviews and surveys. Finally, after finishing the task, they collect feedbacks through questionnaires and interviews.

The targets of evaluations to these approaches are different from each other. Some methods are focused on efficiency, effectiveness, user satisfaction, acceptable, convenience and simplicity [34], and others on precision and recall of the mapping results [35]. For the usability feedbacks collection and measurement, a method called System Usability Scale (SUS) is widely used[16]. It employs a questionnaire with 10 items, and each item has 5 response options. The SUS score will be interpreted by converting it to a percentile rank through normalization. For the precision and recall, different experiments could define different measured content. Noy et al. defined precision as the percentage of suggestions users preferred to follow and recall as the percentage of operations suggested by the system performed by the user.

In this dissertation, the mapping task is used to add Facebook information to schema.org websites. The precision is defined as the proportion of the content retrieved from social network that are relevant to the website, out of all the retrieved documents. The recall is defined as the proportion of the documents from social network retrieved that are relevant to the website, out of all the relevant documents available.

3.3.3 Analysis

The existing mapping tools can be divided into two categories: automatic tool and manual tool. The former completes the mapping process robotically, and the later does the mapping with the help of humans. When there is a need of

[16] http://www.measuringusability.com/sus.php

finishing a big volume of mapping tasks, the manual tool requires more labour input.

3.4 Interfaces of Personalised Systems

This section presents an overview of the personalized systems and evaluation methods for them.

3.4.1 Systems

A personalised system adjusts itself on the basis of the users' preferences and the circumstances. It usually needs to take time to study the goals and preferences of the user through their activities. User model refers to the model that is constituted with user personal data about their knowledge, skills, preferences and other information. The user model is established and improved over time. The user model helps the system to be more and more close to user's requirements. The user interface for the system is important. It presents the way for the system to interact with its users and shows the effects the system influenced by the interaction. It helps the system 'to "say the 'right' thing at the 'right' time in the 'right' way"'[17]. McNee et al. [36] divide personalised user interfaces into three categories: system-controlled interfaces, user-controlled interfaces and mixed-initiative interfaces.

System-controlled Interfaces

A system-controlled interface for an adaptive system is pre-designed. Users could not change it directly or could only alter it following the system's suggestions. When users are using the system, it studies users' preferences, and creates each user's personal model.

The system will adjust itself according to its defined algorithms and the user model. For users, it looks like each user has a system that differs from the others.

The system-controlled interfaces require less effort from the users. The adaptation is highly automatic. The interaction with users required to create user model is predefined and well designed. It is indicated that adaptive system with this kind of user interface could create better user model.

[17] http://en.wikipedia.org/wiki/User_modeling

The limitations of the system-controlled interface are as follows：Firstly, because all the adaptation derives from the system's estimate, the accuracy is low [19]. Secondly, the users cannot sense the adaptation process, or sense only a little involvement. As a result, users usually cannot understand the way and the purposes of the adaptation of the system. Thirdly, sometimes the system makes a wrong decision which is far away from the user's expectation. Users may not be satisfied with the system. Finally, the lack of user satisfaction usually causes the lack of longevity of the system [19]. For

User-controlled Interfaces

The adaptive system with a user-controlled interface gives users more opportunities to express their preferences. For individuals, the system's adaptation has higher accuracy. As a result, it could obtain more satisfaction from the user [36].

As a trade-off, users need to do more work to create their user models. A "Lazy user" may not input sufficient configuration information to meet the system's needs. Moreover, because of the large volume of configuration work, the system seems to be unbearable for new users, which is called "New User Problem" [36].

Mixed-initiative Interfaces

As the name suggests, a mixed-initiative interface combines the system-controlled interface and the user-controlled interface together. Researchers are in favour of comparing those three types of interfaces in the same system supporting a mixed-initiative interface. A system-controlled interface or a user-controlled one is achieved by shielding the other part. A representative sample of mixed initiative systems are discussed next.

MICA

The Mixed-Initiative Customization Assistance (MICA) system is designed for customization of Microsoft Word. See Figure 3-3 from [37]. It allows users to configure their own menus, and provides a way for the exchange between the user-controlled menu (left) and the system controlled menu (right).

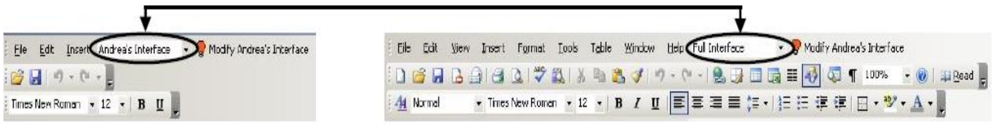

Figure 3-3 MICA Menus – left: user controlled menu, right: system controlled menu

The configuration area is divided into two parts (left). See Figure 3-4 from [37]. The top area presents the user controlled panel, and a "More" button linking to explanation information page (right). The bottom area shows the system controlled panel. Users could click "Accept All" to accept all the system's recommendations.

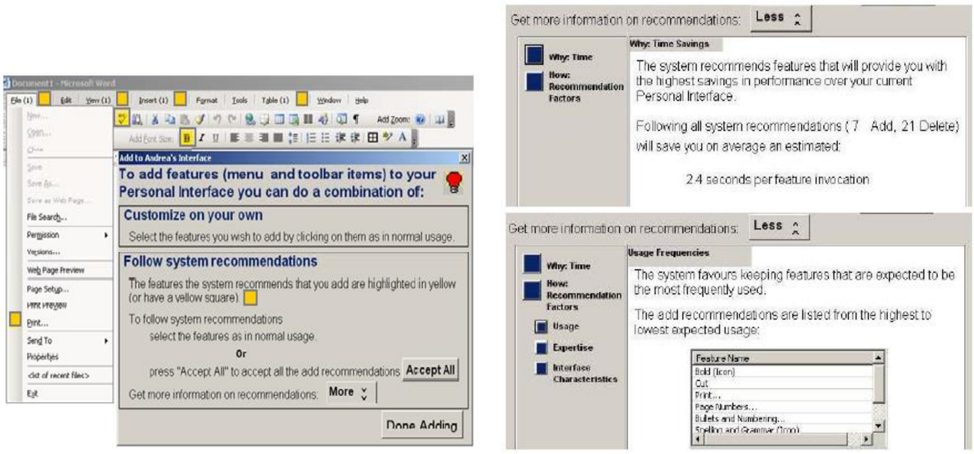

Figure 3-4 MICA Recommendation (left) and explanation page (right)

The system highlights the suggested changes which is easy for users to know the system's decision. Meanwhile, it gives users its recommendation reasons, so that users could know what the adaptation is for.

User controlled panel need to do further click, which is not convenient. Furthermore, it is evaluated that it is hard to find recommendations from the menu list, especially when there are a number of recommendations. In addition to this, the volume of text presented in the panels is big, which requires a high time consumption when reading.

MovieLens

The MovieLens [18] system was created in 1997. It is an online movie recommender system. It uses a mixed-initiative user interface for users to tell their attitudes about movies to the system [36], and its interface is quite simple and clear. It gives useful descriptions, which tells users why to do this work and how to do it. Meanwhile, the system controlled panel provides some hints for users, which let users know what the user controlled panel does mean.

In the author's opinion, although the system tried to explain the reason for doing the task to users, the explanation is almost meaningless. Users still have no idea about how the system will change according the results of the task.

LookOut

The LookOut project focuses on scheduling and meeting management. It is designed for Microsoft Outlook. When a new message arrives, it notifies the user and attempts to insist users to update their schedules [38].

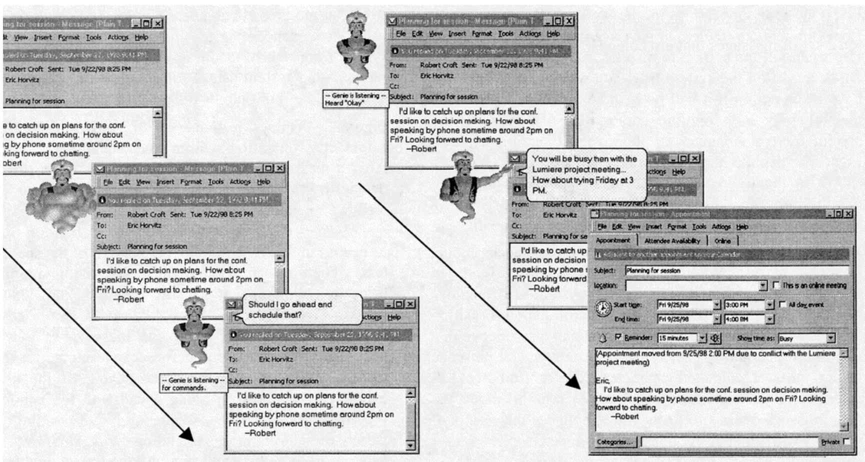

Figure 3-5 LookOut User Interface

The user interface includes two panels: notice panel and configuration panel. See Figure 3-5 from [38]. The notice panel is a dialog with a cartoon Aladdin near it. When receives a new message, the system first analyse the message and then decides if it is worthwhile to interfere the user to pop up the notice panel. The dialog first presents requests like asking for the permit of the task to go

[18] http://movielens.umn.edu

forward. After receiving the confirmation, the system creates a new appointment and opens a new window presents its suggestion.

The cartoon Aladdin is a very interesting design, creates more fun for the user. In addition to this, the pop up dialog gives explanation of the system's decision. Furthermore, the system provides an algorism to guess the user's willing, and then determine if it is time to infer the user or not.

One of the limitations is also the cartoon. In the author's opinion, not every people do like cartoon. Under some serious circumstances like the user is doing an important presentation, it is not suit to pop up a cartoon. Another limitation is that when the dialog is pop up, it requires the user's action, e.g., close it or give a feedback, which interrupts the user in further.

3.4.2 Evaluation Methods for Adaptive Systems

At the earlier stage of evaluation methods for adaptive systems, evaluation approaches tends to compare a system with and without adaptive features. There are some potential difficulties such as selection of non-adaptive controls and equilibrium points. At the later stage, evaluation approaches mainly focus on the adaptive activities and effects of the system itself. E.g., formative and summative approaches, and the latter is widely used. Layered evaluation is one of formative evaluation methods which attempt to solve the problems that claimed before [39].

A widely used method named layered evaluation uses a framework with five layers which unifies previous approaches. For each layer, there are some particular goals and targeted methods. With those goals and methods, the development process could goes forward along with evaluation process.

There are some issues that should be paid attention to, e.g., some systems are not suited for this five-layer-model. Some layers have to be combined together and the associated methods need to be changed accordingly. Following on from this, this model does not cover the meta-adaptivity evaluation. Here adaptivity refers to the term ability of a system to adapt itself to changed environment[19].

[19] http://en.wikipedia.org/wiki/Adaptability

3.4.3 Analysis

Many good features are achieved for an adaptive system [32] [38]:

- Showcase of the reason for the task: Give a motivation for users to do the task.
- Task need to be understandable: This will influence on the accuracy of it.
- Timing of tasks to occur: The system has to determine when to show the task to reduce interruption.
- Number of tasks to present: If the volume of the task if too big, users may not do it.
- Frequency of tasks: To find an appropriate time interval for the task.
- Meaningful selective answers: Give the user some hints to do the task.

Another feature an adaptive system should have but have not been achieved is that the system should have the ability to know the correctness of its decision. Because of the complexity of the algorisms for determine adaption, sometimes the system may make a wrong resolution. In order to confirm the achievements, the system could interact with the user to collect feedbacks. At the same time, the user can also update their model that has been created.

3.5 Overall Analysis

In this section, the author analyses the state of the art related to this research, including the recommender systems that based on social information, the extraction and transformation of Facebook information, the mapping between transformed RDF and schema.org vocabulary, and the display of Facebook information on schema.org websites.

- Social information supported recommender systems: Three recommendation approaches utilise social relations to enhance the effectiveness of suggestions: the item-to-item approach for Amazon, a trust propagation approach based on Epinions.com and an approach utilises multiple relation layer based on Flickr.com.
- Facebook information extraction and transformation: There are a semi-automatic approaches to transform Facebook data to ECORE style or

Turtle RDF, there is also an approach to do the transition from JSON to RDF format.

- Mapping tools: the author presents two kinds of mapping tools: Firstly, COMA++, COGZ and the Tag based mapping tool are manual tools, for which the help of humans is required in the mapping process. Secondly, the Alignment API supports automatic mapping which needs less labour input.
- Displaying Facebook information on schema.org websites: One practice of Tag based mapping tool presents a way to add external information on relevant websites. It designed a Firefox sidebar, and display related information in it. Users can open and close the sidebar at their wishes.
- Evaluation: A series of methods focus on the effectiveness and user satisfaction, such as the precision and recall for effectiveness and the System Usability Scale for user satisfaction.

From the state of the art of recommender systems, we can see the social data has become a promising source to enhance the recommendation effect. The rationale behind this is convincing: A user's subjective opinion on different items could indicate some deep relations between them, which may be hard to find out through objective comparison between the explicit features of the items. However, simply utilizing all the social data without differentiating users, as what is done by Amazon, is still problematic: There might be many users who have very different taste than the current user, or just have malicious purpose. The opinions of these users will pollute the recommendation to the current user. A basic idea to solve this is to rank or cluster the users first, and then different users have different weights on their opinions. The problem for these approaches is that the computational cost is very high. The reason for the current recommender systems using computed relations but not purely existing social circle is that data related to the social circle is limited. However, for the mainstreams social network platforms, most of the users have already created a very sophisticated social circle and share a huge volume of information from many websites into the service platform, such as Facebook. With this data, a recommender system can benefit from these circles that are good references to clustering users who not only have similar taste but are also trustworthy to the current user. So the remaining problem is how to utilize the existing social circle in Facebook to support the recommendation in a system outside Facebook.

In order to design an approach to meet those requirements, the author compares the existing approaches. See Table 3-1. The term “Social Relation” means using social relation information to enhance the relevancy of the recommendations. “Multiple Websites” means recommendations are generated from multiple websites. “Web Content” means suggestions are relevant to the user browsing web content. “Outside Social” means suggestions are provided outside social networks. “Simplicity” means if the user interface is simple and clear, and the operations required to use the system are simple. “User Interruption” means to what extent the system interrupts users.

Other terms in the table need explanation: “Epinions”: the Epinions based approach; “Flickr”: the Flickr.com based approach; “Aims”: the research aims of the dissertation; “Y”: the system has the feature; “N”: the system does not has the feature; “--”: no reference; “Permanent” means recommendations are shown in a webpage permanently. Users do not have the option to not show them. “Optional” means users have the option to choose to show the recommendations or not.

Table 3-1 Table of Comparisons

	Amazon	Facebook	Epinions	Flickr	Aims
Social Relation	N	N	Y	Y	Y
Multiple Websites	N	Y	Y	N	Y
Web Content	Y	N	N	N	Y
Outside Social	N	N	N	N	Y
Simplicity	Y	Y	Y	Y	Y
User Interruption	Permanent	Permanent	--	--	Optional

In the state of the art, there are approaches providing practicable technologies for Facebook information extraction, transformation and display. However, there are still many difficult issues remaining. From the very start of the process when users open a schema.org annotated website, the problem is how to parse the webpage and let the server know the content information that the user is currently browsing. After collecting this data, the server has to match Facebook graph data to it and find relevant items. This relies on the mapping between Open Graph Protocol and schema.org. Although there are already some existing mapping tools, making a choice for an appropriate one is still an issue. When passed to the instance mapping, difficulty emerges from getting graph information from linked websites. Sub types defined by different websites make the process even harder. Finally, designing an appropriate content presentation style which is easy for users to understand and to master is also a challenge.

Chapter 4. Design

This chapter presents detailed information of the approach to using social data to enrich web browsing. Following an overview of the approach, the method for social data extraction and enrichment is presented. After this, a two phase mapping method for matching social data and the web content is introduced.

4.1 Approach Overview

In the approach, social data is used to provide recommendations, and data mapping is used to filter the social data to get items that are relevant to the web content a user is browsing.

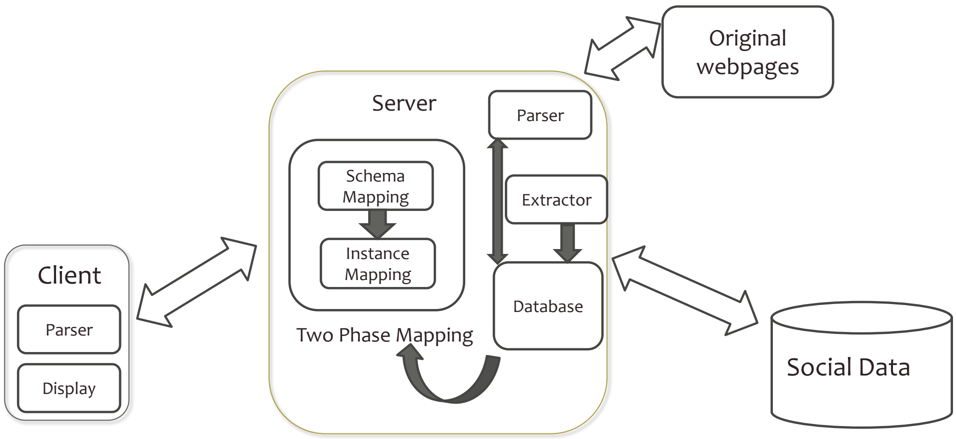

Figure 4-1 System Structure

Figure 4-1 shows the system structure of the approach. The whole system is divided into three parts: part one contains the source data for web enrichment including social data and the original webpages that are shared into social network services by users. Social network platforms maintain the user personal information and sophisticate relations. Meanwhile, users share data from external websites, which contains rich information. Both the relations and original web content are useful for users. Part two is a server, which is in charge of extracting social data and original web content, saving extracted information and filtering this information for users. Part three is a client for displaying the suggestions to users.

The social data is needed to be kept up to date. But in order to simplify the whole process, an assumption is made that all the data is collected one-time. The process proceeds in five steps:

- **Step One:** A component called Extractor (see Figure 4-1) queries the social data from social network services and saves those data in the local database. Gaining permission from users and the social network service is required.
- **Step Two:** A component called Parser extracts metadata from original webpages following the links extracted from social data in step one, and then saves this data into the local database. A difficulty emerges in this part that some websites have self-defined schema subtypes and these subtype metadata cannot be extracted and used by common methods. The strengthen model is designed to tackle the problem which is described in next section.
- **Step Three:** When a user is browsing, the client side component calls the server to ask for relevant information. Metadata in the currently browsing webpage is also extracted and sent to the server.
- **Step Four:** This is the core part of the approach called *two phase mapping*. It contains two components. One is schema mapping, which establishes the relations between two different schema sets annotating the user browsing web content and the social data respectively. The other one is instance mapping, which utilises the results from schema mapping to compute the similarity between the web content and the items saved in local database. Then it selects more relevant ones and sends them to the client.
- **Step Five:** The client receives results from the server and displays the results for users.

The following sections presents the solutions to the two major problems in the above steps, i.e., the strengthen model in the data preparation stage in step two and the two phase mapping method for matching social data to user browsing web content in step four.

4.2 Strengthen Model for Data Enrichment

Social data extracted from social network is mainly in JSON format, and has no schema structural information. To get the schema information, the system follows the links in social data and collects this information by parsing original webpages. However, some of the webpages are structured by self-defined schema subtypes, and the data about the subtypes cannot be parsed through common parsing method for a schema set. In order to use this rich data, a component is required, which is capable to deal with multiple websites.

In this dissertation, the component is called *strengthen model*. It works in two steps: firstly, all the websites that require a particular process for metadata extraction have to register to the system. The registration is carried out by saving a site name and function name pair into the system. Secondly, a function with the registered name is implemented for the particular page parsing. When the Parser is called, it firstly enumerates all the registered websites to check if the website that the current link belonged to were registered. If so, the registered function will be called, otherwise, the common used parser will be called.

The strengthen model has the capability to deal with multiple websites. When a new special site is required to serve, the only thing to do is to register the website and band a function. So the model requires minimal labour input to be capable for a new site.

4.3 Two Phase Mapping

The two phase mapping method acts as a core in the approach. In the schema mapping phase, similarity between classes and properties is calculated using the Alignment API. These results will be used in the instance mapping, which computes the instance similarity and select the items that have higher similarity with the user browsing content.

4.3.1 Phase One: Schema Mapping

Ontology mapping is one of the five main APIs of the Alignment API, which is a research project of Exmo group[20]. It utilises two ontology files as input, and

[20] http://exmo.inrialpes.fr/

provides the alignment results for class names, properties, etc. A series of string distance methods are used in the alignment process.

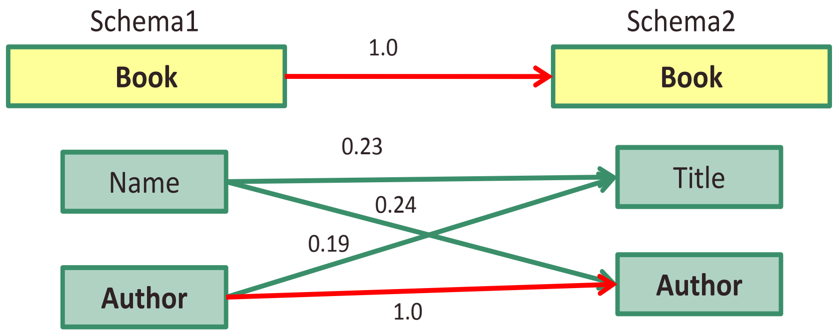

Figure 4-2 Alignment Sample Using String Distance (similarity)

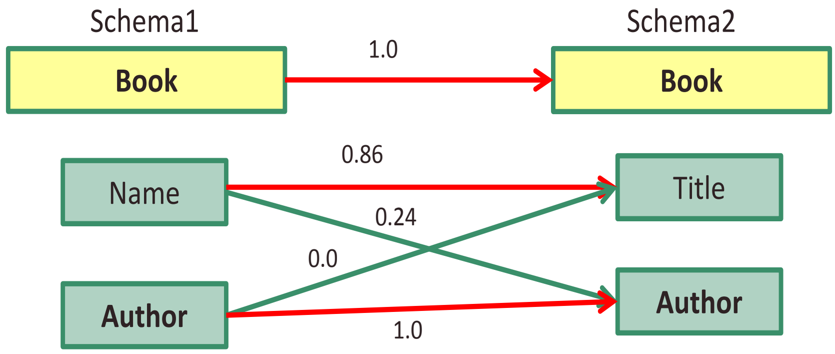

Figure 4-3 Alignment Sample Using Wordnet (similarity)

Figure 4-2 shows a simplified example of the alignment process. The left class Book and the right class Book are defined under different schema sets respectively. A string distance method is used to calculate the similarity between class names and property names. To calculate the similarity between properties, it firstly gets the Cartesian product on all the properties in the two classes. Then, use the string distance method to calculate similarity between each pair. Finally, select the one with the highest similarity as the matched one. In the figure, for Name of the left Book, the similarity of Author of the right Book is higher than that of Title. But the similarity between Author and Author is higher than that between Name and Author. So, only Author and Author are matched. Assume the weight of class name is 0.4, and property is 0.6, then the similarity between the two classes is calculated as: *class-name weight * class-name similarity + property weight * Avg (property similarity) = 0.7*.

Following this way, the result is not ideal. The reason of the low accuracy is that string distance methods cannot get the semantic relations between words, but since the two schemas are defined by different organizations, it is very likely that they use different words with similar meanings to name the same attribute. For example, in the above case, Name and Title are used for the actually same attribute, but because of their different shapes, they are not considered as relevant.

In order to solve this problem, the meanings of attribute names are need to be compared, rather than only string distances. Wordnet is integrated into the mapping process to achieve this.

Wordnet is a lexical database, which maintains the semantic relations between words. Figure 4-3 shows the alignment results using Wordnet. Name and Title have a high similarity value at 0.86. Two matched pairs are found out and the similarity of the two classes is 0.958, which is much accurate than using the string distance method.

However, Wordnet brings high computational cost and this influences the performance of the system. The fact is that in the same schema set, people used to use same words to express similar opinions. For example, in Open Graph Protocol, title means book name in the Book class and it also means film name in the Movie class. Since the same word can appear several times in different class definition in a schema set, caching is used to improve the performance. The resulting similarity of each pair of strings calculated by Wordnet is remembered and then each time before a new pair is fed to Wordnet, the program checks if there is a cached result for this pair, and if so, the cached result is directly returned.

The results returned in phase one comprise two parts of information: the similarity between classes and the similarity between properties. This information is used for computing instance similarity in phase two.

4.3.2 Phase Two: Instance Mapping

Instance mapping calculates the similarity between the instance extracted from user currently browsing web content (source) and the instances transformed from social data and original webpages (targets). This aims to find items that are more relevant to the web content.

The process is as follows. Firstly, the mapping tool selects all the instances related to the current user’s friends in the local database. Secondly, it iterates on the selected instances to calculate the similarity between source and targets by calculating the similarity for all the matched properties. The matching information is obtained from phase one. The property-value-similarity is calculated by a string distance method for the matched property values of the source and the target. Thirdly, it summates the similarity of all the properties and plus the class similarity to get the instance similarity. Finally, it ranks all the instances. The formulas are shown as follows:

Property Similarity: property-name-similarity * property-value-similarity

Instance Similarity: $\sum_{i=1}^{n} property\ similarity_i +\ class\ similarity,$ where n is the number of matching properties.

The following pseudo-code shows the instance mapping process. It iterates on all the items shared by the user’s friends, and computes the instance similarity between these items and the `source` data (metadata the user is currently browsing). `item.type` means the metadata is collected by the Open Graph parser. `item.subtype` means the metadata is collected by the strengthen model. `item.vproperty` means the value of this property for this item.

```
List = []
For item in allFriendShares:
   If mapps.get(item.type)!= None:
      allMatchedProperties = mapps.get(item.type)[0]
      classSimilarity = mapps.get(item.type)[1]
   If mapps.get(item.subtype)!= None:
      allMatchedProperties = mapps.get(item.subtype)[0]
      classSimilarity = mapps.get(item.subtype)[1]
   Sum = classSimilarity
   For property in allMatchedProperties:
      Sum+=property.similarity*similarity(item.vproperty,
      source.vproperty)
   List.append((sum, item))
Sort(list,sum, reverse)
```

4.4 Summary

The approach utilises data mapping and social data to provide recommendations for users. In summary, the whole process can be divided into two stages: data preparation and filtering. In the data preparation stage, social data and metadata from original web content is collected. A strengthen model is used to collect metadata of subtypes for special websites. This information enriches the schema data of items and makes them be possible for calculation in the filtering stage. In the filtering stage, the two phase mapping method is used to filter the information that is relevant to the web content a user is browsing. The schema mapping phase calculates the similarity between classes, including similarity between class names and properties names. The instance mapping phase matches the source metadata and the targets by calculating the similarity between them. In order to improve the accuracy of the schema mapping, Wordnet is used to replace string distances methods. Caching is utilised to solve the subsequent low performance problem brought by the Wordnet.

Chapter 5. Implementation

This chapter presents the implementation of the approach presented in the last section, as a prototype called the Suggestion Tool. The environment information and the system activities about the system are shown in this chapter. In addition, difficulties haven met in the implementation is presented. Finally, preparations for the experiment are shown.

5.1 An Overview of the Suggestion Tool

The dissertation utilises Facebook data for web enrichment of schema.org annotated websites. A Facebook application is required for data collection. At the client side, a Firefox plugin is implemented for displaying the suggestions.

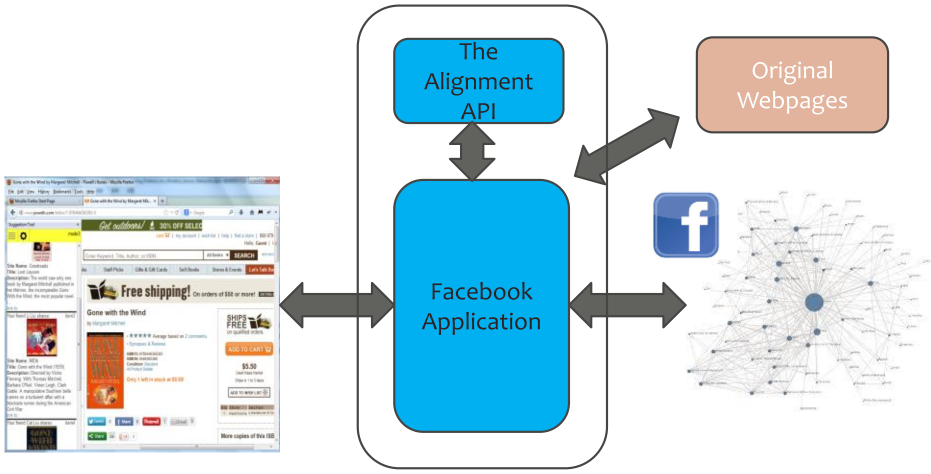

Figure 5-1 Structure of the Suggestion Tool

Figure 5-1 shows the system structure of the Suggestion Tool. The Facebook application and the Alignment API act as the server in the design. It contains the components for Facebook data extraction, original webpage parsing, two phase mapping and the strengthen model. At the client side, the Firefox sidebar is easily to be opened and closed. When a user is browsing schema.org annotated websites, the sidebar will parse the webpage and send a request to the Facebook application to get information that are relevant to the content he/she is browsing. Recommendations provided by the Facebook application are displayed in the sidebar.

5.2 Environment

This section presents the system environment information for the Suggestion Tool. The following table illustrates the detailed information for three core parts of the system.

Name	Runtime environment	Languages
Facebook Application	Google Engine	Python
The Alignment API	Google Engine	Java, OWL
Firefox Plugin	Firefox browser	JavaScript, HTML, XML

Table 5-1 System Environment

The choice of using Google Engine[21] is made for many reasons: Firstly, as a cloud computing platform, it has a high availability at 99.95% and high security design. Secondly, it can adjust itself automatically for the increasing number of requests of an application. Thirdly, it has relatively completed runtime environment, and saving the time spent on setup and configuration. Finally, the data store has encapsulated methods for database operations, such as insert and select.

The reason for using a browser sidebar as the client end is that it is independent from the webpage a user is browsing. Users can open or close the sidebar when they require, which minimises the interruption for users.

5.3 System Activities

Figure 5-2 illustrates the major activities of the system.

[21] https://developers.google.com/appengine/features/

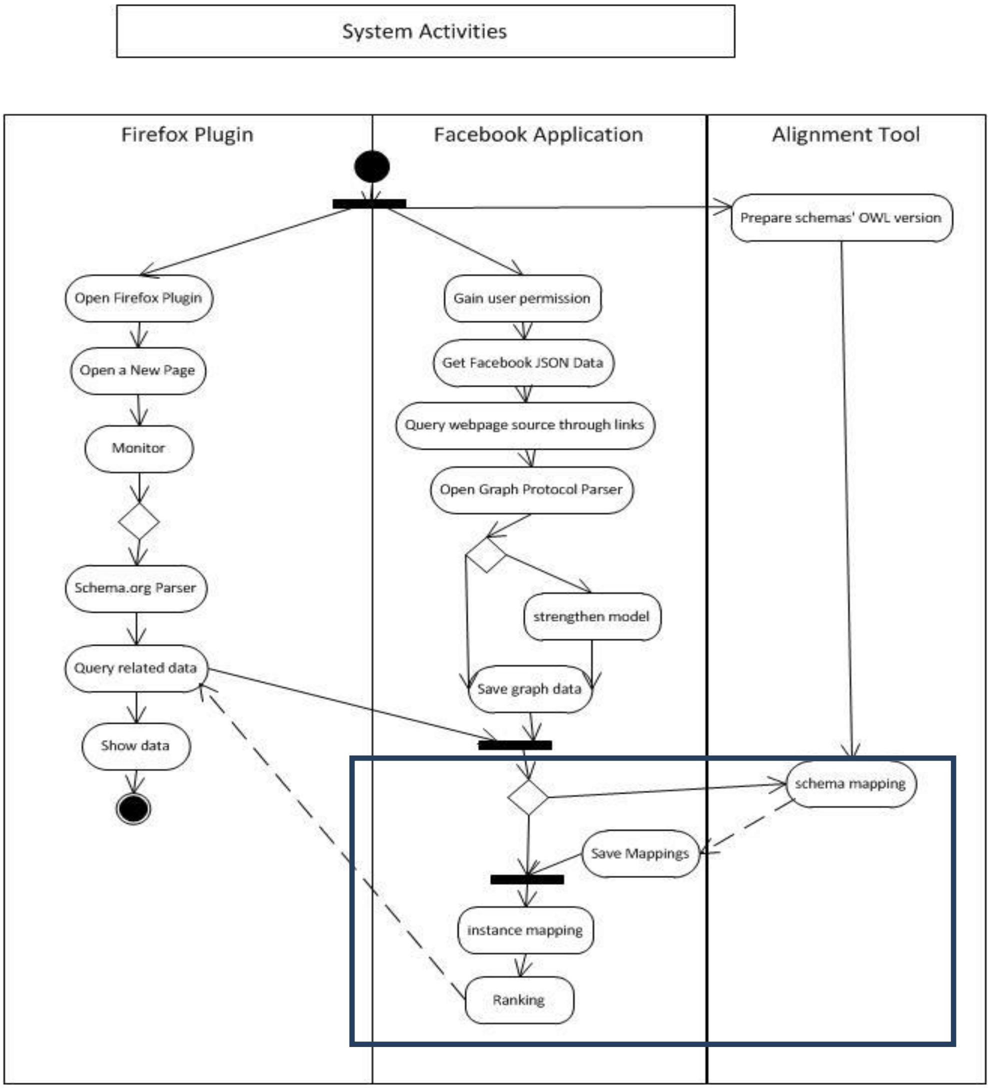

Figure 5-2 System Activities

The swim line in the middle shows the activities of the Facebook application. Target users are required to have a Facebook account. When a user logins the Facebook application at the first time, the application asks for permissions from the user, such as email, read_friendlists and friends_activities. With those permissions, the application can query Facebook information about the user and his/her friends through Facebook graph API. Results are transferred via a HTTP request in JSON format. Activity information contained in JSON data has a little schema information. In order to get more information about shared data, the application follows the links to the original websites, and an Open Graph parser is used to collect Open Graph Protocol metadata. Then it checks if the website has registered to the strengthen model. If so, accordingly registered function is called to further collect schema subtype information. Finally, collected metadata is saved into the data store. When the client calls the application, it filters data in the data store. Activities in the blue rectangle shows the filtering process which is the two phase mapping described in section 4.3.

Figure 5-3 User Interface

The swim line in the left shows activities in the client. When a user opens a new page, a monitor will check if the webpage is annotated by schema.org. Then a schema.org parser collects the schema information and sends it to the Facebook application. After receiving the results, the sidebar displays them as recommended items. See “Suggestion Tool” sidebar on left hand side of Figure 5-3.

5.4 Implementation for Two Phase Mapping

Phase one schema mapping is executed by the Alignment API. The Facebook application sends a HTTP request to the API, with three parameters: source schema (the schema used by the website a user is browsing. In the dissertation, it means schema.org), target schema (Open Graph Protocol) and source type (class name of the source). After calculating, results are sent back via HTTP request in JSON format, including similarity between source class and target class and similarity between matched properties.

NameAndPropertyAlignment class of the Alignment API is used for the Alignment. In the *align* method, string distance algorithm is replaced by a Wordnet based algorithm. *Ws4j.jar* from [12] is a Wordnet version that being runnable on Google App Engine. The following code shows the invoking for WS4J. *st1* and *st2* are strings for comparing. *V* means the similarity results of *st1* and *st2*.

```
v = WS4J.runLIN(st1, st2)
```

Schema mapping results from phase one is saved in local database and the same mapping does not need to do it again. After instance mapping, results are transformed to JSON format and sent back to the client.

5.5 Difficulties Involved

During the implementation, many difficulties emerged and had to be overcome, such as the following:

- The Alignment API is complicated and a lack of completed source code even makes it more difficult to study.
- Standard Wordnet cannot work in Google Engine, and finding an appropriate version requires much effort.
- Finding appropriate parsers for schema.org [22] and Open Graph Protocol[23].
- The lack of tutorial for implementing a Firefox plugin for Firefox new versions (not lower than 22.0)

5.6 Updates for Experiment

For evaluating the effectiveness of the Suggestion Tool in a controlled experiment, the tool had to be updated with three modes, which are required to be switchable to each other automatically. Mode1 utilises a random algorithm, which can be easily implemented as a separate path. The difference between mode2 and mode3 is whether to use the strengthen model or not. But the effect of the strengthen model performs in the data preparation stage, and all three modes

[22] http://getschema.org/
[23] https://github.com/erikriver/opengraph

proceed on the same dataset. In order to solve the problem, the author adds a subtype attribute to the item structure, which saves the schema type collected by the strengthen model. When query the mapping results, mode2 uses the information saved in attribute *type* as a key, and mode3 can use both *type* and *subtype* information. For example, goodreads defines a subtype called *good_reads:book* (no mapping results), which saves as the value of *type*. Strengthen model analyses it as *book* (standard Open Graph Protocol type, and has mapping results) and saves as the value of *subtype*. Without mapping results from schema mapping, the instance mapping cannot work. So, the difference of mode2 and mode3 can be presented.

There are also some design aspects related to content presentation. In order to help participants to grade items, the author also added a sign for each item (e.g., item1, item2) shown at the top right corner. Similarly, mode name is shown on the menu bar and different colours applied to the menu bar for different modes. For reducing the grading time, the number of displayed items is set at six (see experiment section for explanation).

Chapter 6. Evaluation

This chapter presents an evaluation according to research objective: ***RO5:*** *Evaluate effectiveness and user satisfaction of the tool.* To achieve this objective, an experiment was carried out using the prototype called the Suggestion Tool. To allow for comparison, the Suggestion Tool is designed to have three modes. From mode1 to mode2, the functions of the two phase mapping approach are added incrementally. Mode1 only has basic functions and utilises a random mechanism to filter social data. Mode2 utilises the two phase mapping approach described in Section 4.3. Compared to mode2, mode3 worked with the strengthened model (described in section 4.2), which fully implements the proposed approach.

In this section, detailed information about the experimental method, results and result analysis are provided.

6.1 Hypotheses

This experiment has the following hypotheses:

H1: The sidebar shows suggestions that are relevant to the content a user is browsing. The most relevant suggestions will be made by mode 3 (with full support of the approach) which performs better than mode 2 (with only mapping support), and both perform better than mode 1 (random suggestions). This hypothesis addresses the effectiveness of the sidebar tool.

H2: Users are satisfied with the Suggestion Tool. This hypothesis addresses the user satisfaction with the sidebar tool.

6.1.1 Metrics

Appropriate experiment data is selected for measuring the different hypotheses:

H1: Grading scores provided by participants. For each suggestion shown in the sidebar, participants give a score to show how strong they think the suggestion is relevant to the webpage the participant is browsing.

H2: Scores of the user survey and the SUS (System Usability Scale) survey is used to measure satisfaction.

6.1.2 Analysis Method

The analysis methods for each hypothesis are:

H1: The methods for **H1** are as follows:

A: Calculate and compare the average item score of each mode. Summate the scores for each item, and compare them for the three modes.

B: For checking if the order of the mode operation influences the results, the author uses T-test to analysis the results of different participant groups (One group firstly sets the tool to mode1, then mode2 and finally mode3. A second group firstly sets the tool to mode2, then mode3 and finally mode1).

C: For checking if the effectiveness for suggestions linked from different websites is not significantly different, T-test is also used to analyse the scores for suggestions from different websites.

H2: The methods for **H2** are as follows:

A: Calculate the average response time from system logs for each mode and also consider the user feelings for the delay from question 3 in the user survey to see if the delay for adding suggestions to the sidebar is acceptable to the user.

B: Compare the scores of question 5 in the user survey to see if users like the way the sidebar presents content.

C: Compare the scores of question 1 and 6 in the user survey to see if users think the sidebar is useful when they are browsing.

D: Compare the overall SUS scores.

6.2 Experimental Setup

An experiment was setup to validate the hypotheses outlined above. In the experiment, participants install a Firefox sidebar, which is the client side of the

Suggestion Tool, and browse a website that sells books. Then, the sidebar will show recommendations that are considered to be relevant to the book the user is browsing. Afterwards participants give their judgments for the tool.

In this section, information about the experimental environment, scenario, materials, procedure and data collection are presented.

6.2.1 Environment

The server for the Suggestion Tool is built upon Google Engine, and the client end is a Firefox sidebar. For preparing the experiment data, real Facebook accounts are used to share web content from six different websites to Facebook platform. Then, the Suggestion Tool queries the data and follows its link to collect more schema information form original webpages. A function called strengthen model enriches the information for special websites with self-defined schema subtypes. When a user is browsing the book selling website (*www.powells.com*), the server responses differently for three modes: mode1 randomly select some social data and shows them in the Firefox sidebar for users. Mode2 uses the two phase mapping approach to filter the social data (subtype information extracted by the strengthen model is shielded) and get the most relevant ones to the web content the user is browsing. Mode3 also uses the two phase mapping approach to filter the social data and the enriched data by the strengthen model is usable.

For each mode, the tool provides six items, which are shown in two screens. Because participants are required to grade each item according to its relevancy, too many items will cost a lot of time.

The test data is about four different types, and 72 in total. Table 6-1 presents the detailed information. *Number* means the number of items shared from the website into Facebook. *Strengthen Model* means the method for processing data from special website with special schema definition.

Type	Address	Number	Strengthen Model
music	www.vevo.com/	10	No
movie	www.imdb.com	13	No
article	sports.yahoo.com	10	No

book	www.goodreads.com	19	Yes
book	www.ebay.com/chp/books	13	No
book	www.ebooks.com/	7	No

Table 6-1 Experiment Data

In the experiment, participants are required to choose one book to browse. See Table 6-2. *Relevant Items* shows the number of relevant items for each book in all the experimental data according to the dissertation author's design, which will be used as the gold standard during the evaluation.

No	Book Name	Author	Relevant Items
1	A History of Ireland	Edmund Curtis	6
2	Football Genius	Tim Green	4
3	The Great Gatsby	F. Scott Fitzgerald	3
4	Gone with the Wind	Margaret Mitchell	5

Table 6-2 Experiment Target Books

The relevant items are distributed across three websites. For example, there are six items about *A History of Ireland,* and three of them are on *www.goodreads.com,* one of them on *www.imdb.com* and two of them on *www.ebooks.com.* See Table 6-3.

No	Book Name	Total	GoodReads	IMDB	Ebooks
1	A History of Ireland	6	3	1	2
2	Football Genius	4	3	1	0
3	The Great Gatsby	3	1	1	1
4	Gone with the Wind	5	2	1	2

Table 6-3 Target Books Distribution

6.2.2 Experiment Scenario

A participant browses *http://www.powells.com*, which is a book selling website. He chooses one book from Table 6-2, and then searches the book on *powells*. When he opens the page presenting detailed information of the book, the

Suggestion Tool will provide him recommendations that are relevant to the book the participant is viewing. The recommendations are those shared by his/her friends on Facebook (Not real user friends, they are friends of the dedicated test account). The participant looks through the suggestions and see how strongly relevant he/she thinks the suggestion is, and then gives an appropriate score to the suggestion. After grading all the items, he/she switches the sidebar into another mode.

6.2.3 Participants

For the experiment, the total number of participants is 12. All of them are students who study in TCD and they are required to be older than 18. Participants are divided into two groups: group A (from A1 to A6) and group B (from B1 to B6). The orders of mode operation for each different group are not the same. For example, for participant 1, he/she firstly sets the sidebar to mode1, then mode2 and finally mode3. For participant 12, he/she firstly set the sidebar to mode2, then mode3 and set mode1 in the end. For the consideration that users may get some hints when they discover mode2 is better than mode1, a natural guess is that mode3 will be stronger than mode2. This may lead them to give fake scores. Meanwhile, because the number of participants is not big, and the big difference of the three modes is with (mode2, mode3) or without (mode1) the two phase mapping approach, so in the experiment, there are only two orders: mode1->mode2->mode3 and mode2->mode3->mode1.

6.2.4 Materials

All the materials provide for participants are as follows:

- Ethics forms describe required ethics information.
- An introduction sheet presents a briefly overview of the project and the detailed procedure for the experiment. See Appendix A.
- A sample video shows how to install the Suggestion Tool.
- A sample video shows how to use the Suggestion Tool to do the experiment.
- A grading form for users providing scores for items. See Appendix B.
- A user survey for the sidebar tool. See Appendix C.

- A System Usability Scale (SUS) survey for the Suggestion Tool. See Appendix D.

6.2.5 Procedure

In order to answer any queries in a timely manner, participants were accompanied by the author when they were doing the experiment. They followed the procedure described as below:

- Sign the *Participants Consent Form* and read the *Participants Information Sheet*.
- Download videos and the Suggestion Tool from a link.
- Install the Suggestion Tool according to the sample video.
- Have a look at the sample video about the experiment process.
- Choose one of the four books listed in a table below. Then go to http://www.powells.com and search for the book that was chosen. Participants are required to use the same book for different modes, for the sake of fair comparison.
- Open the Suggestion Tool.
- With their participant number, follow the operation sequence listed in a table. For example, Participant 1:

  Sets mode 1, looks at information in sidebar, fills out form

  Sets mode 2, looks at information in sidebar, fills out form

  Sets mode 3, looks at information in sidebar, fills out form
- Fill out Questionnaire form

6.2.6 Data Collection

Experiment data is collected in two ways: one method collects data directly through grading form and questionnaires filled out by participants (subjective user opinions). Another one collects response time automatically by recording system logs (objective data).

6.3 Results

This section presents all the experimental results, including grading results for suggestions of mode1, mode2 and mode3, response time and questionnaires.

The sidebar tool provides six items. Each participant uses the same book for mode1, mode2 and mode3. The higher the score is, the stronger relevant they think the information in the sidebar is to what is being browsed. Score is divided into five levels, and uniformly varies from 0 to 1. The five levels are 0, 0.25, 0.5, 0.75 and 1 respectively. See Table 6-4, Table 6-5, and Table 6-6.

No	Book Name	item1	item2	item3	item4	item5	item6	SUM
A1	A History of Ireland	0.25	0.25	0.25	0.25	0.25	0.25	**1.5**
A2	Football Genius	0.75	0.25	0.25	0.25	0.75	0.5	**2.75**
A3	The Great Gatsby	0	0.5	0	0	1	0	**1.5**
A4	A History of Ireland	0.75	0	0	0	0	0	**0.75**
A5	A History of Ireland	0.25	0	0.25	0	1	1	**2.5**
A6	A History of Ireland	0	0	0	1	1	0.75	**2.75**
B1	The Great Gatsby	1	0	0	0	0	0	**1**
B2	Football Genius	0.25	0.25	0	0.5	0.25	0.5	**1.75**
B3	A History of Ireland	0	0	1	0	0	1	**2**
B4	Gone with the Wind	0	0	0	0.75	0	0	**0.75**
B5	A History of Ireland	0.5	1	0.5	0.5	0.25	0.5	**3.25**
B6	Football Genius	0	0	0.25	0.25	0.75	0	**1.25**
SUM		**3.75**	**2.25**	**2.5**	**3.5**	**5.25**	**4.5**	**21.75**

Table 6-4 User Relevancy Scores of Items for Mode1

Table 6-4 shows that for mode1 (the random suggestion mode), most of the items are considered not relevant to the book the user is browsing. A small percentage of items are recorded as bigger than 0.5 and a few of them are scored 1. Because the size of candidate items is not extremely big, and therefore even randomly selection may still result in relevant data. The biggest number is 3.25 for the book *A History of Ireland*. The smallest number is 0.75 for **A4** and **B4**, and both have only one book that the user agree to be relevant.

Table 6-5 shows the grading scores of items for mode2 (the mapping approach mode).

No	item1	item2	item3	item4	item5	item6	SUM
A1	1	1	1	0.25	0.25	0.25	**3.75**
A2	0.75	0.25	0.5	0.25	0	0	**1.75**
A3	1	1	0.25	0	0.25	0.75	**3.25**
A4	1	1	0.75	0	0	0	**2.75**
A5	1	1	1	0	0.25	0	**3.25**

A6	1	1	0.75	0	0	0	**2.75**
B1	1	1	0.25	0.25	0.5	0.5	**3.5**
B2	0.75	0.25	0.25	0.25	0.25	0.25	**2**
B3	1	1	0.75	0.25	0	0	**3**
B4	0.75	1	1	0	0	0	**2.75**
B5	1	1	1	0.5	0.25	0.25	**4**
B6	0.75	0.25	0.25	0.25	0.25	0.25	**2**
SUM	**11**	**9.75**	**7.75**	**2**	**2**	**2.25**	**34.75**

Table 6-5 User Relevancy Scores of Items for Mode2

It can be easily seen that the results of mode2 is better than that of mode1. 75% of the participants gave three items a good score as 0.75 or 1.

Table 6-6 shows the grading scores of items for mode2 (the full approach mode).

No	item1	item2	item3	item4	item5	item6	SUM
A1	1	1	1	1	1	1	**6**
A2	1	1	1	1	0.25	0	**4.25**
A3	1	1	1	0	0.5	0.5	**4**
A4	1	1	0.75	1	1	0.75	**5.5**
A5	1	1	1	1	1	1	**6**
A6	1	1	1	1	1	0.75	**5.75**
B1	1	1	1	0.25	0.5	0.25	**4**
B2	1	0	0	0.75	0.25	0.25	**2.25**
B3	1	1	0.75	1	1	0.75	**5.5**
B4	0.25	0.25	0.75	1	1	0	**3.25**
B5	1	1	1	0.75	0.25	0.25	**4.25**
B6	1	1	1	0.75	0.25	0.25	**4.25**
SUM	**11.25**	**10.25**	**10.25**	**9.5**	**8**	**5.75**	**55**

Table 6-6 User Relevancy Scores of Items for Mode3

For mode3, most of the items achieved a score not less than 0.5, which is a pleasing result. The items with a background colour are considered outliers according to the gold standard. For the item2 and item3 of **B2**, the participant gave a comment that there is no need to suggest if a book were the same as the one he is browsing.

The table below shows the status of the system response time. It is a time interval from parsing the web page the user is browsing to showing the suggestions in the sidebar. The values of in the total columns show the response time which is a sum of parsing time and filtering time. *Parsing* means the time used for parsing the web page and getting the book information the user is browsing. *Filtering* means the time used for getting suggestions from server to the client end, including filtering time and transformation time. See Table 6-7.

No	**mode1**	**mode2**			**mode3**		
	total	**total**	**parsing**	**filtering**	**total**	**parsing**	**filtering**
A1	4	6907	6896	11	3094	3082	12
A2	6	6411	6394	17	9397	9379	18
A3	2	5983	5968	15	7275	7264	11
A4	6	5186	5174	12	3468	3453	15
A5	4	3578	3559	19	3486	3471	15
A6	14	3132	3112	20	2979	2962	17
B1	5	4496	4486	10	4777	4759	18
B2	10	4746	4728	18	5969	5953	16
B3	5	6687	6676	11	2944	2926	18
B4	21	4315	4299	16	4700	4690	10
B5	16	4088	4070	18	5540	5529	11
B6	17	4629	4611	18	4690	4673	17
AVG	9	5013	4998	15	4860	4845	15

Table 6-7 Response Time (milliseconds)

The score for questions in the user survey is also in five levels: 0, 0.25, 0.5, 0.75 and 1. The bigger the number is, the better the result is.

No	**1**	**2**	**3**	**4**	**5**	**6**	**SUM**
A1	0.75	1	0.75	0.75	1	1	**5.25**
A2	0.75	0.25	0.25	0.75	1	1	**4**
A3	0.75	0.75	0.75	0.5	1	1	**4.75**
A4	0.75	0.75	0.5	0.5	0.5	0.75	**3.75**
A5	0.75	0.75	0.25	0.75	0.75	0.75	**4**
A6	0.75	1	1	0.75	0.5	0.5	**4.5**
B1	0.75	0.75	0.25	0.5	0.75	1	**4**
B2	0.5	0.5	0.75	0.75	0.25	0.75	**3.5**
B3	0.75	1	0.75	0.75	1	0.75	**5**
B4	0.75	0.75	0.75	1	0.75	1	**5**
B5	0.75	0.5	0.75	1	0.5	0.75	**4.25**
B6	0.75	1	0.75	0.75	0.5	0.75	**4.5**
SUM	**8.75**	**9**	**7.5**	**8.75**	**8.5**	**10**	**52.5**
AVG	**0.73**	**0.75**	**0.63**	**0.73**	**0.71**	**0.83**	**4.38**

Table 6-8 User Survey

Table 6-8 shows the results of the following features of the Suggestion Tool: 1: usefulness; 2: understandability; 3: response time; 4: content presentation; 5: number of displayed items; 6: opinion about recommendations. Appendix B shows the detail survey information.

The following table shows the scores for SUS. See Table 6-9.

No	1	2	3	4	5	6	7	8	9	10	SUM
A1	7.5	7.5	7.5	7.5	7.5	10	10	10	7.5	10	**85**
A2	2.5	7.5	7.5	10	7.5	10	10	10	7.5	5	**77.5**
A3	5	10	10	10	7.5	10	10	10	10	10	**92.5**
A4	5	2.5	5	2.5	7.5	5	7.5	5	7.5	5	**52.5**
A5	7.5	10	7.5	10	7.5	10	10	7.5	7.5	10	**87.5**
A6	7.5	10	10	10	7.5	7.5	10	10	7.5	10	**90**
B1	2.5	10	10	7.5	7.5	7.5	10	5	10	7.5	**77.5**
B2	5	7.5	7.5	7.5	5	7.5	7.5	7.5	7.5	7.5	**70**
B3	7.5	5	7.5	7.5	7.5	5	7.5	7.5	7.5	7.5	**70**
B4	5	7.5	5	10	7.5	7.5	10	10	7.5	10	**80**
B5	7.5	5	10	10	10	10	10	10	7.5	10	**90**
B6	7.5	10	10	10	7.5	10	7.5	10	10	10	**92.5**
AVG	**5.83**	**7.71**	**8.13**	**8.54**	**7.50**	**8.33**	**9.17**	**8.54**	**8.13**	**8.54**	**80.42**

Table 6-9 System Usability Scale

The questions of the SUS are simplified as: 1: like the system or not; 2: complexity; 3: easy to use; 4: handleability; 5: function integration; 6: consistency; 7: learnability; 8: be cumbersome or not; 9: confidentiality; 10: easy to use and learnability. Appendix D shows the detailed information about the SUS questions.

6.4 Analysis

6.4.1 H1: Effectiveness Analysis

Hypothesis H1: *The sidebar shows suggestions that are relevant to the content a user is browsing. The most relevant suggestions will be made by mode 3 (with full support of our approach) which performs better than mode 2 (with only mapping support), and both perform better than mode 1 (random). The hypothesis addresses the effectiveness.*

Firstly, in order to get an overview of the status of suggestions for the three modes, the comparison of average scores is presented. In addition, the contrast between the sums of scores of each item for three modes gives more detailed information. Secondly, as supplementary description, a T-test is used to measure if there is a significant difference between results of two participant groups. The same measure is applied to check if suggestions from different websites affect the effectiveness of the tool differently.

Figure 6-1 presents the comparison results of the average scores of all participants and all suggestions for the three modes. It can be easily seen that mode3 performs better than mode2 which is then followed by mode1. Mode3 nearly triples the figure of mode1.

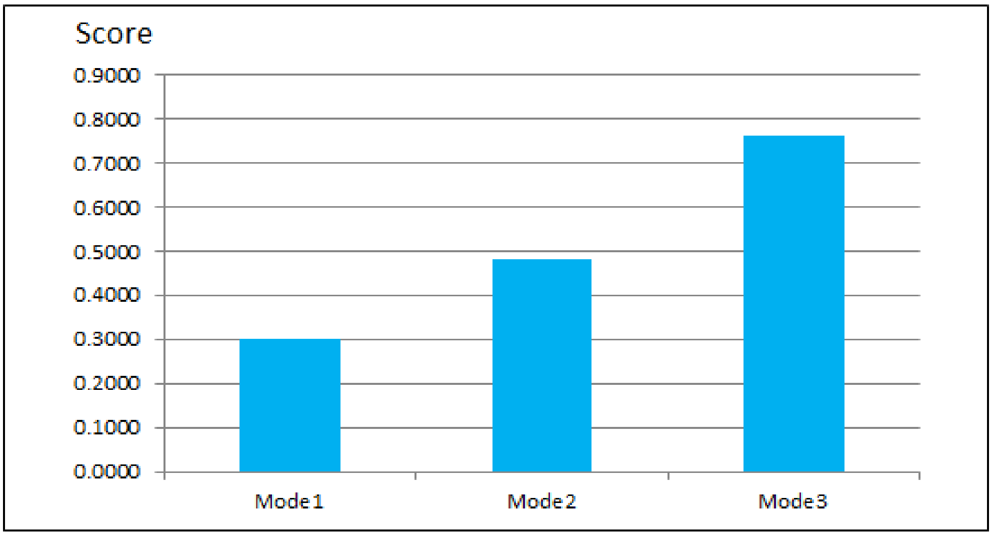

Figure 6-1 Average User Relevancy Scores for Three Modes

The following figure presents the comparison results of the sum of scores of all participants.

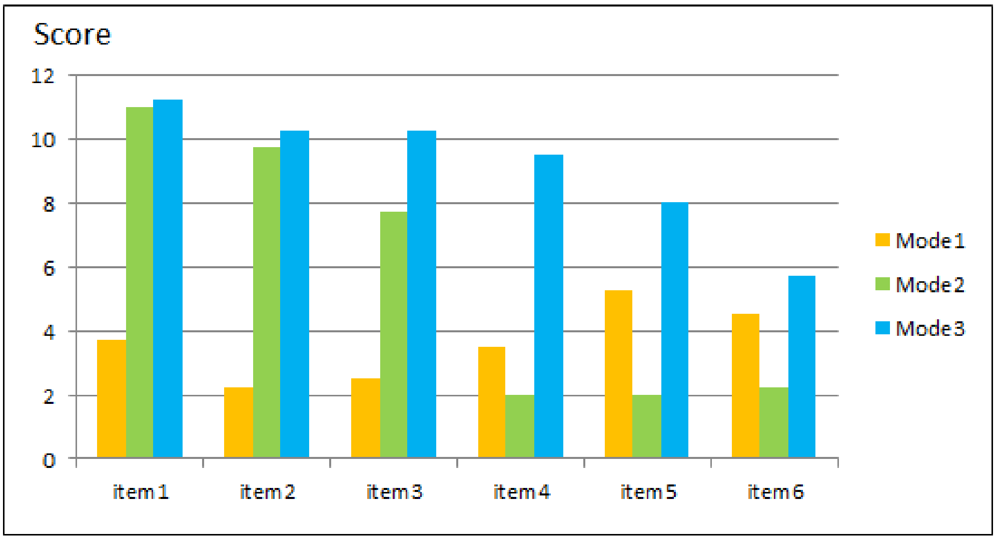

Figure 6-2 Sum of Item Scores for Three Modes

It can be seen that mode3 performs better than mode1 and mode2 for every item. For the first three items, mode2 have a considerable good result, but mode1 presents better results in the last three items. This is because mode2 only can get part of the relevant items, and mode1 is in a random state, which all the suggestions have a possibility to be relevant. Furthermore, there is a ranking for all the items shown in the sidebar for mode2 and mode3, and the most relevant item displays on top (The order for showing items is from item1 to item6). It can be seen that the items ranked higher, their score is higher, which also indicates that users agree with the item ranking. See Figure 6-2.

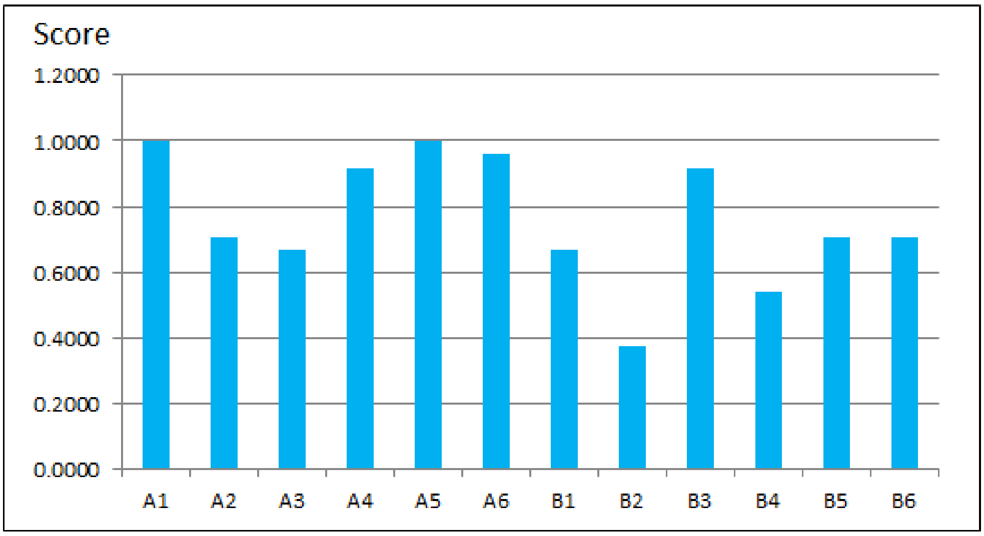

Figure 6-3 Average Item Score for Participants for Mode3

Figure 6-3 shows the average item score for all the participants. B2 and B4 gave a lower score. One of them thought there is no point to suggest a same book as the one he is browsing. Another thought books written by the same author is not important to him.

In the experiment, participants are divided into two groups, and different group operates in different order (mode1->mode2->mode3 or mode2->mode3->mode1). If the results from two groups are not significantly different, then we are more confident that all the grading scores are credible.

Unpaired two tailed T-test [40] is used to judge the trueness of the hypothesis. It computes the summated results in Figure 6-1. Because participants chose different books in different group, and the scores for various books are not the same. So, the author selects results of the people in both groups that chose the same book for calculation. *A History of Ireland* is a book that half the participant chose, so the results of this book are used. In Table 6-10, α=0.05 means the threshold for statistical significance. If P-value is smaller than α, the two group results are considered significantly different.

Source Data	P-value (α=0.05)	Significant Difference
Sum of scores	$t=0.254$, $p= 0.261 > 0.05$	None

Table 6-10 T-test for Two Participants Group

The T-test result shows that the grading results are not influenced by the operation order.

The data source of suggestions that a user’s friends shared from different websites are not significantly different to the sidebar.

For checking the significant difference between suggestions shared from different websites, user results are compared to the gold standard in Table 6-11. The data in columns titled *Goodreads, Ebooks and IMDB* means the number of relevant items user agreed (score >= 0.75) are linked from these websites. GS means that number in the gold standard. The cells with a background colour highlight the different results between user scores and the gold standard.

No	Book Name	Goodreads	GS	Ebooks	GS	IMDB	GS
A1	A History of Ireland	3	3	2	2	1	1
A2	Football Genius	3	3	0	0	1	1
A3	The Great Gatsby	1	1	1	1	1	1
A4	A History of Ireland	3	3	2	2	1	1
A5	A History of Ireland	3	3	2	2	1	1
A6	A History of Ireland	3	3	2	2	1	1
B1	The Great Gatsby	1	1	1	1	1	1
B2	Football Genius	1	3	0	0	1	1
B3	A History of Ireland	3	3	2	2	1	1
B4	Gone with the Wind	0	2	2	2	1	1
B5	A History of Ireland	3	3	1	2	0	1
B6	Football Genius	3	3	0	0	1	1

Table 6-11 Items from Different Websites

In order to find the significant difference level of the results of different websites, unpaired two tailed T-test is used. Data (quotient of experiment results and the gold standard) of three different websites are calculated as pairs: Goodreads / GS VS Ebooks / GS (Group1), Goodreads/ GS VS IMDB / GS (Group2) and Ebooks / GS VS IMDB / GS (Group3). The table below shows the T-test results (Equal variances not assumed).

Source Data	P-value (α=0.05)	Significant Difference
Group1	t=-0.93, p= 0.367 > 0.05	None
Group2	t=-0.437, p= 0.666 > 0.05	None
Group3	t=0.447, p= 0.661 > 0.05	None

Table 6-12 T-test for Data from Different Websites

In the experiment, participants also discovered some relevant items that are not in the gold standard. For example, the film Forrest Gump is considered relevant to the book Football Genius because the participant thinks Gump is a good football player. Another item is considered to be relevant because they are both hot-blooded stories. This indicates there is certainly room for improvement of the system effectiveness.

In conclusion, from the analysis results, the indications are that most users got better suggestions when using mode3. With the support of the two stage mapping process and the strength model, the effectiveness of the tool is significantly improved. Similarly, from the overall view, mode2 performs better than mode1. Meanwhile, indications are that the operation order did not influence the judgement of users, and scores provided by users are credible. Furthermore, the difference of source website of suggestions does not cause significant difference for the results. This would indicate that effectiveness of the tool will not be affected by the changing of the source data (the changing should vary among the tool's capable websites). All the evidence indicates that **H1** is true.

6.4.2 H2: User Satisfaction Analysis

Hypothesis H2: *Users are satisfied with the Suggestion Tool.*

The analysis aims to answer three questions: Q1: Do users agree that the delay for presenting suggestions is acceptable? Q2: Do Users like the way the sidebar presents content? Q3: Do users like using the tool when they are browsing?

Q1 will be evaluated according to both response time information and the user satisfaction for the delay. From Table 6-7, the average response time is extracted. For mode1, the response time is very low, because suggestions are provided randomly and there is no need to parse the website. For mode2 and mode2 the average filtering time is the same, which means the strength model did not cause extra delay. The parsing time varies depending on the response time of the schema parsing server.

No	mode1	mode2			mode3		
	Total	total	parsing	filtering	total	parsing	filtering
AVG (ms)	9	5013	4998	15	4860	4845	15

Figure 6-4 shows the user feeling about the delay for displaying suggestions. One them is not satisfied with the response time because a lack of running status of the tool.

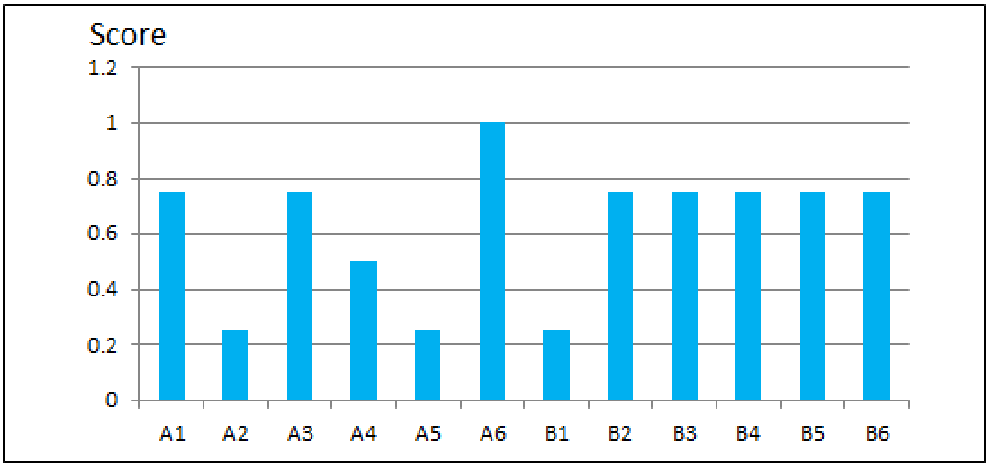

Figure 6-4 User Satisfaction for the Response Time

The parsing time is a bottle neck for the Suggestion Tool. Currently, the sidebar tool calls a parsing server which is an external implemented application. If the function can be integrated into the server of the tool, the response time can be significantly improved. Even though, near 75% of participants think the delay is acceptable and 67% of participants tend to do not mind the delay strongly. So the answer for Q1 is *yes*.

In the user survey, question 4 is: *I like the way the sidebar presents content.* For question 4, one participant comments that the tool has perfect content presentation, and another participant likes the tool because it is implemented as a Firefox sidebar which is easy to open and to close.

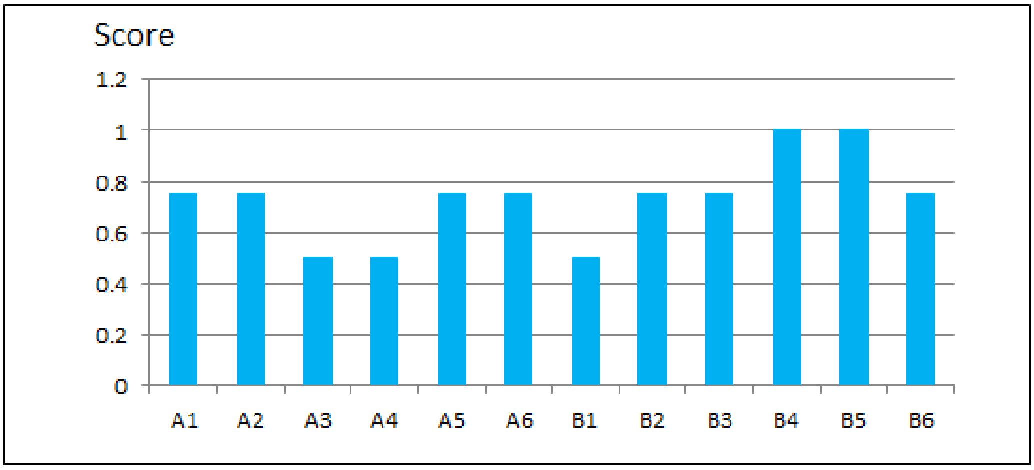

Figure 6-5 Content Presentation

Figure 6-5 shows most of participants like the content presentation of the tool, and only three participants keep neutral. The answer for Q2 is *yes*.

Question 1 and question 6 in the user survey are designed to evaluate this hypothesis. Question 1 is: *I think the sidebar can provide useful content when I'm browsing*. Question 6 is: *In general I like products with recommendations support*. Figure 6-6 shows the average score for the two questions.

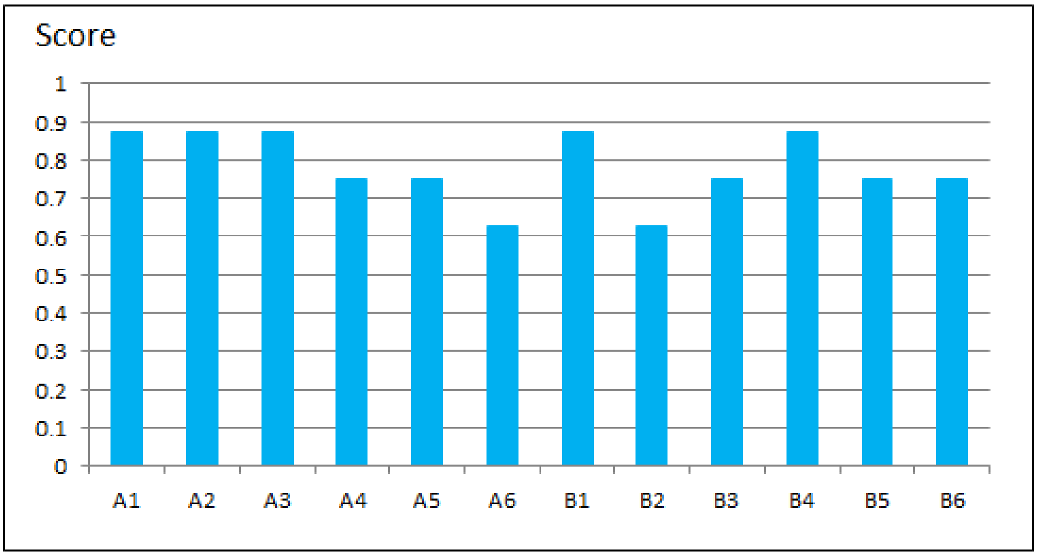

Figure 6-6 Usefulness of the Tool

All the users agree that the tool is useful when they are browsing. The answer for Q3 is *yes*.

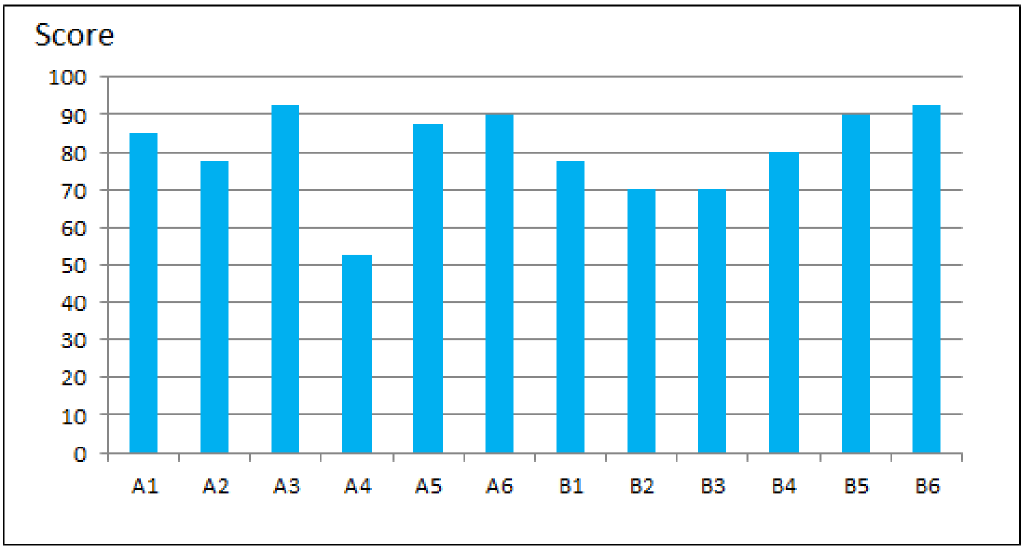

Figure 6-7 System Usability Scale

In addition to this, the Standard System Usability Scale survey is used to evaluate the overall usability of the system. Figure 6-7 shows the experiment results. More than half of the participants give a score above 80, and only one people the score is below 70. The participant thinks the system is complex and needs a technical people’s support to use it.

Comments Analysis

Table 6-13 highlights some user comments (simplified), which are divided into two categories: about usability and about effectiveness.

Usability	**Effectiveness**
1. Easy to understand and to use, perfect content presentation	5. For a book, a movie is not as relevant as a book
2. Optional sidebar: avoided some of the shortcomings of common recommendation tools	6. Do not understand the irrelevant items
3. Hopes the number of display items is configurable	7. No need to recommend same books
4. Hopes to show the running status of the browser	

Table 6-13 Comments

Comment 1 and comment 2 in Table 6-13 shows they think the sidebar tool is easy to master and the presentation style is good. In the future research, the author will keep these features of the sidebar. For comment 3, currently, the number is six, but this number is designed to suit evaluation purposes. For the fact is that too much items will cost a very long time and makes people tired, six would be an appropriate test number. But for a real product, making the number of items displayed configurable is a good suggestion. Comment 6 and comment 7 shows that sometimes ‘too relevant’ and ‘too not relevant’ will make users confused. So, configuration for the range of relevant level will improve the user satisfaction of the sidebar tool. The table below presents the extracted requirements for future research. See Table 6-14.

Keep	**Create**
Operation process	Configuration for display number of suggestions
Content presentation style	Configuration for the range of relevant level
	Display running status of the browser

Table 6-14 Future Requirements

6.5 Evaluation Summary

This chapter presents an overview of the evaluation process and the key points arising from the analysis.

The experiment is designed to analyse the hypotheses extracted from the research question: *To what extent is it possible to efficiently and effectively leverage social network data to provide browsing recommendations based on webpage meta-data that users are satisfied with?* It is conducted on a prototype called Suggestion Tool, which is in the form of a Firefox plugin. The sidebar tool has three modes: mode1 provides suggestions randomly, mode2 utilize the two-stage mapping process and mode3 adds the strengthen model upon mode2. All the results about three modes are collected during the experiment. 12 volunteers participate in the experiment, and work in two different groups: one sets the sidebar from mode1 to mode2 and then mode3. Another one sets it from mode2 to mode3 and then mode1. Results from two groups are compared and analysed. Questionnaires about the system usability are also used to investigate user feelings about the tool.

The analysis results give an answer to the question: *Does the prototype fully meet the requirements stated in the research question?* The following paragraphs discuss those results.

Effectiveness: The user relevancy scores of suggestions are compared between three modes. The comparison results show that with full functions support, the effectiveness of the sidebar tool is greatly improved. Through user study, the author also has some interesting findings. Some of the relevance between items is dig out by participants according to their very detailed information or some kinds of coincidence, e.g., the relation between football and Gump. This shows an opportunity to improve the effectiveness of suggestions in the future.

For concerning if the operation order for three modes would influence the judgement of participants, a comparison between results of two groups is conducted. The unpaired two tailed T-test results show there is no significant difference between two data sets. So, the scores provided by participants are credible.

The effectiveness of the tool also benefits from its capability of collecting and filtering suggestions shared by Facebook users from multiple websites. The T-test results for user grading scores show that there no significant difference for the tool to process the data from these websites.

User Satisfaction: Factors like efficiency and user satisfaction are evaluated through analysing the system log and the feedback of questionnaires.

- **Efficiency:** 75% of the users agree that the delay for presenting suggestions is acceptable (score >= 0.5). By analysing the system response time, the results show that the webpage parsing is the bottle neck of the sidebar tool. This also creates a requirement for the future research.
- **Content Presentation and Usefulness:** All of the users express they like the content presentation of the sidebar tool and 75% of them give a very high score at >= 0.75. Even better, 100% of participants agree the tool can provide useful suggestions when they are browsing, and 83% of them give a score >= 0.75. Similarly, for the SUS scores, only one result is below 70. The overall results show that the user satisfaction is achieved.

Other Findings: The user comments also provide three good suggestions:

- The display number of suggestions is better to be configurable.

- The range of relevant level (how much relevant) for suggestions is better to be configurable. Because some users do not like suggestions to be ‘too relevant’ or ‘too not relevant’.
- Display running status of the browser. Let the user know the browser is executing the task or not.

In conclusion, the prototype fully meets the requirements stated in the research question, which is an approach to be efficient, effective and with user satisfaction.

Chapter 7. Conclusion and Future Work

Social network services gather a huge volume of data and maintain sophisticated social relations for users. This data contains rich information in many aspects such as books, movies, articles and music, which is useful for enriching the web content that users are browsing with recommendations related to them. Because the social data is based on social relations, the suggestions can be more useful and credible than that from general data sets. To bring useful information out of social network services, the difficulty is how to tackle the problem of matching social data and the browsing web content.

In the state of the art, approaches based on social data taken in recommender systems have been designed to providing recommendations within a particular system. The dissertation utilises a two phase mapping approach to bring the social data outside social network services and let this rich data serve people when they are browsing. This approach works in the context that both the social data and the web content are annotated by structural languages that can be interpreted by machines.

The process can be divided into two stages: data preparation and two phase mapping. In the data preparation stage, social data is extracted from social network services and enriched by parsing the original webpages. The strengthen model is helpful for extracting metadata from websites with self-defined subtypes. Data preparation reduces the number of requests to network services and guarantees the performance of the system. Two phase mapping method is in charge of filtering the social data and get relevant items for users. In the schema mapping phase, the Alignment API is used to analyse ontology files and compute similarity between classes. It establishes the relations of social data and user browsing content at the schema definition level. By default, the API applies a series of string distance algorithms to calculate the similarity. But the accuracy is low because string distance methods are lack of the capability for analysing semantic relations between words. Wordnet based methods are utilized to improve the accuracy, and a caching mechanism is used to reduce the computational cost brought by Wordnet. With the help of Wordnet and caching, the Alignment API is capable to work in a recommender system. In the instance mapping phase, instance similarity is computed based on the property similarity and class similarity calculated in phase one. The one with higher similarity is considered to be more relevant to the web content a user is browsing.

Based on the approach, a prototype called the Suggestion Tool is implemented. It utilises Facebook data to provide recommendations, which are listed in a Firefox sidebar. In the experiment, participants browse a book selling website, and the Suggestion Tool filters the Facebook data shared by the user's friends to get the items that are relevant to the book the user is browsing. The evaluation results show that recommendations provided by the Suggestion Tool are considered to be effective and with user satisfaction. With the help of the two phase mapping approach, social data is brought out of social network platform to serve people when they are browsing.

Following this direction, there is still research that can follow on.

Firstly, at this stage, the ranking of items is according to the content similarity, but sometimes potential relations between items are hard to be dug out. For example, the film *Forrest Gump* is considered to be relevant to the book *Football Genius* because Gump himself is a football genius. Can the system study the choices from users and improve the two phase mapping method?

Secondly, some users do not have big number of friends on a single social network service, and a small data source will significantly influence the recommendation effect. Is it possible to combine data from multiple social network services together or to use user relations in multiple dimensions?

Thirdly, considering the fact that users usually use different browsers, at the client side, the sidebar is better to be suitable to multiple browsers.

Finally, the webpage parsing is the bottleneck of the system performance, because the author outsources the parsing to a third-party web service which is quite slow and very unstable. A parser running on the same server as the filter is required to address this issue.

In summary, the dissertation presents a two phase mapping approach to tackling the problem of matching social data and web content in order to provide users with useful content when they are browsing. The approach utilises ontology mapping to compare data types, and utilises Wordnet library to improve the mapping accuracy. Also a strengthen model is used to enrich the social data linked from multiple websites. The dissertation contributes to the state of the art about recommender systems for bringing social data shared by their friends out of social networks to provide users with suggestions based on the webpage metadata they are currently browsing.

Chapter 8. References

[1] P. Resnick and H. R. Varian, “Recommender systems,” *Communications of the ACM,* vol. 40, no. 3, pp. 56-58, 1997.

[2] X. Zhou, Y. Xu, Y. Li, A. Josang and C. Cox, “The state-of-the-art in personalized recommender systems for social networking,” *Artificial Intelligence Review*, vol. 37, no. 2, pp. 119-132, 2012.

[3] C.-N. Ziegler and G. Lausen, “Analyzing correlation between trust and user similarity in online communities,” in *Trust management*, Springer, 2004, pp. 251-265.

[4] R. R. Sinha and K. Swearingen, “Comparing Recommendations Made by Online Systems and Friends.,” in *DELOS workshop: personalisation and recommender systems in digital libraries*, 2001.

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

[6] K. Swearingen and R. Sinha, “Beyond algorithms: An HCI perspective on recommender systems,” in *ACM SIGIR 2001 Workshop on Recommender Systems*, 2001.

[7] D. Mielach, “Americans Spend 23 Hours Per Week Online, Texting,” 2 7 2013. [Online]. Available: http://www.businessnewsdaily.com/4718-weekly-online-social-media-time.html. [Accessed 5 8 2013].

[8] StatisticBrain, “Social Networking Statistics,” 11 12 2012. [Online]. Available: http://www.statisticbrain.com/social-networking-statistics/. [Accessed 5 8 2013].

[9] P. Mika, “Schema.org Update,” Yahoo, 2012. [Online]. Available: http://www.slideshare.net/AlexShubin1/schemaorg-iswc2012-15283142. [Accessed 5 8 2013].

[10] Y. Kalfoglou and M. Schorlemmer, “Ontology mapping: the state of the art,” *The knowledge engineering review,* vol. 18, no. 1, pp. 1-31, 2003.

[11] EXMO, “Alignment API and Alignment Server,” [Online]. Available: http://alignapi.gforge.inria.fr/. [Accessed 16 7 2013].

[12] H. Shima, “WS4J,” [Online]. Available: https://code.google.com/p/ws4j/. [Accessed 16 7 2013].

[13] C. Bizer, T. Heath and T. Berners-Lee, “Linked data-the story so far,” *International Journal on Semantic Web and Information Systems (IJSWIS),* vol. 5, no. 3, pp. 1-22, 2009.

[14] N. Shadbolt, W. Hall and T. Berners-Lee, “The semantic web revisited,” *Intelligent Systems, IEEE,* vol. 21, no. 3, pp. 96-101, 2006.

[15] P. Brusilovsky and N. Henze, “Open corpus adaptive educational hypermedia,” *The adaptive web,* pp. 671-696, 2007.

[16] J. Weaver and P. Tarjan, “Facebook Linked Data via the Graph API,” *Semantic Web.*

[17] D. Gašević and M. Hatala, “Ontology mappings to improve learning resource search,” *British Journal of Educational Technology,* vol. 37, no. 3, pp. 375-389, 2006.

[18] N. S. J. R. a. J. C. Silva, “E-Business interoperability through ontology semantic mapping,” *Proc. of the Processes and Foundations for Virtual Organizations,* pp. 315-322, 2003.

[19] X. Shen, B. Tan and C. Zhai, “Implicit user modeling for personalized search,” in *Proceedings of the 14th ACM international conference on Information and knowledge management,* 2005.

[20] J. B. Schafer, J. A. Konstan and J. Riedl, “E-commerce recommendation applications,” *Data mining and knowledge discovery,* vol. 5, no. 1, pp. 115-

153, 2001.

[21] J. O'Donovan and B. Smyth, “Trust in recommender systems,” in *Proceedings of the 10th international conference on Intelligent user interfaces*, 2005.

[22] M. R. ,. P. H. Òscar Celma, “Foafing the music: A music recommendation system based on RSS feeds and user preferences,” in *in ISMIR*, 2005.

[23] G. Linden, B. Smith and J. York, “Amazon. com recommendations: Item-to-item collaborative filtering,” *Internet Computing, IEEE,* vol. 7, no. 1, pp. 76-80, 2003.

[24] B. Sarwar, G. Karypis, J. Konstan and J. Riedl, “Item-based collaborative filtering recommendation algorithms,” in *Proceedings of the 10th international conference on World Wide Web*, 2001.

[25] “Facebook Advertising,” [Online]. Available: https://www.facebook.com/help/364957366911074. [Accessed 26 8 2013].

[26] E.-A. Baatarjav, S. Phithakkitnukoon and R. Dantu, “Group recommendation system for facebook,” in *On the Move to Meaningful Internet Systems: OTM 2008 Workshops*, 2008.

[27] P. Massa and P. Avesani, “Trust-aware collaborative filtering for recommender systems,” in *On the Move to Meaningful Internet Systems 2004: CoopIS, DOA, and ODBASE*, Springer, 2004, pp. 492-508.

[28] P. Kazienko, K. Musial and T. Kajdanowicz, “Multidimensional social network in the social recommender system,” *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, vol. 41, no. 4, pp. 746-759, 2011.

[29] A. K. Milicevic, A. Nanopoulos and M. Ivanovic, “Social tagging in recommender systems: a survey of the state-of-the-art and possible extensions,” *Artificial Intelligence Review,* vol. 33, no. 3, pp. 187-209,

2010.

[30] M. Wischenbart, S. Mitsch, E. Kapsammer, A. Kusel, B. Pr{\"o}ll, W. Retschitzegger, W. Schwinger, J. Sch{\"o}nb{\"o}ck, M. Wimmer and S. Lechner, “User profile integration made easy: model-driven extraction and transformation of social network schemas,” pp. 939-948, 2012.

[31] M. Rowe and F. Ciravegna, “Getting to Me--Exporting Semantic Social Network Information from Facebook,” 2008.

[32] C. Conroy, *Simple Semantic Mapping Over Time,* 2011.

[33] M. Rowe, “Applying semantic social graphs to disambiguate identity references,” *The Semantic Web: Research and Applications*, pp. 461-475, 2009.

[34] C. Conroy, R. Brennan, D. O’Sullivan and D. Lewis, “User evaluation study of a tagging approach to semantic mapping,” *The Semantic Web: Research and Applications,* pp. 623-637, 2009.

[35] N. Noy and M. Musen, “The PROMPT suite: interactive tools for ontology merging and mapping,” *International Journal of Human-Computer Studies,* vol. 59, no. 6, pp. 983-1024, 2003.

[36] S. McNee, S. Lam, J. Konstan and J. Riedl, “Interfaces for eliciting new user preferences in recommender systems,” *User Modeling 2003,* pp. 148-148, 2003.

[37] A. Bunt, C. Conati and J. McGrenere, “Supporting interface customization using a mixed-initiative approach,” in *Proceedings of the 12th international conference on Intelligent user interfaces*, 2007.

[38] E. Horvitz, “Principles of mixed-initiative user interfaces,” in *Proceedings of the SIGCHI conference on Human factors in computing systems: the CHI is the limit*, 1999.

[39] A. Paramythis, S. Weibelzahl and J. Masthoff, “Layered evaluation of interactive adaptive systems: framework and formative methods,” *User Modeling and User-Adapted Interaction,* vol. 20, no. 5, pp. 383-453, 2010.

[40] Wikipedia, “T-test,” [Online]. Available: http://en.wikipedia.org/wiki/Student's_t-test. [Accessed 1 8 2013].

Appendix A

Description

The appendix presents the project introduction and procedure information prepared for participants in the experiment.

Project Introduction

The research proposes an approach using meta-data embedded in web-pages, data mapping and Facebook data to enrich user browsing of the web. The Suggestion Tool was developed to explore the feasibility and efficiency of the approach.

The Suggestion Tool is a Firefox plugin and works with a Facebook application (it uses a dedicated test account, not your live FB account). The Suggestion Tool presents additional relevant sites or links when users are browsing schema.org annotated websites based on the contents of the current page. Facebook data such as relevant shares of their friends will be added to the Firefox sidebar. The idea is that this data will enrich the content that is shown alongside the webpages.

Experiment Procedure

1. Please sign the *Participants Consent Form* and read the *Participants Information Sheet*.
2. Please download videos and the Suggestion Tool from the link: https://www.dropbox.com/sh/j9p0jj7mnsqmvhq/qUGHpfNQHX. The videos are without sound.
3. Install the Suggestion Tool according to the video *tool-install.mp4*.
4. Have a look at the sample video *experiment-process.mp4*.
5. Choose **one** of the four books in the table below. Then go to http://www.powells.com and search for the book you choose. Please use the **same** book for different modes of the sidebar tool.
6. Open the Suggestion Tool.
7. With your participant number, follow the sequence in the table below.
8. For example, Participant 1:
   a. sets mode 1, looks at information in sidebar, fills out form
   b. sets mode 2, looks at information in sidebar, fills out form
   c. sets mode 3, looks at information in sidebar, fills out form

9. Fill out Questionnaire form

Please choose one of the four books in the table below.

No	Book Name	Author
1	A History of Ireland	Edmund Curtis
2	Football Genius	Tim Green
3	The Great Gatsby	F. Scott Fitzgerald
4	Gone with the Wind	Margaret Mitchell

Please follow the sequence

Participant Number	Order
1, 2, 3, 4, 5, 6	mode1--> mode2 --> mode3
7, 8, 9, 10, 11, 12	mode2--> mode3 --> mode1

Appendix B

Description

The appendix presents the grading form for grading items provided by the Suggestion Tool used by participants in the experiment.

Grading the Content Shown in the Sidebar

Participant Number:
Book Title:
How strong correlation do you think between the items shown in the sidebar tool and the book you are browsing? From 1 to 5, the bigger the number is, the stronger the relation is. Please select an appropriate one from 1 to 5 by placing a √ in the box.

Mode 1							
item1	Not Relevant At All	1	2	3	4	5	Strongly Relevant
(optional) why?:							
Item2	Not Relevant At All	1	2	3	4	5	Strongly Relevant
(optional) why?:							
Item3	Not Relevant At All	1	2	3	4	5	Strongly Relevant
(optional) why?:							
Item4	Not Relevant At All	1	2	3	4	5	Strongly Relevant
(optional) why?:							
Item5	Not Relevant At All	1	2	3	4	5	Strongly Relevant

(optional) why?:							
Item6	Not Relevant At All	1	2	3	4	5	Strongly Relevant
(optional) why?:							

Mode 2							
item1	Not Relevant At All	1	2	3	4	5	Strongly Relevant
(optional) why?:							
Item2	Not Relevant At All	1	2	3	4	5	Strongly Relevant
(optional) why?:							
Item3	Not Relevant At All	1	2	3	4	5	Strongly Relevant
(optional) why?:							
Item4	Not Relevant At All	1	2	3	4	5	Strongly Relevant
(optional) why?:							
Item5	Not Relevant At All	1	2	3	4	5	Strongly Relevant
(optional) why?:							
Item6	Not Relevant At All	1	2	3	4	5	Strongly Relevant
(optional) why?:							

Mode 3							
item1	Not Relevant At All	1	2	3	4	5	Strongly Relevant

(optional) why?:							
Item2	Not Relevant At All	1	2	3	4	5	Strongly Relevant
(optional) why?:							
Item3	Not Relevant At All	1	2	3	4	5	Strongly Relevant
(optional) why?:							
Item4	Not Relevant At All	1	2	3	4	5	Strongly Relevant
(optional) why?:							
Item5	Not Relevant At All	1	2	3	4	5	Strongly Relevant
(optional) why?:							
Item6	Not Relevant At All	1	2	3	4	5	Strongly Relevant
(optional) why?:							

Please write any further comments overleaf
Thank you for your help

Appendix C

Description

The appendix presents the user survey for participants in the experiment.

User Survey of the Suggestion Tool

Participant Number:
For each statement on this form, please select a number from 1 to 5 by placing a √ in the box. The bigger the number is, the more strongly you agree with the statement.

1. I think the sidebar can provide useful content when I'm browsing.						
Strongly Disagree	1	2	3	4	5	Strongly Agree
(optional) If not, why? :						
2. I can understand the data in the sidebar without clicking 'linkto'.						
Strongly Disagree	1	2	3	4	5	Strongly Agree
3. I think the delay of showing data in the sidebar is acceptable.						
Strongly Disagree	1	2	3	4	5	Strongly Agree
(optional) If not, why? :						
4. I like the way the sidebar presents content.						
Strongly Disagree	1	2	3	4	5	Strongly Agree
(optional) If not, why? :						
5. I think six items is an appropriate number to show in the sidebar.						
Strongly	1	2	3	4	5	Strongly Agree

Disagree						
(optional) If not, how many do you prefer?						
6. In general I like products with recommendations support.						
Strongly Disagree	1	2	3	4	5	Strongly Agree

Please write any further comments overleaf

Thank you for your help

Appendix D

Description

The appendix presents the user survey about system usability for participants in the experiment.

System Usability Scale

Participant Number:
For each statement on this form, please select a number from 1 to 5 by placing a √ in the box. The bigger the number is, the more strongly you agree with the statement.

1. I think that I would like to use this system frequently.						
Strongly Disagree	1	2	3	4	5	Strongly Agree
2. I found the system unnecessarily complex.						
Strongly Disagree	1	2	3	4	5	Strongly Agree
3. I thought the system was easy to use.						
Strongly Disagree	1	2	3	4	5	Strongly Agree
4. I think that I would need the support of a technical person to be able to use this system.						
Strongly Disagree	1	2	3	4	5	Strongly Agree
5. I found the various functions in this system were well integrated.						

Strongly Disagree	1	2	3	4	5	Strongly Agree
6. I thought there was too much inconsistency in this system.						
Strongly Disagree	1	2	3	4	5	Strongly Agree
7. I would imagine that most people would learn to use this system very quickly.						
Strongly Disagree	1	2	3	4	5	Strongly Agree
8. I found the system very cumbersome to use.						
Strongly Disagree	1	2	3	4	5	Strongly Agree
9. I felt very confident using the system.						
Strongly Disagree	1	2	3	4	5	Strongly Agree
10. I needed to learn a lot of things before I could get going with this system.						
Strongly Disagree	1	2	3	4	5	Strongly Agree

Please write any further comments overleaf

Thank you for your help