Improving Interlinking

by

Rajan Verma

Dissertation

presented to the

Trinity College, University of Dublin

in partial fulfillment of the requirements for the degree of

Master of Science in Computer Science

September, 2014

Supervisor: Dr. Declan O'Sullivan

Assistant Supervisor: Dr. Kevin Chekov Feeney

Knowledge and Data Engineering Group

School of Computer Science & Statistics

Declaration

I declare that the work described in this dissertation is, except where otherwise stated, entirely my own work and has not been submitted as an exercise for a degree at this or any other university.

Rajan Verma
August 29, 2014

Permission to lend and/or copy

I agree that Trinity College Library may lend or copy this dissertation upon request.

Rajan Verma
August 29, 2014

Acknowledgements

I am taking this opportunity to express my sincere gratitude to everyone who helped me in this research. Firstly, I would like to thank my supervisor Dr. Declan O'Sullivan for allowing me to pursue this research under his supervision. I want to thank him for his continued enthusiasm and motivation. I would also like to thank my assistant supervisor Dr. Kevin Chekov Feeney for his consistent support and guidance throughout this research.

Besides them, I want to thank all the volunteers who participated in the experiments of this research. Last but not the least, I would like to thank my friend Neeraj Dixit for his valuable suggestions and support, my parents for their consistent support throughout my life.

Abstract

In the past few years there has been a considerable amount of growth in the Web of Linked Data. A large amount of the information available on the web is in the unstructured form. With the increase in the number of data resources, linking such heterogeneous resources has now become a challenge. Even though, there are many automatic tools available to interlink heterogeneous data sources, the links generated by these approaches are often found not to be precise. Semi-automatic tools have been proven to be more efficient and precise. However, not much attention has been given to the user interface of these tools. People often have found it difficult to use these tools. This dissertation proposes an approach to interlink heterogeneous data sets using a semi-automatic approach. The dissertation also determines how users behave differently to different user interfaces for data-interlinking. An evaluation approach was formulated in order to evaluate the efficiency, effectiveness and accuracy of the implemented system based on the proposed approach. In addition, the evaluation also determines how different user interfaces influence the task of data interlinking in linked data applications.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In recent years, the methods provided by Linked Data have now become a standard way to publish documents and datasets on the semantic web [Bizer2009]. The goal of Linked Data is to publish all the information on the web in a common format and create links between different information sources [Volz2009]. However, linking such information resources is a time consuming process and requires a lot of effort [Turnbull2007, Wolger2011]. In order to automate this task, various tools have been developed which can create links in no time. However, the quality of links generated by these tools is not of a high standard. Semi-automatic tools have been proven to be a better alternative for generating high quality links [Lehmann2013]. Even so, still not many people are using Linked Data applications. This gives rise to the questions on the quality of user interfaces of Linked Data applications [Heim2011, Partarakis2009 and Mac´ıas2012]. This dissertation presents the detailed study on how the user interface of a semantic application can influence the data interlinking task.

This chapter will first define the background and motivation of the dissertation. Then it will define the problem definition and research goal of the dissertation. It will conclude with the structure of the rest of the document and some introduction to remaining chapters.

1.1 Background and Motivation

Today, a large amount of information available on the web is in the form of static web pages. These web documents are linked to each other through hyperlinks. The information available in the web documents is present in raw form and is usually made available from a database. Machines can search words in the documents, but other than finding a word, machines cannot get the meaning out of these documents. To make machines understand the meaning of the data available in these documents, Tim Berner Lee introduced the concept of Linked Data [Bizer2009].

Linked Data is about creating links between different data sources on the web. It recommends best practices of publishing and linking data on the web which allows us to explore the data on the web in a more efficient way. Linked data is constantly growing at a very rapid rate and now almost all big companies are moving towards this technology [Bizer2009].

A variety of methods have been developed for publishing the data available on the web. The data available on the web is often represented in RDF (Resource Description Framework) format, which represents data in a simple format as compared to XML [Hsu2012]. It is found that linking data in RDF format is much easier and fast. An entity defined in RDF format is defined by using RDF triples: subject, predicate and object. Objects are often defined as property value, which are mapped to a subject using predicate. Predicate defines the relationship between subject and object. Each of the entity is identified by using unique URI's [Bizer2009].

However, linking such heterogeneous resources is a difficult task. Earlier, people used to create links manually. It was a very time consuming process, and it requires a lot of effort from knowledge engineers [Raimond2008]. Also, it was difficult to find people with enough background domain knowledge. In order to address these issues, people have now come up with automatic tools, which create links automatically within no time. Wolger et al [Wolger2011] describe the latest available tools used for generating links automatically. For creating links or traversing through the links a structured query language is used called SPARQL [Prud'hommeaux2006]. SPARQL is specifically designed for documents in RDF format. SPARQL is the most popular query language used for RDF datasets [P´erez2009].

However, Celino et al [Celino2012] and Hildebrand et al [Hildebrand2012] found that the links generated by automatic approaches are often found not to be precise or of sufficient quality. Even though there are supercomputers these days, but still some tasks which are difficult for computers are trivial for humans. Such tasks can be undertaken by users without much effort. Wolger et al [Wolger2011] describe various tools that involve humans in contributing towards the semantic web. It has been found that tools that involve humans often results in higher precision links than automatic tools [Celino2012, Hildebrand2012]. Semi-automatic approaches generally involve humans for the validation of links generated by automatic tools.

One of the motivations of this dissertation was to find how semi-automatic approaches can be helpful in creating links with third party data sets. There has been a significant amount of research in this area, but still not many people are using Linked Data applications. Linked data applications are complex and often require people with some background domain knowledge.

Another motivation for this dissertation was to discover what problems user generally face while using semantic tools and how using human computer interaction (HCI) principles one can improve the interaction of users with linked data applications. While developing such applications one should keep in mind the needs of an end user. García et al [García2006] describe how the usability of semantic applications can be improved by following User Centered Design approach. However, not everyone likes the same user interface. Some people might not like what other people like. This process is called Design Space Exploration (DSE) [Saxena2010]. The goal of Design Space Exploration is to look for the alternate designs which are ideal for the given set of functions. Also, it is difficult to explore complex semantic data in traditional user interfaces [Ma2000].

1.1.1 Dacura Framework

In order to fully leverage the data available on the web, high quality, consistent and more correct data sets are needed. Dacura framework [Feeney2014] aims at creating structured high-quality datasets using a large amount of information available on the web that can be consumed easily by third parties and linked with other data sets. It ensures the quality of data by using both manual and automatic approaches. Dacura workflow involves extracting the data from existing web sources using both automatic and manual approaches. Data harvesters extract the information from unstructured sources in the form of reports. Domain experts then generate facts based on these reports. This step is then followed by validation of the information either with the help of consumers who can correct or give suggestions on the collected data. Finally, the data architects decide how to represent such information. Figure 1.1 shows the Dacura workflow diagram and the roles involved in it.

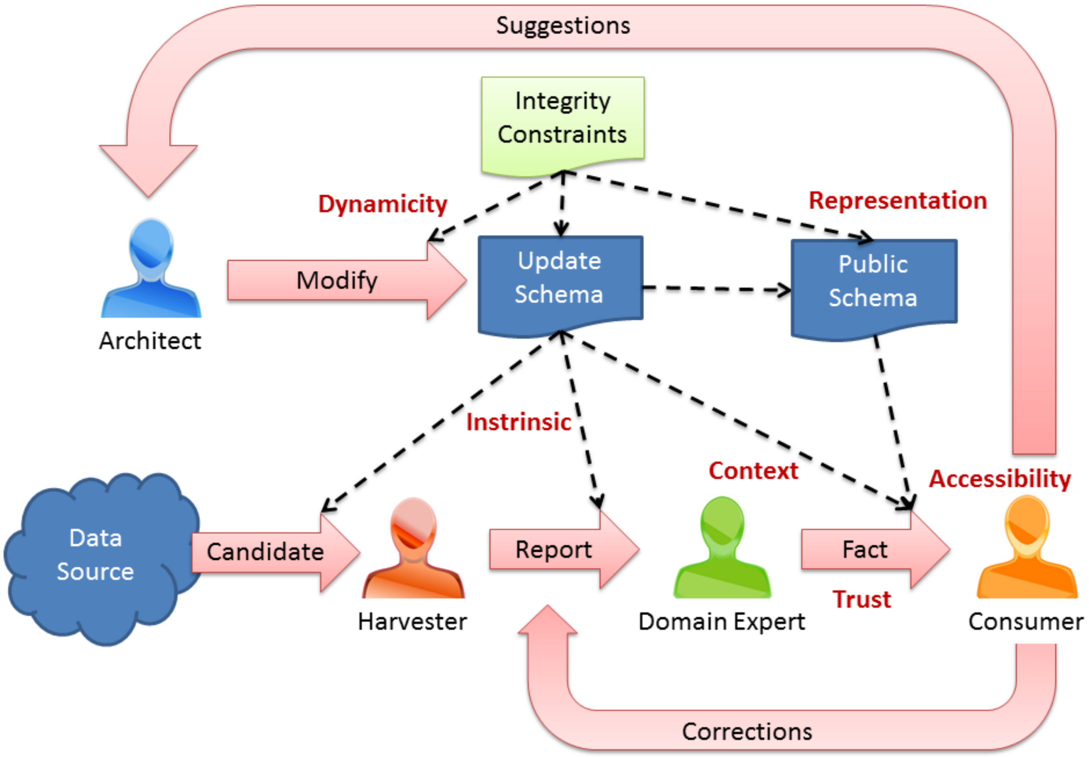


Figure 1.1: Dacura workflow [Feeney2014]

The data extracted from existing web sources are encoded in the form of RDF graph and kept in triplestore. Now, the next step is to link the locally managed dataset with other data sets available on the web. This dissertation aims to link locally collected data set with the DBpedia, which is a centrally linking hub of the web of data [Kobilarov2009]. The dissertation proposes a novel approach to perform this task with an emphasis on improving the linking experience for the users.

1.2 Research Question

This research was conducted as a part of an ongoing KDEG project – Dacura [Feeney2014]. Dacura workflow has been divided into four phases: Data Harvesting, Knowledge enrichment, Expert analysis and Publishing and Consumption. This research was undertaken as a part of the knowledge enrichment phase of Dacura. The knowledge enrichment phase includes annotating, refining and linking with other data sets available on the web. Thus for this dissertation research, the following research question was posed:

'How to build Linked Data applications and tools to interlink different data sets efficiently, easily, effectively and more pleasingly in a manner as to provide users with the best experience?'

Data interlinking is the process of navigating through different data sources and search out the additional information [Bizer2009]. The term efficient in this question refers to if the users will be able to accomplish the data interlinking task with least amount of effort. The term easily in this question is whether the user will perceive data interlinking to be not complex and is easy to achieve. The term effectively examines whether the users will be able to create interlinks sufficiently between different sources. The term pleasingly in this question refers to if the process of interlinking will be acceptable for a user. Users will be provided with alternate user interfaces to assist them in examining complex semantic data and exploring the available options.

1.3 Contributions

The main contribution of this dissertation is in evaluating how well users perform the same task with map and table user interface objects in the web of linked data. These findings are a direct contribution to the state of the art, given the limited number of previous studies. This would contribute to the state of the art in three ways:

1. This research exemplifies the possibility of improving the usability of Linked Data applications by comparing multiple interfaces.

2. This comparison of multiple interfaces can also be used to demonstrate how the process of interlinking different data sets can be improved.

3. The research exhibits the examination of alternate user interfaces in the field of Linked Data.

1.4 Technical Overview

This section gives an overview of approaches followed during the development of the system. First, it describes the technical approach followed and then it presents the evaluation of the developed system.

1.4.1 Technical Approach

Before starting the actual implementation of the system, already available data linking systems were analyzed in the state of the art. The major focus in the review was on the user's interaction with linked data systems. It was found that while designing linked data applications not much attention has been given to the design of the user interface. In this dissertation, the development of the systems happened in two iterations. In the first iteration, system finds the places from DBpedia [Kobilarov2009], which are similar to historical location of Dacura data set. Two interfaces were chosen to represent the extracted data: map and table. In order to deal with database inconsistencies like incorrect and misspelled locations, Google places auto-complete API was used, which gives suggestions for the entered location. The first experimental results helped in identifying the issues that users faced in the context of undertaking the interlinking task within the Dacura linked data application.

In the second iteration, improvements were made to the system based on the feedback given in the first experiment. Rather than displaying interfaces on different pages, both the interfaces were displayed on the same page. Also, the approach for extracting the information was also changed to improve the performance of the system. Detailed explanations of design decisions made during the development are discussed in the third chapter of this document. The second experiment was conducted to evaluate efficiency, effectiveness and usability of the developed system. In addition, the experiment also evaluated, how the users responded to different interfaces.

1.4.2 Evaluation Overview

A task-based evaluation approach was chosen in order to answer the research question. Each of the two experiments was carefully designed to ensure that the research question could be answered from the experimental results and data collected. Participants were asked to link historical records of Dacura data set with DBpedia. The linking was done based on the location attribute of each record. For each record the system finds the similar places from Dbpedia and presents them in two interfaces: map and table. Participants were asked to find potential matches between Dacura and Dbpedia places, using both the interfaces. In the first experiment participants were asked to link records using both the interfaces. In the second experiment, participants were divided into two groups: map and table interface group. Participants were asked to link records using only one interface based on their group. Generally, the efficiency of the system is evaluated based on the time spent by the user for the given task and effectiveness of the system is evaluated based on the accuracy of the task undertaken by the users. Usability of the system was evaluated based on the user's response to the questionnaire and their feedback. Data collected in log files recorded the users' interactions with the system.

1.5 Dissertation Overview

This section details the structure of the rest of the document.

1. Chapter 2 presents the state of the art, which discusses the available automatic and semi-automatic approaches for data interlinking, problems associated with such approaches and finally work done in incorporating HCI principles in the web of linked data.
2. Chapter 3 details the design of the developed system, challenges faced during development and steps taken to tackle those challenges.
3. Chapter 4 describes the implementation of the whole system in detail. It details the technologies, technical architecture and algorithms used to implement the system.
4. Chapter 5 presents the experiments conducted to evaluate the developed system. This chapter deeply describes the procedure and evaluation results of the experiments.
5. Chapter 6 presents the future work and conclusion.

Appendix A contains the interview questionnaire used in the experiment.

Appendix B contains the instruction material given to the participants before experiment.

Appendix C contains the table of contents of the accompanying DVD media.

Chapter 2

State of the art

This chapter introduces the ongoing work in the field of data interlinking in Linked Data. In particular, the emphasis is placed on the techniques and approaches taken by people to interlink heterogeneous data set. The first part of this chapter gives you an overview of the Data interlinking problem along with a motivating example. Then an overview of available automatic tools is presented with emphasis on quality of links generated by these tools. Next, an overview of semi-automatic approaches is presented focusing on the approaches taken to motivate users to contribute and the problems associated with the user interface of semi-automatic tools. Finally, the next section gives a summary of the role of human user interaction principles in the field of Linked Data with emphasis on usability aspects of Linked Data applications.

2.1 Overview of Data Interlinking Problem

One of the biggest challenges in the semantic web is to exploit the data available on the web. Linked Data introduced by Tim Berner Lee, defines the best practices to publish the data on the web, so that information on the web can be linked reliably with other information sources. Linked Data use RDF (Resource Description Framework) format for representing information in a dataset [Bizer2009]. It represents a common framework so that the information can be exchanged efficiently between different applications. I-Ching et al [Hsu2012] describe the benefits of representing project related information in RDF format. There are many automatic tools available which converts an unstructured data into structured format i.e., RDF format. LODifier is one of those tools which convert data from text to RDF format [Augenste2012]. LODifier uses Semantic web vocabulary and word sense disambiguation to extract the entities from unstructured data and represent them in RDF format.

While there are numbers of tools available which can convert data on web to structured format (RDF), interlinking the resulting data sources is still a big challenge for both humans and machines [Wolger2011]. Interlinking millions of links is a time-consuming task and requires a lot of effort from humans. Lack of incentives and non-user-friendly tools are some of the reasons why interlinking process is a challenge for humans [Cuel2012]. Machines can create links automatically, but it is difficult for them to differentiate between different domains [Wolger2011].

2.1.1 Motivating Example

Figure 2.1 shows an entity from the Dacura triple store and the list of similar entities from Dbpedia triple store. Both these datasets use different vocabularies to represent the same entity. It can be seen from the Figure 2.1, for an entity, there is only one potential match between both the datasets. However, if an entity has different meaning in different domains, it becomes difficult for machines to find potential matches. There is no single appropriate algorithm that can deal with records of all domains. Such entities can easily be linked by humans. However, for humans, it takes a lot of time to go through each and every record in order to find the best match. Thus a data interlinking approach is needed which create links efficiently, effectively and easily between different data sets.

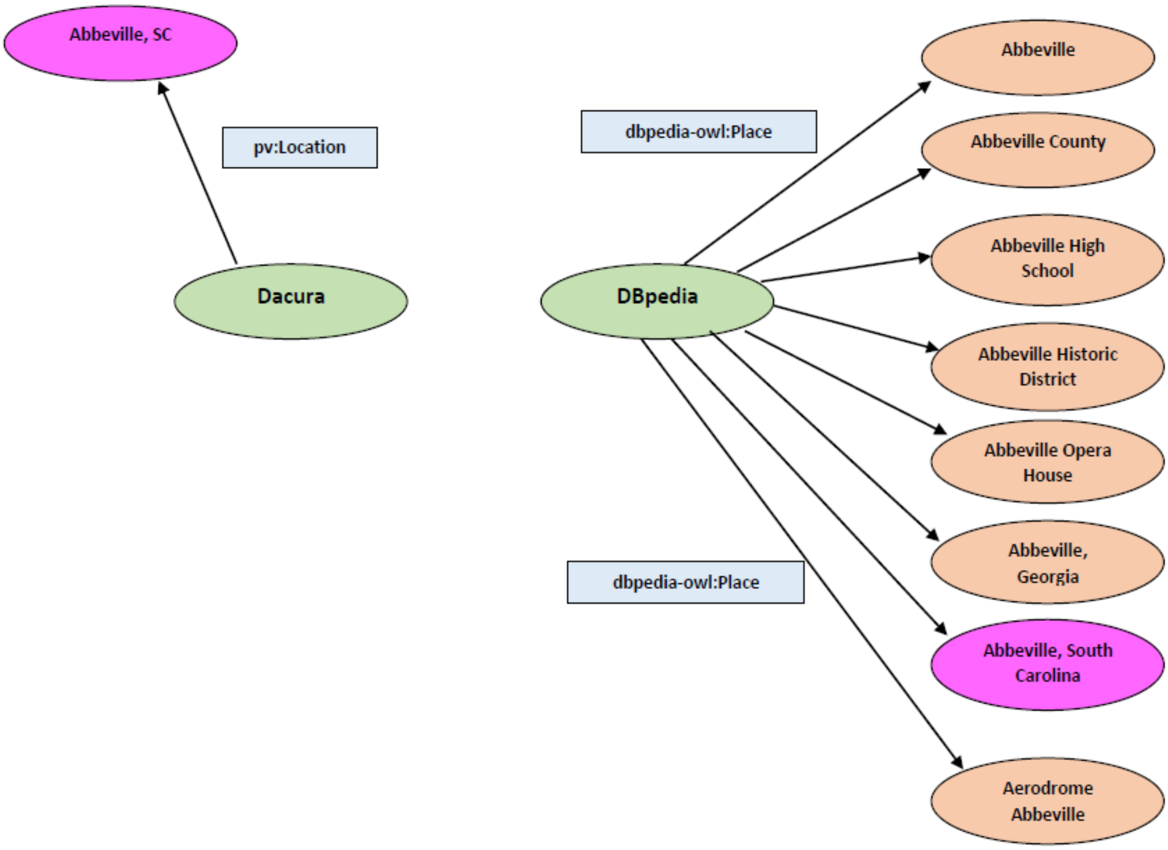


Figure 2.1: Entities of Dacura and Dbpedia data set

2.1.2 Difficulty in interlinking data

According to Alexander et al [Alexander2009], during the first phase of Linked Data focus was mainly on publishing data and high-quality practices. But now the focus has changed to the user interface, application's usability and data available in structured form. The data linking task is now the important task among all since the number of semi-structured and structured data on the web is continuously growing [Halb2008]. The data linking process mainly deal with three types of problems: Ontology matching, Dirty data and duplicate results [Ferrara2011 and Fellegi1969].

1. *Ontology Matching* - The datasets containing similar entities are usually defined using different ontologies. It becomes necessary to bring together these ontologies before starting linking process.

2. *Dirty data* - A dataset may suffer from the problem of dirty data. It is difficult to compare corrupt data with structured data. *Value matching* approach is generally used for matching

entities of different data set. *Value matching* approach compares two individuals on the basis of their object's value. If the value of an individual is misspelled in any of the data sets, those two individuals are considered different, even though they are referring to same real world object.

3. *Duplicate results* - Another problem with data linking is to detect the duplicate records. This problem is often called record-linkage problem.

Ferrara et al [Ferrara2011] describe the main problems associated with data interlinking. According to them, the data interlinking process generally shifted to three notions of identity. According to the first notion of identity, the two entities are similar if they are referring to same real world object. According to the second notion of identity, the two entities are similar if one can be substituted for another without altering the significance of the expression. The third notion of identity says two entities are same if one entity is superimposed data on another entity like ISBN code for books.

However, the problem with the notion of identities in the world of interlinking is that the interlinking process often occurs on the basis of similarity and sometimes it becomes difficult to differentiate between the concept of similarity and identity. Consider an example: Suppose there are different editions of the same book. It is possible that an object's description in different books might refer to the similar thing. But since the editions of the books are different, the object's description should be considered different. Thus, one cannot differentiate entities only on the basis of 'identity'.

Fortunately, a lot of work has already been done in the area of linking. Numbers of tools are available these days that can create links automatically. These tools take the two data sets as input and create links between them. Wolger et al. [Wolger2011] describe the latest available tools used for generating links automatically. The next section of this chapter discusses the techniques followed in these tools for interlinking data sources.

2.2 Data Interlinking – Automatic approaches

This section gives an overview of automatic data interlinking systems. Five automatic interlinking systems have been found which are discussed in this section: KnoFuss [Nikolov2008], OAI-ORE protocol [Carlos2010], Interlinking Social Distributed Graphs [Rowe2009], CaMiCatzee [Hausenblas2008] and SERIMI [Araujo2011]. The main aim of this examination is to find out the data interlinking approaches used in these systems. Section 2.2.6 presents the detailed comparison of these tools and critique of each of these tools.

Table 2.1: Available automatic linking tools

Interlinking System	Short Description
KnoFuss	KnoFuss focus on instance level integration of OWL ontologies data sets.
OAI-ORE protocol	This system focuses on interlinkingheterogeneous Geo-spatial services to build scalable Linked Data applications.
Interlinking Social Distributed Graphs	This system focuses on exporting social data in Linked data format and linking those data sets.
CaMiCatzee	CaMiCatzee focus on creating links between multimedia data sets and other information sources such as foaf.
SERIMI	It interlinks by matching instances of different data sets without prior knowledge of the domain or schema.

2.2.1 KnoFuss

Nikolov et al [Nikolov2008] developed an architecture called KnoFuss which does instance-level integration of datasets. KnoFuss is particularly designed for data sets which are structured according to OWL semantic ontology. KnoFuss performs three main subtasks of data fusion: Co-referencing, Conflict detection and Inconsistency resolution. Co-referencing is a phenomenon of generating mapping between individuals which are supposed to be similar. Each of these tasks can be performed by using both general and domain-dependent methods.

Knofuss consist of two main components: A library of methods, complex algorithms to deal with the data and fusion ontology, which guides through the fusion process. The linking process is performed as follows. A source of RDF data is given as its input, on which all the fusion tasks are performed. The output links from these methods are then stored in a new Knowledge database.

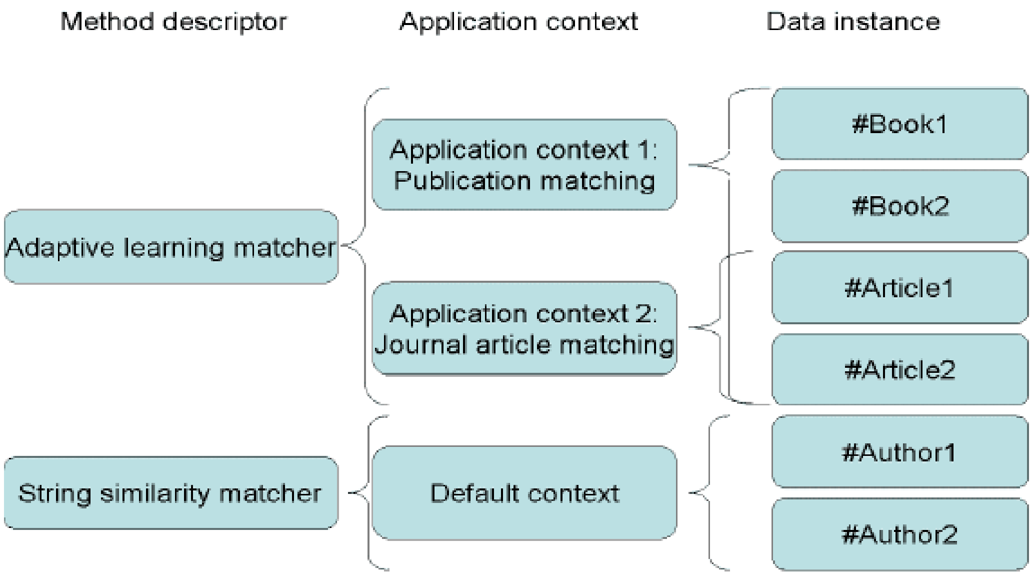

Method descriptor
Application context
Data instance
Adaptive learning matcher
String similarity matcher
Application context 1: Publication matching
Application context 2: Journal article matching
Default context
#Book1
#Book2
#Article1
#Article2
#Author1
#Author2

Figure 2.2: Method Selection process [Nikolov2011]

Before starting the fusion process a set of methods are selected depending on the domain of knowledge data set. The method descriptor object decides the methods to be performed on input dataset. Once the methods are selected, the parameters of these methods are set according to the given data. For example, string matching algorithm can be applied to almost every dataset, but the performance and accuracy of this method may differ if applied to an object of different set. Application Context is responsible for setting up these parameters. The assignment of these parameters requires a significant amount of effort. In order to automate this process, the parameters are often generated using machine learning algorithms.

2.2.2 OAI-ORE protocol

Carlos et al [Carlos2010] discuss the approach of linking the heterogeneous geospatial datasets to build scalable and cross domain web applications. Geospatial datasets are available in large quantity on the web and are constantly growing. These datasets are not only available through SDI (Spatial Data Infrastructures) but also produce by the millions of users on the internet through social networking websites like Facebook, Twitter, Flickr etc. The good thing about using this information is that non-expert users provide up to date information on the web. By linking these heterogeneous Geospatial datasets, the users can access structured datasets for their cross domain Geospatial applications.

Here, Carlos et al [Carlos2010] mainly concentrates on the suitable representation of geospatial services, which then can be used by users to build a collection of geospatial services for their applications. The approach to link geospatial datasets is based on two techniques: one is OAI-ORE protocol and other is principles of Linked Data. The OAI-ORE protocol is generally used in the Digital Library Domain which is based on the principles of Linked Data. It is assumed that each of the resource is identified by a URI and these resources can be connected to each other through typed relationships. This model also allows the use of existing vocabularies like RDF schemas and Geonames. It is always good to use existing vocabularies rather than creating your own, since it is easy to link datasets which are based on well-known vocabularies.

The figure 2.3 shows the architecture of the proposed approach. The architecture is divided into three layers: application, service and data layer. The geospatial resources are first modeled according to OAI-ORE model and then passed through an RDF / XML synthesizer, which converts data to RDF format.

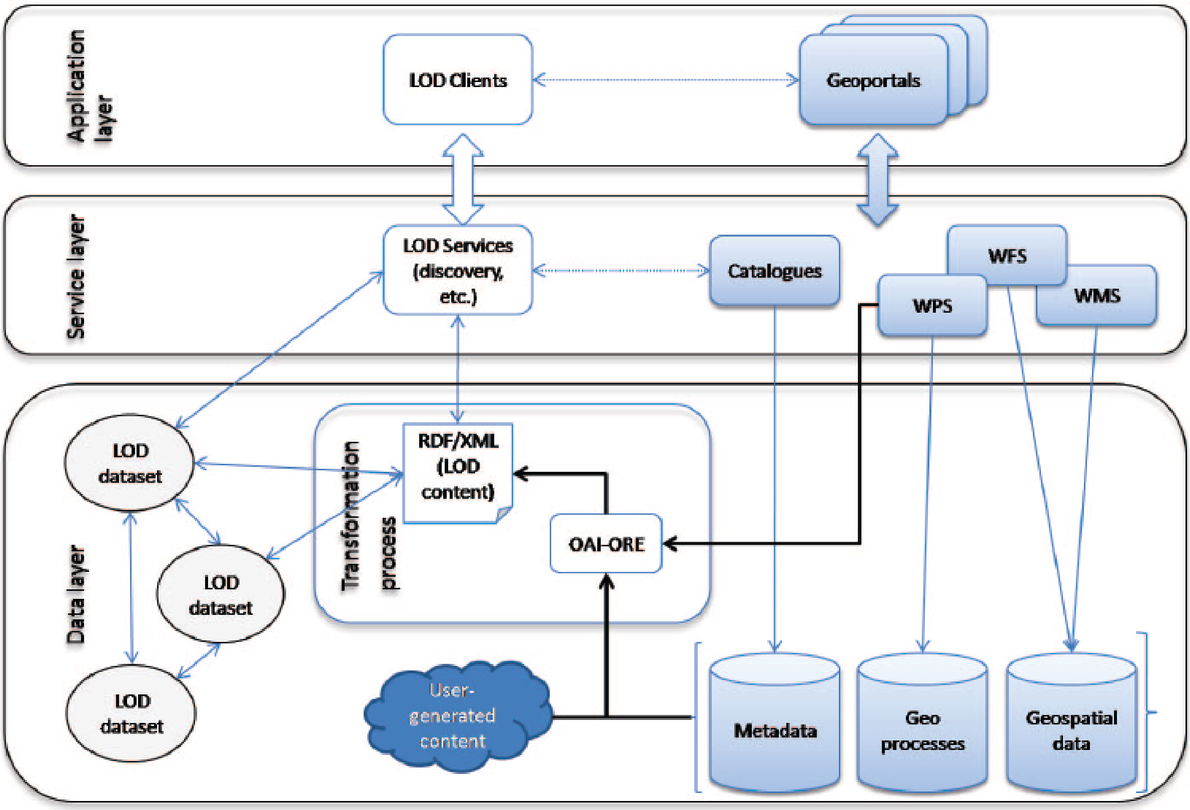


Figure 2.3: Overview of Architecture [Carlos2010]

Some assumptions have been made while designing this architecture like the geospatial services can be addressed using URI's, OAI-ORE protocol can be used since the geospatial datasets are a type of web resource. The OAI-ORE model represents the aggregation of geospatial resources which are then further modeled into a set of interlinked resources. Another assumption Carlos et al [Carlos2010] made is that all the geospatial datasets can be referenced using URI's since they are part of web resource. This may not be true as there are datasets which are available in unstructured form on the web and cannot be referenced through URI's. Such datasets cannot be linked with other datasets.

2.2.3 Interlinking Distributed Social Graphs

Rowe et al. [Rowe2009] designed a system which exports the data from existing social web services into RDF format and create links between them. First the data from Facebook is collected. The data returned from Facebook web services is in XML format. XML format data is then mapped to FOAF ontology. FOAF ontology is appropriate for capturing social data as it provides you with extensive description of identity information. Similarly, the data is exported from Twitter in the form of a graph.

In order to link exported data sets first the name of entities is matched. Each entity with subject foaf:Person has a property foaf:name which represents the name of a person. The names are compared using Lavenstein matching algorithm. If the names matched, the other properties like phone number, email-id and homepage (identifier properties) are matched. This step is done in order to increase the confidence to confirm two entities are same. Geographical information is also one of the factors that have been used while deciding similarity between the entities of exported data sets.

2.2.4 CaMiCatzee

"Catch me if you can" CaMiCatzee is a multimedia interlinking system which creates links between different multimedia sources on the web. In Flickr, pictures can be annotated with user information, entities in picture and user's comment. However, nothing has been done to link this information with other information sources such as Foaf. CaMiCatzee takes the advantage of the finely grained annotation ability of Flickr to annotate images of users semantically. Users just have to submit their Foaf document or enter their name and location in CaMiCatzee. The system will fetch the pictures of that user from Flicr on the basis of the attributes defined in Foaf document or other provided information. The annotated objects in the pictures are shown by link rdfs:seeAlso, which has been offered using XHTML+Rdfa. CaMiCatzee makes use of User Contributed Interlinking [Hausenblas2008a, Hausenblas2008b] to link multimedia data sets on the web. Also, the rdfs:seeAlso links can be consumed by machines and humans. The figure 2.4 shows the architecture diagram of the system.

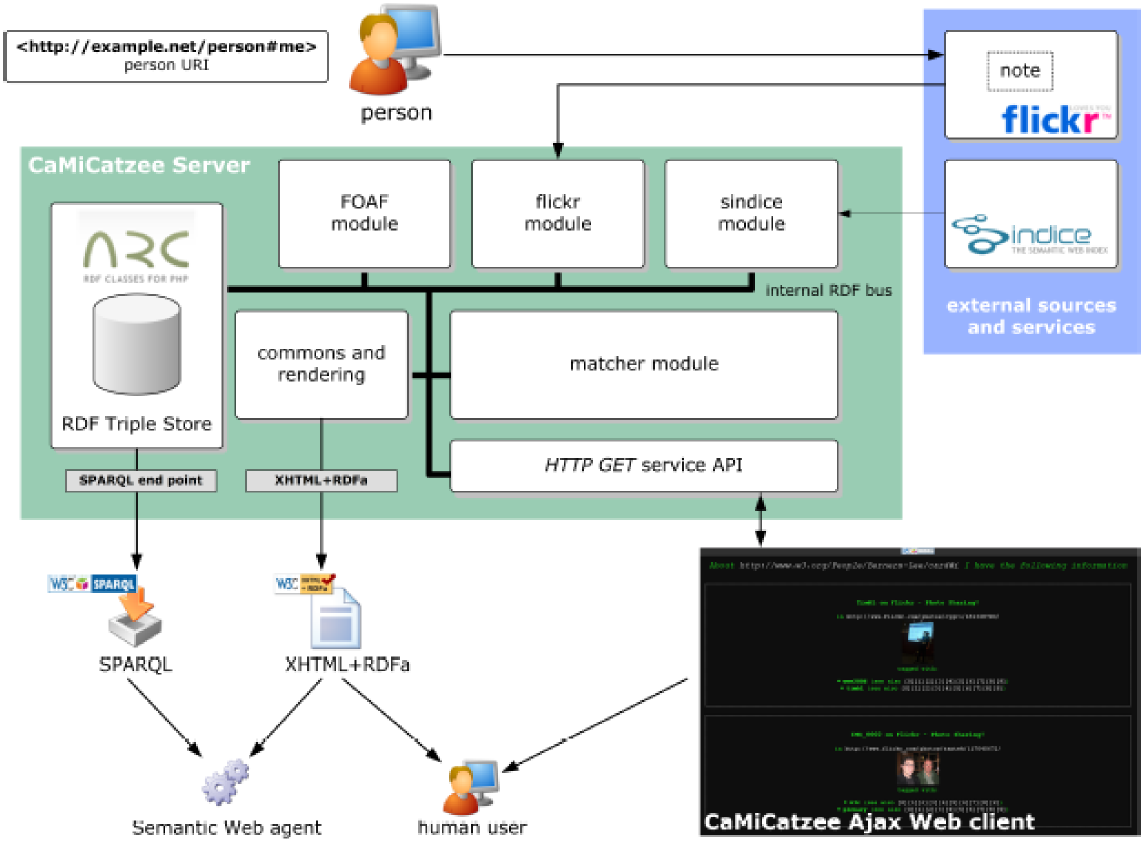


Figure 2.4: CaMiCatZee Architecture [Hausenblas2008]

2.2.5 SERIMI

SERIMI is a method to interlink similar entities of two different data sets. SERIMI's interlinking process is divided into two phases. In the first phase, an instance of a source data set is searched in a target data set. The instances are matched based on string matching techniques (e.g. [Levenshtein1996], [Hamming1950]). These matching techniques return a list of entities from the target data set that is similar to the instance value of label data set. The second phase of SEIMI is disambiguation phase. In this phase filtering of unwanted entities takes place. The higher the similarity between the entities of two resources the higher the probability those two entities refer to the same objects.

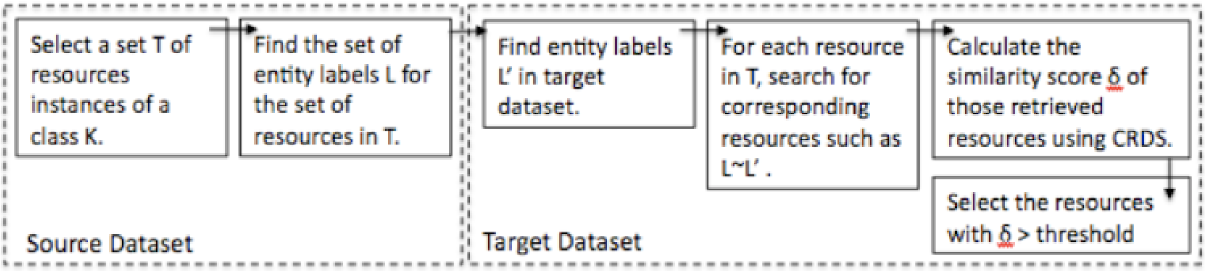

Select a set T of resources instances of a class K.
Find the set of entity labels L for the set of resources in T.
Source Dataset
Find entity labels L' in target dataset.
For each resource in T, search for corresponding resources such as L~L' .
Calculate the similarity score δ of those retrieved resources using CRDS.
Select the resources with δ > threshold
Target Dataset

Figure 2.5: SERIMI's Algorithm [Araujo2011]

SERIMI uses a term pseudo-homonyms for the entities that are found similar in the first phase of the algorithm. Pseudo-homonym set may contain entities that contain instance labels of other classes. For example, if someone search for 'Indiana' in target data set, he may find that the pseudo-homonym set contains instances like universities, schools etc., which are located in 'Indiana'. It becomes necessary to filter these records from the pseudo - homonym set to avoid confusion. SERMI uses RDS (Resource Description Similarity) approach to filter out such instances.

SERIMI calculates similarity score between entities on the basis of RDS approach. If the similarity score value is greater than a threshold value, then system will select the entity in pseudo-homonyms set for interlinking process.

2.2.6 Analysis

This section compares the five automatic interlinking tools discussed in the last section. The comparison is mainly made on the matching techniques and domain of the interlinking tools. Matching techniques details the algorithms used in identifying the similar entities of different data sets. Most of the tools have used string matching algorithms for performing this task. Domain criteria define the domain of the data set for which the tool has been developed. These two criteria have been used for the comparison to show what matching techniques are used for interlinking data set of different domains.

Table 2.2: Comparison of automatic linking tools

Interlinking Systems	Matching Techniques	Domain
KnoFuss	String matching algorithms such as Jaro-Winkler	Any
OAI-ORE protocol	Aggregation by linking	Geo-spatial data
Interlinking Social Distributed Graphs	Levenstein string matching, Graph matching	Facebook, Twitter graphs
CaMiCatzee	Semantic indexer	Multimedia data
SERIMI	String matching and Resource Description similarity methods.	Any

2.2.7 Problems with automatic approaches

Despite of having so many efficient automatic interlinking tools, it has been found that the links generated by automatic techniques lack precision and quality [Lehmann2013]. Even though the tools like KnoFuss and SERIMI are designed to link data sets of any domain, the links created by these tools are often found not to be précised. These tools work on some predefined assumptions. KnoFuss assumes that both the input data sources are constructed using same ontology. Another issue with KnoFuss is method selection process, which currently is based on the class of an entity. However, it is found that the performance of the method mainly depends on the data format of the data source. OAI-ORE protocol assumes that all the geospatial datasets can be referenced using URI's since they are part of web resource. This may not be true as there are datasets which are available in unstructured form on the web and cannot be referenced through URI's. Such datasets cannot be linked with other datasets.

SERIMI performs poorly in case of missing alignments in the reference set. Other factors that contribute to the poor performance of SERIMI are data inconsistencies and inaccurate alignments in the reference set. Thus, it is impossible to achieve 100% accuracy with automatic approaches. The algorithms used in GNAT, takes a long time to create a playlist out of the available information on the web. As a side effect of this, it's not feasible right now to take this process to the next level.

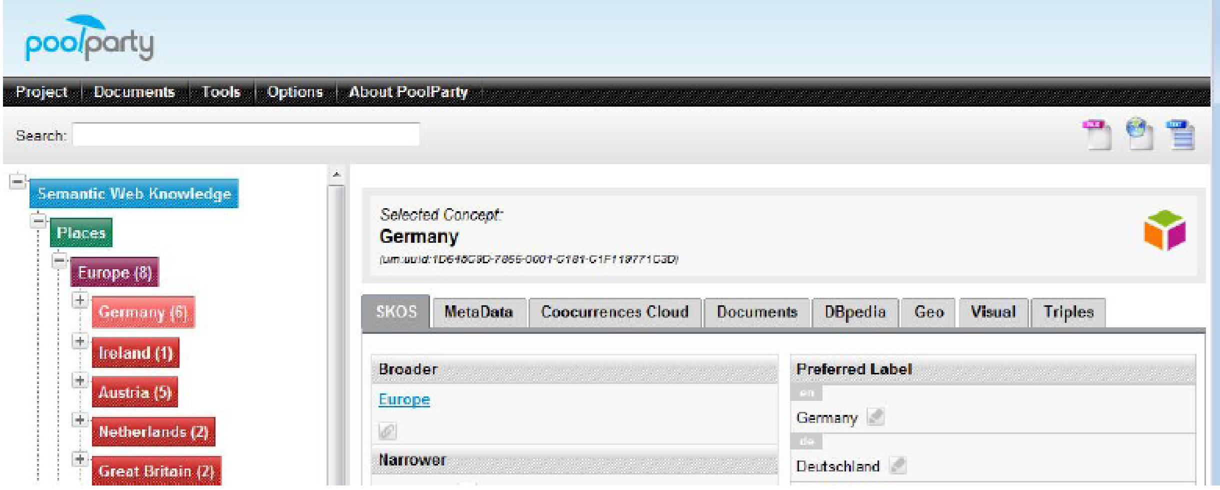
According to Siorpaes et al. [Siorpaes2008], tasks which are often difficult for machines are trivial for humans. Humans with little training can master those tasks. Such tasks may include ontology matching, annotation generation in Linked data. Human computation can be used to create links of high quality. Ahn [Ahn2009] describe how human computation can be used to perform tasks that computers cannot. CAPTCHAs on the internet help to prevent automatic programs to hack online services. CAPTCHAs are random alphanumeric characters ask to fill by the users while creating a new email account or downloading songs from the internet. It ensures the security of the systems on internet.

Another way of harnessing human brain powers is by the use of games [Ahn2008]. In a game two users can be asked to annotate an image with correct descriptions. If both the users use the same label for annotating an image, they receive points. Such tasks are still difficult to perform using computer vision techniques. The next section of the state of the art discusses how semi-automatic approach can be useful in creating high-quality links.

2.3 Semi-automatic Approaches

Semi-automatic interlinking techniques involve both humans and machines for the data interlinking task. Machines can automatically create a link, but it is hard for them to validate those links. Human computation can be used for the validation of links. Users can select the most appropriate link from the set of links suggested by the machines. Wolger et al [Wolger2011] describe the available semi-automatic interlinking tools.

PoolParty [Schandl2010] is a thesaurus management tool, which uses the concepts of Linked Data to maintain and manage thesauri. The accessible user interface of this tool allows domain engineers to manage thesaurus without knowledge of semantic languages. Users play a very important role in managing and updating thesauri. When the user use this tool to analyze documents, it not only maps the concepts in that document which are already part of thesaurus, but also detect unknown concepts in that document. These unknown or new concepts can be approved and updated by thesaurus manager.



Selected Concept:
Germany
(urn:uuid:1D648C9D-7856-0001-C181-C1F119771C3D)

Preferred Label: Germany, Deutschland

Figure 2.6: PoolParty User Interface [Schandl2010]

PoolParty can be linked with the other linked data sources such as DBpedia. It uses an instance or a label matching technique to return the list of suitable matches from DBpedia. The user can select the resource which best matches the labels in the thesaurus.

LinQuer [Hassanzadeh2009] is another tool which allows humans to take part in the data interlinking process. LinQuer allows users to write discovery methods that suits best for their domains. This tool provides various linking methods and facility to write methods such as User Defined Functions. Users can change the existing methods and queries according to their own requirements. Here, users are not involved in the validation of the links; rather they are involved in writing linking algorithms and queries.

2.3.1 Problem

The data interlinking task may involve validation of millions of links. Validating millions of links is a time consuming task. Users may get bored of doing the same thing over and over again. There has to be some motivation for the users to contribute. Cuel et al [Cuel2012] describe how one can build addictive semantic applications. In order to build addictive semantic applications first one need to understand the interests of the users, who are going to use such applications? Every user has different motivations for doing the work. Some people work to impress others, some people work for the incentives and some people works for the community. The user's interest should be considered while designing semantic applications.

Siorpaes et al [Siorpaes2008] and Ahn et al [Ahn2008] propose a new concept for motivating users' i.e 'Games with a Purpose for semantic web'. People of almost all the age groups love playing online games. Games can be used for motivating users to contribute. These games are designed in such a way that the users those who are playing these games, don't know that they are actually creating links at the back-end. However, several things need to be kept in mind while designing such games. There should be clarity of tasks that needs to be done at the back-end. Applications should have a mechanism to detect untrustworthy users playing these games. Such users can degrade the quality of links created by these games. Ahn et al [Ahn2008] describe techniques to identify player's collusion.

2.3.2 Semi-automatic Interlinking tools

This section discusses semi-automatic interlinking tools. Four semi-automatic tools have been found which are discussed in this section: UrbanMatch [Celino2012], Taste It, Try It [Cuel2012], VeriLinks [Lehmann2013], Waisda [Hildebrand2012]. The main focus of this examination is to find how humans have contributed to interlinking tasks and what approaches have been taken to motivate them. Section 2.3.3 presents the comparison and critique of these tools.

Table 2.3: Semi-automatic interlinking tools

Interlinking System	Short Description
UrbanMatch	Mobile application which links the point of interest in one data set to images in other data set.
Taste It, Try It	An application which provides information about the ambience and quality of food of nearby restaurants.
VeriLinks	A multiplayer game which asks users to verify links.
Waisda	String matching and Resource Description similarity methods.

UrbanMatch

Geo-spatial data is available in large quantity on the web. According to [Celino2012], by using Geo-spatial data a significant contribution can be made in building Smart Cities. There can be many objectives for linking these data sets, like linking road topography with traffic sensor data to find the most suitable path. However, despite of such a large amount of data, it has been found that the poor quality of links between two different Geo-spatial data sets can hamper the use of Geo-spatial data.

In order to resolve this issue Celino et al [Celino2012] proposes that by including humans, for validating the links can significantly improve the quality of links. Some of the tasks that are difficult for the machines or computers are trivial for humans. While creating links between two datasets, the links are classified in two categories: 1) Correct and 2) Incorrect. Now to assign the links in these two categories a threshold has been set for both the categories. The links which are between the two threshold values will be treated as unqualified links. Such links will be assessed by an expert. In order to involve humans for validation, the author builds a mobile application called ‘UrbanMatch’.

‘UrbanMatch’ is a mobile based application which is used to link the sites (Dataset A - point of interest) with the images (Dataset B). In ‘UrbanMatch’ the players are asked to link the given photos with the appropriate point of interest. While designing such applications various thing needs to be taken into mind, like the incentives given to the user to motivate users to play games, to identify untrusted users in a game, how to decide if the link is correct or not and things to keep in mind while designing user interface.

Taset It, Try It

Cuel et al [Cuel2012] discuss about how to motivate users to perform tasks effectively and enthusiastically. The authors suggest that for adding incentives in your application one has to understand the motivations of the users. Every user has different motivations in their life, some of them do work to impress others, some of them love their work, some of them do work for money and some of them do work for the community. It is important to understand users to whom you are designing an application.

As a part of this research authors developed a semantic application called ‘Taste It, Try It’. Users can use this application to write reviews of the restaurants they visit. The values of the fields like cuisine type, used in this application are taken from DBpedia. In this application, authors have used auto-completion mechanism which suggests suitable values to be used. This application is not only using Linked Data but also creating annotations whenever a user submits a review about any restaurant. Incentives models that authors have used in this application are: 1) User-friendly user interface. 2) Award user on for good reviews and also includes the concept of sociability design. 3) Use auto-completion mechanism to provide suggestions to the user.

As a part of the experiment, this app was used by students. Students were awarded with points on submission of each review. It was found that students were more interested in submitting reviews in order to get as many points as possible to top in their group. The incentive factor here was awarding points to users on submission of good reviews.

Waisda

Hildebrand et al [Hildebrand2012] describe how meta-data generated by users can be used for video tagging. Currently, videos are identified as whole by using linked data annotations. Tagging video frames with the tags can greatly improve the access to the contents within videos. By doing so one does not have to search through the whole video manually.

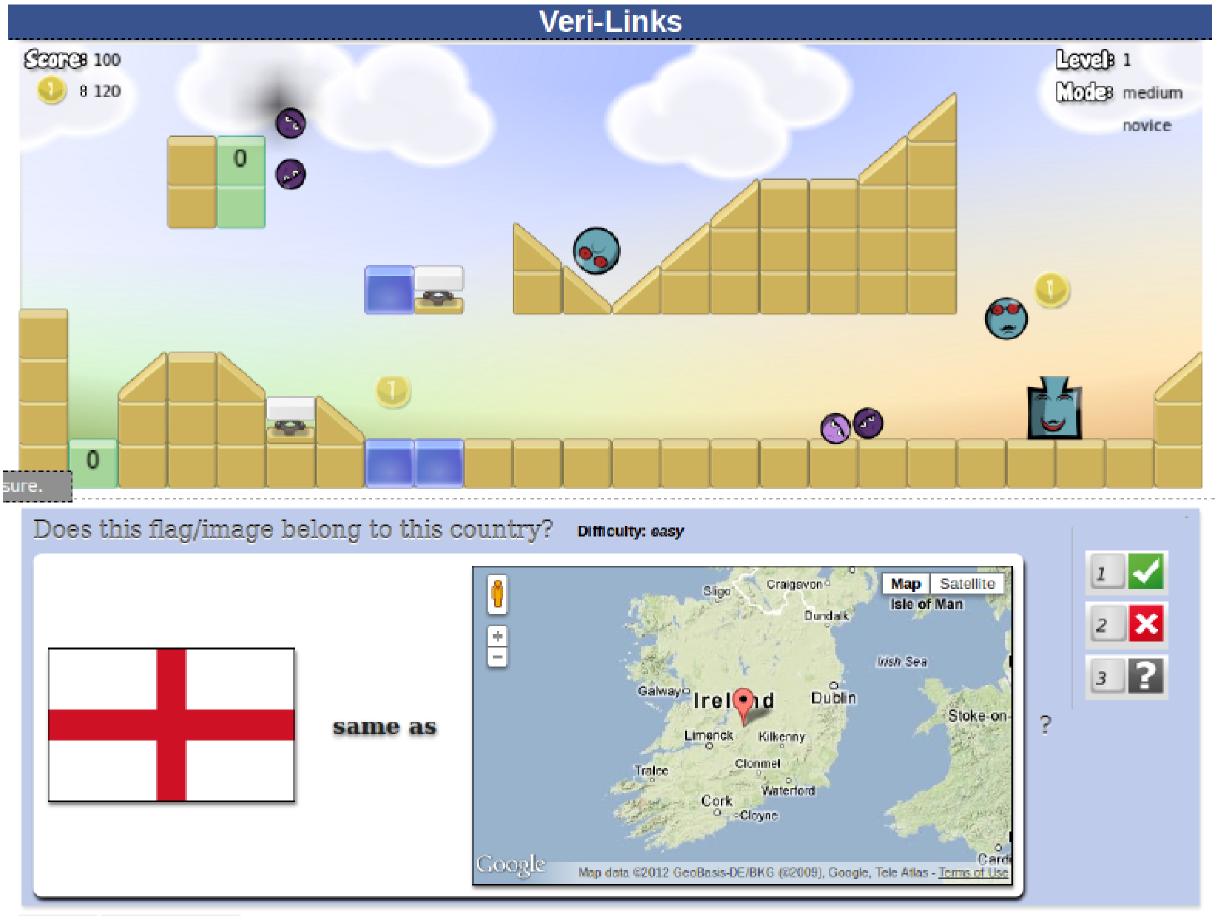
However, the tags created by the users while playing games may not be same as searched by a searcher. In order to resolve this issue, Hildebrand et al [Hildebrand2012] suggest of using 'reconciliation services'. Reconciliation services provide suggestion based on the textual search. User can select the most appropriate result from the suggestions and can link it to the video frames. There are many reconciliation services available like 'Freebase' and 'Europeana'. A video - sharing website like YouTube use automatic approaches for tagging their videos. However, it has been found that the tags created by automatic tools are not precise and cannot be used for long time.

Hildebrand et al [Hildebrand2012] develop a multiplayer game called 'Waisda', in which at a time only two users can compete with each other. If both the users select the same tag, then they will win, otherwise not. Users are provided with an option to choose the reconciliation service from a set of services. Users select one of the tags suggested by the reconciliation service. Tagging can be done based on the people in the video or the place or monument in the video. These tags can be used for navigating through the videos.

Verilinks

Lehmann et al [Lehmann2013] describe how semantic games can be used to improve the user verification stage in semi-automatic approaches. As a part of this research Lehmann et al [Lehman2013] developed a computer based game platform ‘VeriLinks’. This game is very much inspired from Ahn et al [Ahn2008] ‘Games with a Purpose’. Two games were developed on the top of this platform – pea invasion and space ships. In order to improve the verification process, Verilinks presents links graphically in the games.

Verilinks can be applied to any linksets. One just needs to specify the links to be created, SPARQL end points containing the information about the entities to be linked and a template which describe how to represent a link in the games. It provides users with a user interface, using which one can design a template for the linkset.



Does this flag/image belong to this country?

Figure 2.7: Pea invasion user interface [Lehmann2013]

These games are generally played between two players. The difficulty of the question depends on the user’s response on the previous questions. For every correct answer users are rewarded with an award. The authors conclude at the end that presenting images to the players and hiding strings in the background, revealed promising results and improved the quality of data sets.

2.3.3 Analysis

Previous section describes how semi-automatic approaches are better than automatic approaches. It also discussed available semi-automatic tools for data linking. This section presents the comparison between those tools. The main focus of the comparison is the motivating techniques that these tools used to encourage users to contribute. Despite having so many user focused applications, still linked data applications are not famous among common users. This section presents the problems associated with the semi-automatic tools. When the applications are designed to involve humans, special attention must be given to the usability of that application. Even though applications like ‘Taste it, Try it’ and ‘waisda’ have taken this thing into account, but still these applications are not in use.

Table 2.4: Comparison of semi-automatic interlinking tools

Interlinking Systems	**Motivating Techniques**	**Domain**
UrbanMatch	Games with a purpose	Urban data set
Taste It, Try It	User friendly interface, auto-completion	Restaurants
VeriLinks	Games with a purpose	Any
Waisda	NA	Tagging Video frames

In Waisda, the authors have not considered if these reconciliation services are reliable or not. There can be a case where these services do not provide any suggestions for tagging. This could result in inconsistent tagging of frames. Also, there is no provision for checking spelling of tags. Also, the users who are involved in tagging are needed to have some prior knowledge of vocabularies used or the reconciliation services. This has limited the scope of users who can use this game.

One should not forget the sole aim of building their semantic application. The main aim of semantic application is to contribute in the field of semantic web. One should not build applications for users at the cost of its sole aim. It is important to consider users' interests while building applications, but that should not affect the purpose of application, otherwise the application will be of no use.

While designing ‘UrbanMatch’, not much attention is given to the usability of the game. It is important to build an application that is easy to use and hassle free. Also, the authors failed to take into account the quality of the photos. The photos in the application are taken from two sources: Filckr with Wikimedia. Sometimes, the photos presented to the user may not be of good quality, in which case the user will be end up creating bad links.

2.4 Human Computer Interaction principles in the web of Linked data

This section presents research that has been undertaken to improve the usability of the semantic applications. People, who are not aware of semantic technologies, generally find it difficult to use them. Cuel et al [Cuel2012] and Koutsomitropoulos et al [Koutsomitropoulos2011] describe how machines can help users by giving auto-complete suggestions for an entered string. Users can select the most appropriate options for them. Also, auto-complete suggestions make user's life easy.

Heath et al [Heath2006] describe the challenges involved in developing the semantic application. One should be clear about what to show users and what not. While developing such applications one should know what data he wants from the user and how that data can be used in other applications. The same applies to the ‘Games with a purpose’ also [Siorpaes2008, Celino2012]. The user interface and storyline of the game should be good, otherwise it will make users feel bored after some time. These applications should be built in such a way, that even the users who are not familiar with the concepts of linked data should be able to use it. Heim et al [Heim2011] introduced an interactive model strategy where both humans and computers interact with each other to create and retrieve unambiguous data in semantic web. Macias et al [Macias2012] describe how end user development and model based user interface design approaches can help end users to manage semantic applications without having knowledge of semantic languages.

Also, while developing applications one should check the usability of the application. It should not be what other users don’t want. It should be simple and easy to use. It is always important to identify the needs of end users. García et al [García2006] describe how the usability of semantic applications can be improved by following User Centered Design approach. User should not find any difficulty while navigating through the application.

There is a new way of addressing this issue suggested by Partarakis et al [Partarakis2009]. According to them, one can build semantic web applications which can support user interface adaptation on semantic web. The benefit of employing this mechanism with the semantic web is that the semantic web can allow more complex rules to be applied to the user profile which will help in developing appropriate user interfaces. The next section presents the related work undertaken in the field of human computer interaction in the semantic web.

2.4.1 Related Work

User Interactions and Challenges

Heath et al [Heath2006] describe how the application developed for 3rd European Semantic web conference is perceived by the users. They developed a semantic application for the conference, which provides them with the information of other delegates, social network use on semantic web and a chat application. A survey related to the usage of the application was conducted after the web conference. It was found that the functionalities provided by the development team were not sufficient.

Heath et al [Heath2006] describe the various challenges involved while developing a semantic application. Users using semantic applications should not need to have prior knowledge of languages used in the semantic web. Semantic applications should be built in such a way that even users who don't know about semantic web or linked data should be able to use it.

Another important aspect is the user interface of an application. The user interface should be simple and should not be what users are not used to. In this paper the authors talk about if the semantic layer should be seen as separate from the existing web or not. Another challenge that authors faced is the coherency of these applications. One should have clear knowledge of what data to collect from the users and how to use that data in other applications. Above mentioned issues are needed to be addressed in order to build successful semantic web applications. The authors chose semantic web conference to evaluate a semantic application developed them. This is a good way to evaluate, since they are getting feedback from the researchers who are already working in this field.

However, if the same research was done with common people (who don't have knowledge of semantic web) the results could have been different, since at the end the ultimate goal is to bring this technology to a common user. The members involved in the survey are well versed with these technologies. They might not have faced the issues which common user would have faced.

Interactive Model Strategy

According to Heim et al [Heim2011], both manual and automatic approaches to publish and retrieving information is not sufficient. This process can be improved by involving both computers and humans. Also, by improving the interaction between humans and computer in semantic web can give way to new business opportunities and big applications.

In order to improve human computer interaction they introduced an interactive model strategy where both humans and computers interact with each other to create and retrieve unambiguous data in semantic web. In this model both human and computers inform each other at short intervals of time about what they have understood and identify any misunderstandings if any. An interaction model works same as mental models, where there has to be common alignment between the things going on in the heads of two people.

In interactive alignment, the user first sends the request in the form of syntactic-lexical to the semantic web. In order to generate the response to a request, keywords from the request are matched to generate a semantic representation. The response is then sent back to the user in syntactic-lexical form. If the response sent could not fulfill the objective of the user, further steps are executed. As a part of this model authors have developed a tool 'Relfinder' which works on the principles of interactive model. In this tool whenever a user search something, the computer gives suggestions based on the keywords search. If the user selects one of the suggestions a response is sent back to the computer where the corresponding syntactic-lexical representation can be stored unambiguously using URI's for future use. This paper describes how by improving human and computer interaction, benefits of semantic web could be reached to an average user. The interactive model approach taken by the authors has many advantages. When both computers and human interact with each other, they can lead to creation of error free information. Also, by involving many people can generate detailed information. The information generated through this approach is of high quality. This high quality and detailed information can be used in building big semantic web applications.

However, if there is any need of changing some information, it could lead to inconsistencies in the dataset. Heim et al [Heim2011] failed to take this into their consideration. Also, sometimes the machine may not be able to suggest suitable suggestions. In that case users objective may not get fulfill. Also, the authors have not considered the involvement of bad users in this process. Unauthentic users can deteriorate the quality of information.

Model based user interfaces

Macías [Macías2012] describe an innovative approach to develop semantic applications. According to author semantic web applications can be built by using both Model based user interface and End user development approach. The end user development approach allows users to change the existing semantics of ontology without even having knowledge of complex semantic languages. In order to build end user development environment, the author used some specific tools using which one can add semantic knowledge and can also render semantic knowledge in the form of web pages. PEGASUS is based on model based user interface design approach. This tool is used to build front-end pages with the basic aim of separating knowledge and presentation. PERSUES is a tool which is used by domain expert users to generate ontologies in an efficient manner. DESK tool is used to convert the end-user response or changes into corresponding changes in the domain knowledge at the back-end.

The main difference between the PEGASUS and DESK is that the former represents the semantic domain knowledge in the form of web pages, whereas the latter generates semantic content automatically whenever end users modify web pages. In order to evaluate proposed approach author asks users from different backgrounds to use developed application. None of the users had any prior knowledge of semantic web except of how to develop basic HTML pages. It has been found that most of the users found, the application is easy to use. The benefit of end user development approach is that it allows the non-expert users to use semantic applications in an easy way. This will allow more people to contribute towards the semantic web. Also, users don't have to make changes directly at the back-end. These tools allow users to make changes in HTML pages only. The changes made by the end users in HTML pages are automatically applied to domain knowledge at the back-end using these tools. This allows users with no prior domain knowledge to contribute. However, it has been found that there are some limitations with DESK. This tool can only be used with specific representations. Also, the author has not considered the mistakes made by a common user. This can result in poor domain knowledge. There should be a process to check if the changes made by an end user are correct or not. There is no such measure discussed in this paper.

User Centered Design approach

Garcia et al [García2006] suggest of following User Centered Design approach for building semantic applications. According to them major emphasis should be laid on end-users while developing any semantic application. From the onset of the design phase till the end of the project, emphasis should be laid on end-users. According to User Centered Design approach first task is to identify the target users.

They used the same approach in developing a web semantic browser called ‘Rhizomer’. There are many semantic web browsers available these days which work on the principles of semantic web. Mostly all of these browsers follow ‘Subject-Centric approach’. In a Subject-Centric approach, metadata can be fetched from various sources through fragments. The fragment is a list of all set of triples of the same subject. However, the problem with this type of approach is that the fragments sometimes become so large that it becomes difficult for users to keep track of subject. Also, sometimes it becomes difficult to find the source of anonymous sources. In order to resolve this issue, the authors have slightly modified this approach for their web browser. ‘Rhizomer’ displays all the information in the same context together. This improves user experience and also, makes it easy to keep track of anonymous sources. On the generation of fragments, it is rendered into HTML page. The rendered information can be displayed in a user interface with which end-users are comfortable. In order to enhance user experience, Rhizomer also allows end-users to edit the metadata information through the user interface.

The benefit of user centered design is that it laid most emphasis on the end users. It is always important to understand the requirements of end-users before developing any application. This will allow end-users to contribute to semantic web more efficiently. Also, the web browser developed by the authors provides users with better user interface than any other semantic web browser on the web. Semantic web browsers with good user interface will motivate users to switch from traditional browsers to semantic browsers.

However, there are some limitations with this approach. End-users are allowed to edit metadata information. Some dishonest users may try to degrade the quality of metadata information.

Authors must have come up with some steps to address this issue. Also, even after following user centered design approach, the user interface of Rhizomer is still not that good as traditional browsers are.

User interface adaptation on semantic web

Partarakis et al [Partarakis2009] proposed an architecture using which one can support user interface adaptation on semantic web. Some of the tools that are available are The Eager and UI adaptation in mobile services. These toolkits automatically select the features from the given set of features according to the user's profile. The user profile is either created before initiating interaction between user and toolkit or it is created during the interaction between both. The real life applications that are available are EDEAN and AVANTI browser. These two applications use above mentioned toolkits for generating dynamic user interface.

In the proposed architecture, the user interface is generated on the basis of three parameters: User profile, Context profile and User interaction. These parameters can be represented using OWL language according to semantic rules. For modeling a user profile authors have created separate classes for users with disability and users without any disability. Users with disability are further divided into sub-classes. The Reasoner module then on the basis of user type and context profile generates an appropriate user interface for the user. The reasoner module also consists of rule engine which run classification rules.

The benefit of employing this mechanism with the semantic web is that the semantic web can allow more complex rules to be applied to the user profile which helps in creating most appropriate user interfaces. The proposed architecture can significantly address the ongoing problems in the field of human computer interaction in the semantic web. Users with some disability can also enjoy the benefits of semantic web using this architecture. Also, other benefits of this architecture are that more users will start using semantic applications, which in return will produce more information. This system will also help in determining the preferences of users. The system can record how users interact with it and generates user interface accordingly.

But as every coin has two sides, every method has its own advantages and disadvantages. The authors have not talked about how scalable are these systems. It is important that such systems should be able to support many users at the same time. Also, the authors should have considered how precisely these systems can create user interface according to user profiles. Inaccurate profile modeling can irritate end-users.

2.4.2 Summary

This section describes how the interaction between users and computers can be improved by incorporating HCI principles in the semantic web. Then it presents various techniques to improve the interaction between users and machines.

2.5 Overall Analysis of State of Art

Over the past few years, there has been a tremendous growth in Web of Linked Data. The main aim of Linked data is to link all the available open datasets on the web in such a way that computers can read them automatically. With the increase in the number of data resources, linking such resources has now become a challenge. This state of the art review first describes the ongoing work in the field of linking data and then how quality links can be created between heterogeneous data resources on the web. It has been found that the links generated by automatic tools are of low quality and of low precision. This review indicates that quality links can be generated by motivating users to contribute. However, linking manually, such big data resources is a time consuming task.

This review discusses the various approaches to motivate users to contribute. 'Games with a purpose' approach found to be a good way to motivate users. Since many users love playing online multiplayer games, one can build semantic web games which can engage users for long hours. Users playing these games generally do not know what is happening behind the game.

However, while designing such games one should keep in mind the HCI principles. This review discusses the various HCI techniques which can make users to contribute to linked data. It has been found that in order to build a successful application one should take care of the design and other characteristics of HCI.

Chapter 3

Design

3.1 Requirements

Earlier, the data set of Dacura contains data which was extracted from online historical archives. The next step of Data enrichment process is to link it with data sets available on the web. In this dissertation research, the goal is to link the Dacura data set entities with the Dbpedia entities. The following are the requirements derived over the period of this research. These requirements are derived based on the task need to complete:

3.1.1 Functional Requirements

1. Find the similar entities on DBpedia which are similar to the entities of Dacura.
2. Fetch the required information from DBpedia for an entity and filter the irrelevant results.
3. Display the relevant results in two different interfaces: Map and Table interface. These interfaces are selected based on the *'Fundamental User Interface Designs'* described by Ma et al [Ma2000].
4. Update the local data set with DBpedia links.

3.1.2 Non-Functional Requirements

Performance – The system should not take a long time to query Dbpedia data set. Since, one of the objectives of this dissertation is to evaluate the user interface of semantic web applications, if the system takes long time to fetch records that would affect the evaluation of the results.

Programming Language – JavaScript is identified as the main programming language, since it's easy to learn and best-suited for client side programming. JQuery library is required for creating dynamic UI's at the client side. JQuery is best suited for creating animations on client side.

3.2 Functional Architecture

Figure 3.1 shows the functional architecture of the system. This section will give a brief introduction about how the system works. The detailed explanation will be provided in the next section.

1. Get the entity information from the Dacura data set.

2. Search the similar entity in Dbpedia data set.

   a) In case no relevant matches are found, use Google places auto-complete API and search again.

   b) If the number of records returned is not zero and none of the records found similar to the desired one, use Google search to get more results.

3. Filter irrelevant or duplicate records.

4. Display Dbpedia entities in two interfaces: Map and Table.

   a) MapQuest API to display records on the Map.

5. Update the Dacura dataset with Dbpedia link.

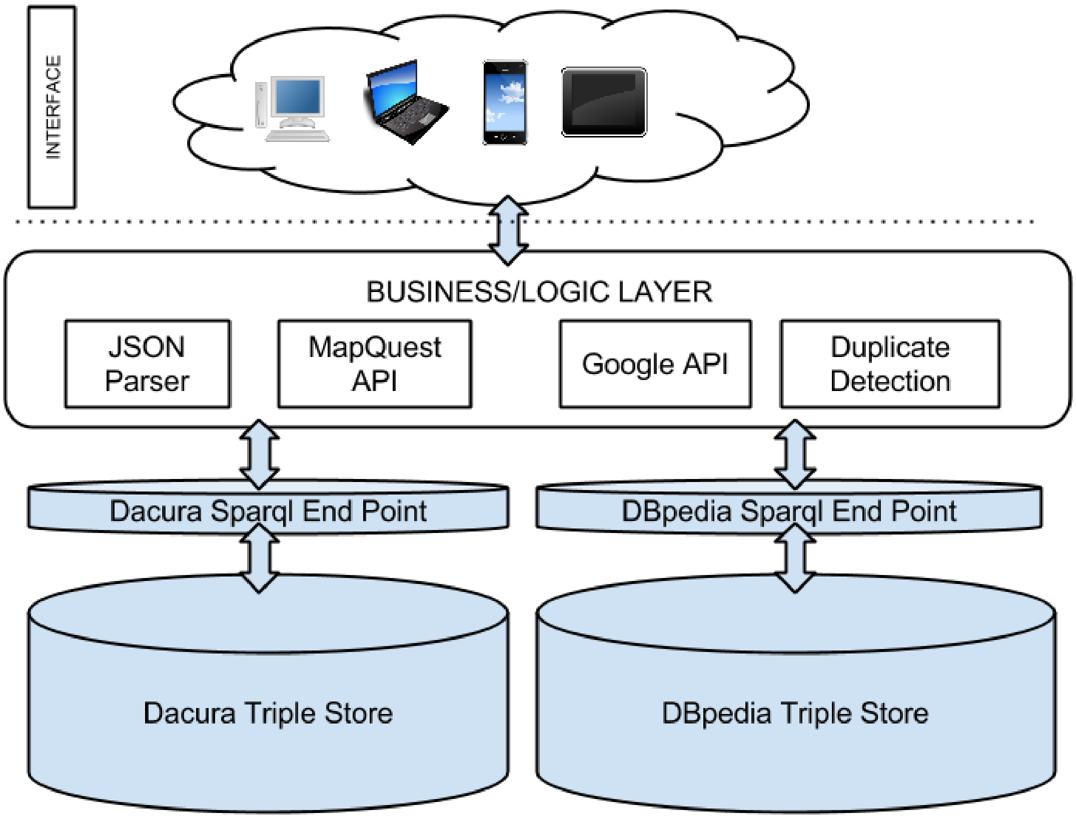


Figure 3.1: Functional Architecture Diagram

3.3 Design Challenges

The research question *'How to build Linked Data applications and tools to interlink different data sets efficiently, easily, effectively and more pleasingly as to provide users with the best experience?'* given in chapter 1 is divided into following sub-questions:

1. *How to build Linked data applications that enhance the use of existing data available on the web through interlinking?*
   One of the biggest challenges in building Linked data applications is to link heterogeneous data sets. There are many problems associated with linking process, which has already been discussed in section 2.1. The existing Dacura data set is not complete enough to be able to link it with the data on the web easily. Also, it suffers from various problems like misspelled records and incorrect information. These problems make it difficult to link data efficiently and effectively with other data sets. This question will be addressed in the next section.

2. *How to design user interface for Linked data applications?*
   Even after so many years, not many people are using linked data applications. Linking data sets is a time consuming and cumbersome task, and can become worse if the user interface of the application is difficult to use. It's not just about making interlinking process easy; it's also about how to display information on the UI. Too much information or too little information can work against the usability of the application. The next section will discuss how this issue has been addressed.

3. *How to evaluate Linked data applications in an efficient way?*
   One of the main challenges in evaluating Linked data applications is to find participants with desired background knowledge. It's difficult to evaluate such applications with common users. Also, since there are not many linked data applications, the best way to evaluate is to compare it with non-linked data applications. But there are many metrics that need to be evaluated while comparing it with other applications. Non linked data applications may not perform the same tasks.

3.4 Addressing the Design Challenges: Design Decisions

The last section details the challenges that are faced while designing Linked data applications. This section discusses the decisions made to address those challenges. Table 3.1 shows how requirements and influences from state of the art map into the design decisions.

Table 3.1: Motivations behind design decisions

Design Decision	Requirement Influence	SOA influence
Entity Recognition	Requirement 1	· Determine set of methods to determine entity to be linked before fusion process, Nikolov et al [2008], Section 2.2.1 · Support for the users, auto-completion, Cuel et al [Cuel2012] and Koutsomitropoulos et al [Koutsomitropoulos2011], Section 2.4
Information Extraction	Requirement 2, Requirement 3	Lavenstein matching algorithm, Rowe et al. [Rowe2009], Section 2.2.3
Data Visualization	Requirement 4, Requirement 5	· Semi-automatic Approahces, Section 2.3 · User Interfaces to display records, Section 2.4.1

The design of the system is divided into various modules which are discussed below:

1. **Data Interlinking**

   Entity Recognition -
   Identifying the attributes of the entity that are to be used for the linking process is the first step in the interlinking process. This module correctly identifies the attributes of the entity. Since the information in the Dacura data set is limited, only the location attribute of each entity could be used to link it with Dbpedia. While linking entities with Dbpedia, it was found that for some of the entities the value of the location attribute was not correct. Some of the problems were misspelled locations and incorrect location. In order to address this issue, Google places auto-complete API is used, so that the users can select the correct location from the suggested list of options from Google. This technique has been inspired from [Cuel2012, Koutsomitropoulos2011].

   Information Extraction -
   This module is responsible for extracting relevant information from the Dbpedia. In order

to join two data sets, one must find the equivalent entities in both the data sets. This is called schema matching problem [Monge1996]. Following steps are taken to search similar entities on DBpedia:

a) The entity value must be same as desired location value – This is done to avoid extraction of irrelevant records.

b) The entity must be of type place - Since the entities are searched using the location attribute, the system should not fetch entities of others than of type place.

c) The entity must belong to a country – This is done to skip entities with no country attribute. While doing linking process, it was found that for some places, there is no country attribute on Dbpedia. Such entities could result in extraction of irrelevant entities.

d) The entity should be part of the United States of America – Since there can be many places in the world with the same name, only places within the US are considered.

Duplicate Detection -
This module is also responsible for detecting duplicate and irrelevant records in a list of records. This is done to ensure that only matched and unique results are presented to the user. String matching algorithm is used to filter out the duplicate records.

2. **Data Visualization** -
The data visualization module displays the data in a simple and elegant way. It addresses the second challenge: design UI for linked data applications. According to Ma et al [Ma2000] traditional interfaces are not enough for exploring complex scientific data. A change is needed in order to efficiently display such complex data; it can be a spreadsheet (Excel) like format, graph-based or other some other novel way to display data.

   In order to tackle this challenge, data is presented using two user interfaces. Since, it is difficult to find what interface user's might like; two different user interfaces are used: Table and Map.

   Table interface is chosen because it presents the data in the similar way as relational databases or Excel (spreadsheet), i.e. in the form of rows and columns. Technical users find it easy to visualize data. Map interface is chosen because some users who are familiar with geographical locations generally find it easy to visualize places on the map as compared to other interfaces. Two interfaces are chosen to find out how users would respond to these interfaces.

The data table library is used for presenting data in table interface. This was chosen because it provides better UI than a normal HTML table interface. Also, it provides search functionality, which allows users to search through the records in the table.

Google Map API is used for displaying Map interface. Earlier, the latitude and longitude locations were fetched from Dbpedia along with other required information. But later it was found that for some records the latitude and longitude values were not correct. To display places correctly on the map, Map Quest geocoding API is used. MapQuest API returns precise latitude and longitude values for the requested place. The figure 3.2 shows the iterative model used for improving the interlinking process. Based on the user feedback, the improvements are made in the used interfaces.

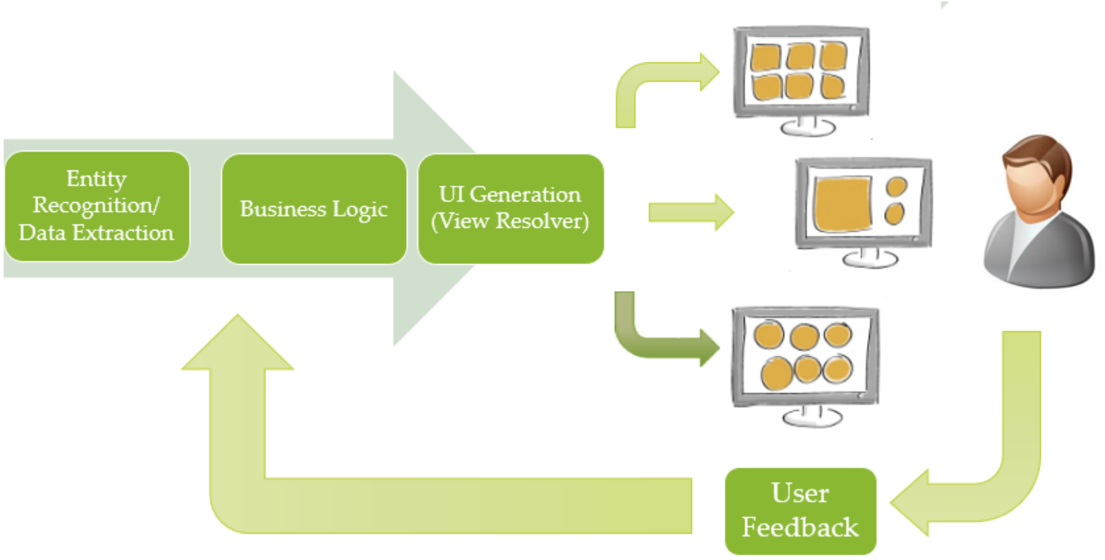

Entity Recognition/ Data Extraction
Business Logic
UI Generation (View Resolver)
User Feedback

Figure 3.2: Iterative model to improve linking process

The last challenge, ‘how to evaluate Linked data applications’ is discussed in detail in the evaluation section.

3.5 Summary

In this chapter, first both functional and non-functional requirements are discussed. Then a brief introduction to system architecture is presented. Then the challenges related to linked data application are discussed. In the last, decisions designed to tackle all the challenges are discussed.

Chapter 4

Implementation

This chapter first discuss about the relevant libraries and frameworks used for implementing data interlinking technique. In the next section, technical architecture of the system is discussed. The last section presents how interlinking task is achieved. The system is developed over the existing architecture of Dacura application.

4.1 Programming languages and libraries

JavaScript is the main programming language used to implement client side functionality. PHP is the main programming language used to implement server side functionality. However, not much programming is done in PHP, since only minor changes were required on the server side. The SPARQL query language used for querying structured data sets.

4.1.1 Libraries

1. JQuery – It's a JavaScript library. It makes client-side programming more comprehensible and easy. In this project, JQuery is mainly used for the UI components.

2. Places Auto-complete – This is a Google Map API library. This library is used to provide suggestions to the user about the places in the United States.

4.1.2 Plug-ins

Data Table – Data Table is a JQuery plugin. It displays results in the tabular form. Apart from displaying results it provides other functionalities as well, such as pagination, a search option and an expressive user interface. This Plug-in is used for displaying DBpedia records in a tabular form.

4.1.3 API's

1. Google Map JavaScript API – This API is used for displaying DBpedia places on the map. Each marker on the map corresponds to a different place.

2. Map Quest Geocoding API – This API is used for getting the latitude and longitude of DBpedia places to show them on a Google map.

4.2 Technical Architecture

This section presents the architecture of the designed system. The system is made up of four parts:

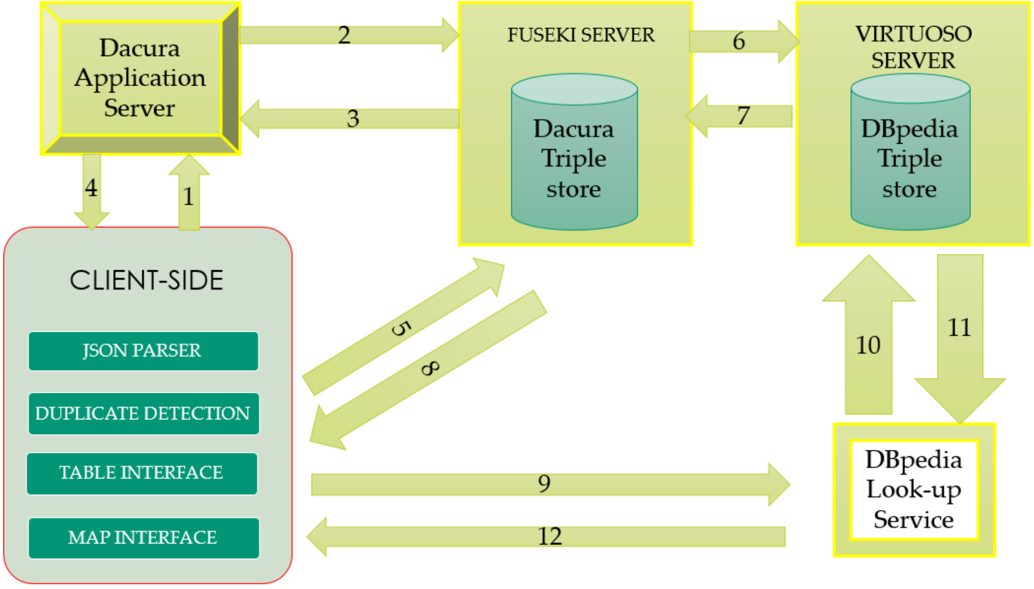


Figure 4.1: Technical Architecture Diagram

1. **Application Server** - Application server runs on the Apache server. It is responsible for populating data on the first page. At the start of the Dacura application, an Ajax request is sent from client to application server; requesting all the records from Fuseki server in JSON format. On receiving client request, Application server writes a SPARQL query and sends it in the request message to Fuseki. On receiving response from Fuseki, it sends it back to the client. The application server is also responsible for the authentication of a user.

2. **Fuseki Server** - Fuseki is a server for SPARQL query language. It runs SPARQL protocol over HTTP to answer all the SPARQL update and SPARQL query requests. The response returns by Fuseki is in graph form which is then parsed into JSON, CSV or text format. The SELECT request is forwarded to query endpoint and UPDATE request is forwarded to update endpoint.

   Query Endpoint –
   s-query --service http://dacura.cs.tcd.ie:3030/politicalviolence/query

   Update Endpoint –
   s-query --service http://dacura.cs.tcd.ie:3030/politicalviolence/update

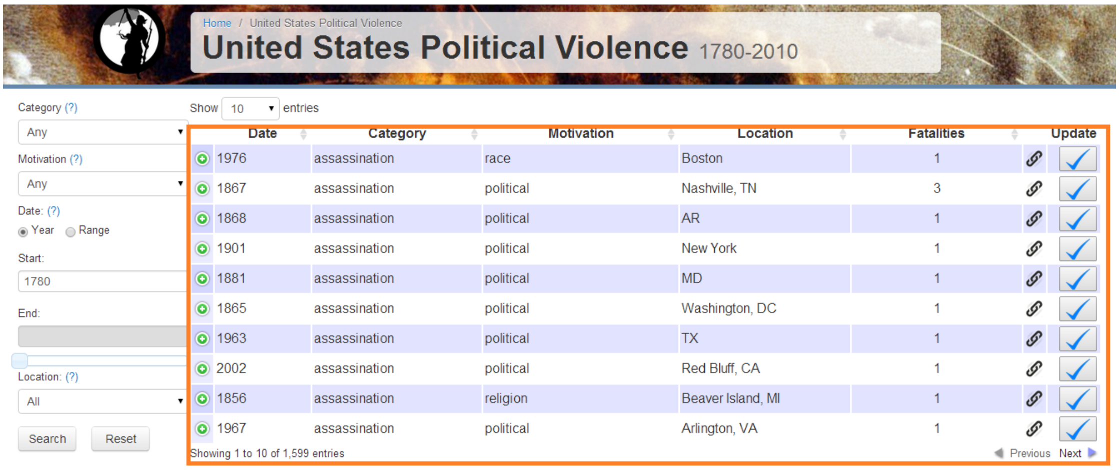
   Fuseki process the records based on the type of request. If the request is to query the records from the data set, it sends response in the graph format. All the updates are stored in the memory of the server.

   Dacura Triple store - Dacura data set is present in the Turtle format. It's a collection of political and fatal incidents from the eighteenth century till now. Such information is mainly fetched from online historical articles. This project aims at linking the entities of Dacura data set with DBpedia.

3. **Virtuoso Server** - Virtuoso is a universal database server. DBpedia use Virtuoso to manage and update its data set. The virtuoso is designed by Open Link software.

   DBpedia Triple store - Dbpedia saves its data in the RDF format. Wikipedia update this data set many times in a year. Dbpedia is a core of Linked Open Data (LOD).

4. **Client Side** - Most of the client side programming is done using JQuery and JavaScript. Client is responsible for displaying records from Dbpedia or Dacura data set to the end-user. On the load of the first page, the client sends an Ajax request to the application server. On receiving the response, the response is filled in the Data table as shown in figure 4.2.



Home / United States Political Violence

United States Political Violence 1780-2010

Category (?) Any

Motivation (?) Any

Date: (?) ● Year ○ Range

Start: 1780

End:

Location: (?) All

Search Reset

Show 10 entries

Date	Category	Motivation	Location	Fatalities	Update
1976	assassination	race	Boston	1	✓
1867	assassination	political	Nashville, TN	3	✓
1868	assassination	political	AR	1	✓
1901	assassination	political	New York	1	✓
1881	assassination	political	MD	1	✓
1865	assassination	political	Washington, DC	1	✓
1963	assassination	political	TX	1	✓
2002	assassination	political	Red Bluff, CA	1	✓
1856	assassination	religion	Beaver Island, MI	1	✓
1967	assassination	political	Arlington, VA	1	✓

Showing 1 to 10 of 1,599 entries

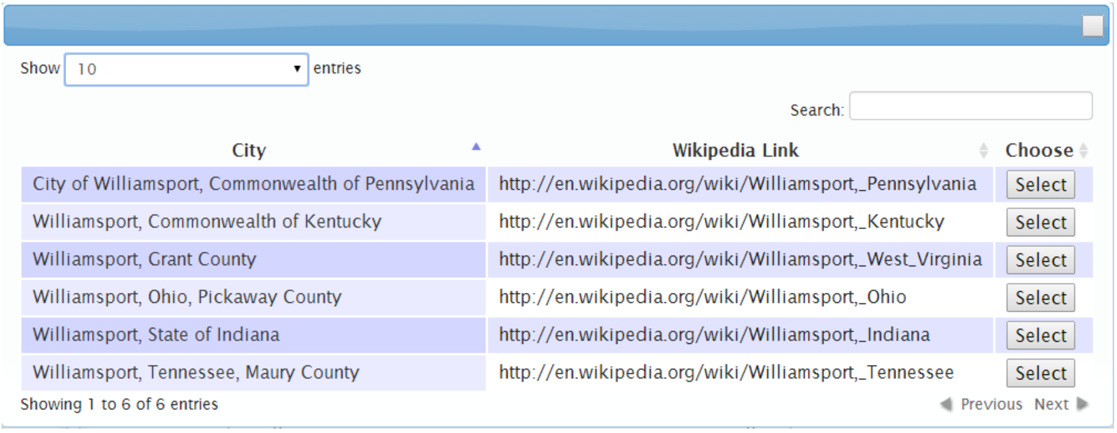
Previous Next

Figure 4.2: Dacura User Interface

JSON Parser - This module converts Fuseki response in JSON format. Response from Fuseki comes in the form of a graph. Firstly, the response is converted into JSON format and then we go through each and every object to get its property values.

Duplicate Detection - This module finds duplicate records in the JSON array object (returned by JSON parser). This module is also responsible for filtering the objects whose location attribute value is different from the desired value. This is done using string matching algorithm. For each record location value is checked, if the record's location attribute doesn't match or contain the desired string, those records will be skipped.

Table Interface - Table Interface is used to display the Dbpedia records in the tabular form. Data table plug-in for JQuery is used which provides additional other features like search table, pagination. Figure 4.3 shows the table interface with Dbpedia results.



Show 10 entries

Search:

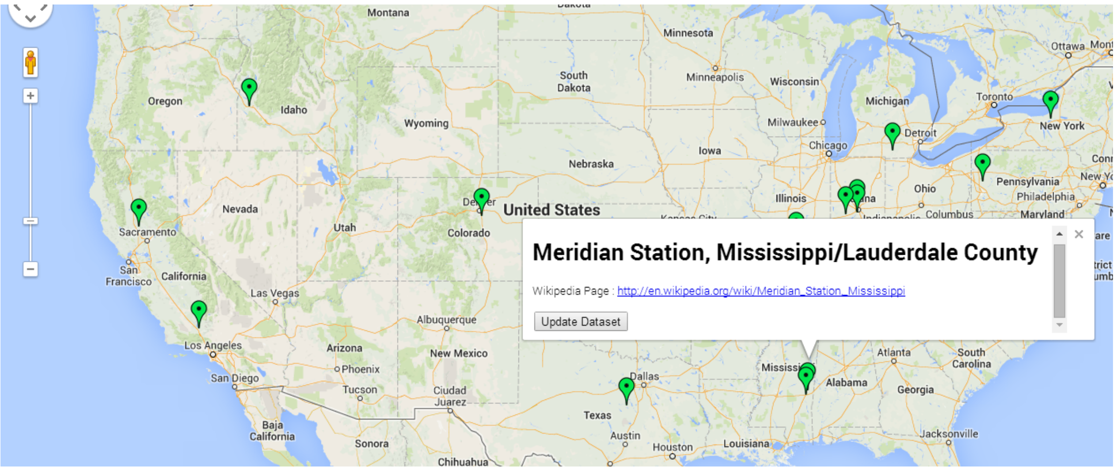
City	Wikipedia Link	Choose
City of Williamsport, Commonwealth of Pennsylvania	http://en.wikipedia.org/wiki/Williamsport,_Pennsylvania	Select
Williamsport, Commonwealth of Kentucky	http://en.wikipedia.org/wiki/Williamsport,_Kentucky	Select
Williamsport, Grant County	http://en.wikipedia.org/wiki/Williamsport,_West_Virginia	Select
Williamsport, Ohio, Pickaway County	http://en.wikipedia.org/wiki/Williamsport,_Ohio	Select
Williamsport, State of Indiana	http://en.wikipedia.org/wiki/Williamsport,_Indiana	Select
Williamsport, Tennessee, Maury County	http://en.wikipedia.org/wiki/Williamsport,_Tennessee	Select

Showing 1 to 6 of 6 entries

Previous Next

Figure 4.3: Table Interface

Map Interface - Google Map Api is used to display places on Google Maps. Each marker on the map corresponds to different locations in the United States of America. User can choose to update record by clicking on any of the markers and then update button. 'Green' colored markers signify that the location of the place is correct. 'Red' colored markers signify that the location of the place may not be correct.



Meridian Station, Mississippi/Lauderdale County

Wikipedia Page : http://en.wikipedia.org/wiki/Meridian_Station_Mississippi

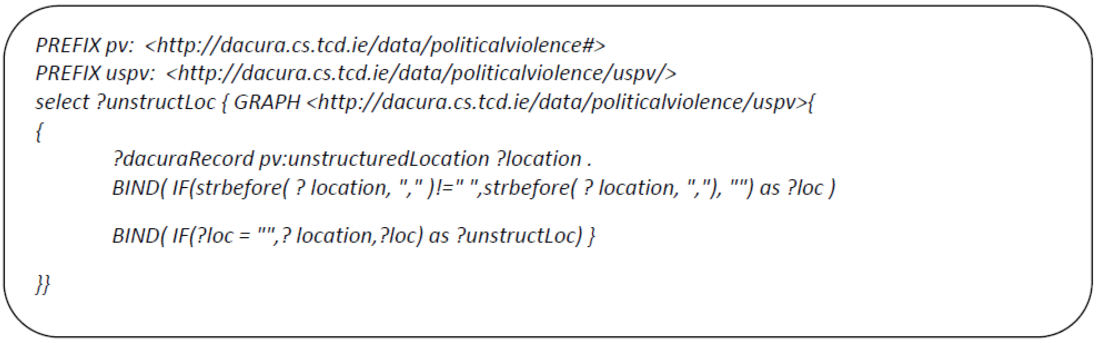
Figure 4.4: Map Interface

4.3 Interlinking Task

Interlinking task is divided into two parts: *Selection* and *Visualization phase*.

4.3.1 Selection Phase

Entity selection is done using string matching technique. When the user selects a record to update, a SPARQL query is run on the Dacura data set which gives the unstructured location value of the corresponding record. SPARQL query to fetch unstructured location is shown in figure 4.5.



```
PREFIX pv:  <http://dacura.cs.tcd.ie/data/politicalviolence#>
PREFIX uspv:  <http://dacura.cs.tcd.ie/data/politicalviolence/uspv/>
select ?unstructLoc { GRAPH <http://dacura.cs.tcd.ie/data/politicalviolence/uspv>{
{
        ?dacuraRecord pv:unstructuredLocation ?location .
        BIND( IF(strbefore( ? location, "," )!=" ",strbefore( ? location, ","), "") as ?loc )

        BIND( IF(?loc = "",? location,?loc) as ?unstructLoc) }

}}
```

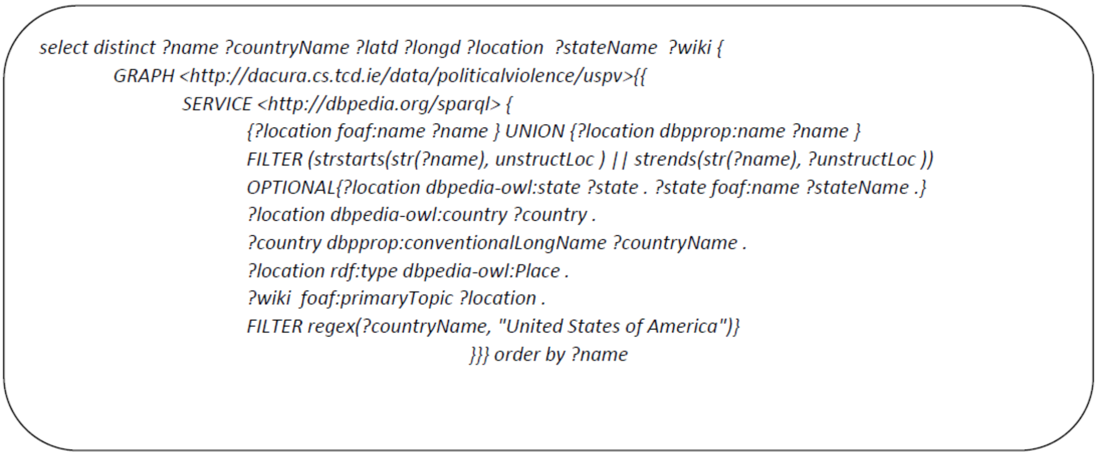
Figure 4.5: Entity selection - SPARQL query

Before First Experiment

The unstructured location value is then used for finding similar entities on Dbpedia. This process is called entity matching. Following Dbpedia properties have been used to find relevant entities.

1. dbpprop:name – Object of this property represents the name of the entity.
2. dbpedia-owl:Place – Object of this property is of type place.
3. dbpedia-owl:country – Object of this property is of type country.
4. dbpprop:conventionalLongName – Object of this property represents the conventional long name of an entity.
5. dbpedia-owl:state – Object of this property is of type state.

Since for some records the unstructured location value in Dacura data set is incorrect or misspelled, the Dbpedia search SPARQL query returns zero records from Dbpedia. In order to address this issue, Google Places auto-complete API has been used. This API gives a list of suggestions for an entered location string. User can select a record from the list of suggestions and can search again. SPARQL query to get a list of records from Dbpedia which are similar to unstructured location attribute of Dacura is shown in figure 4.6.



```
select distinct ?name ?countryName ?latd ?longd ?location  ?stateName  ?wiki {
        GRAPH <http://dacura.cs.tcd.ie/data/politicalviolence/uspv>{{
                SERVICE <http://dbpedia.org/sparql> {
                        {?location foaf:name ?name } UNION {?location dbpprop:name ?name }
                        FILTER (strstarts(str(?name), unstructLoc ) || strends(str(?name), ?unstructLoc ))
                        OPTIONAL{?location dbpedia-owl:state ?state . ?state foaf:name ?stateName .}
                        ?location dbpedia-owl:country ?country .
                        ?country dbpprop:conventionalLongName ?countryName .
                        ?location rdf:type dbpedia-owl:Place .
                        ?wiki  foaf:primaryTopic ?location .
                        FILTER regex(?countryName, "United States of America")}
                                                        }}} order by ?name
```

Figure 4.6: DBpedia Entity Extraction - SPARQL query

After the first experiment, most of the users complained about the time the system took to fetch records from Dbpedia. In order to improve the performance of the system, Dbpedia lookup service is used. Dbpedia lookup service provides two types of search methods: Text-based search and auto-complete search.

In this system, both types of search mechanism have been used. First the Text-based search is used, if it could not fetch any records then auto-complete search mechanism is used. In order to get results from Dbpedia, an Ajax call is sent to Dbpedia lookup service with the location value to be searched. Dbpedia lookup service returns results in JSON format which is then parsed into the desired format.

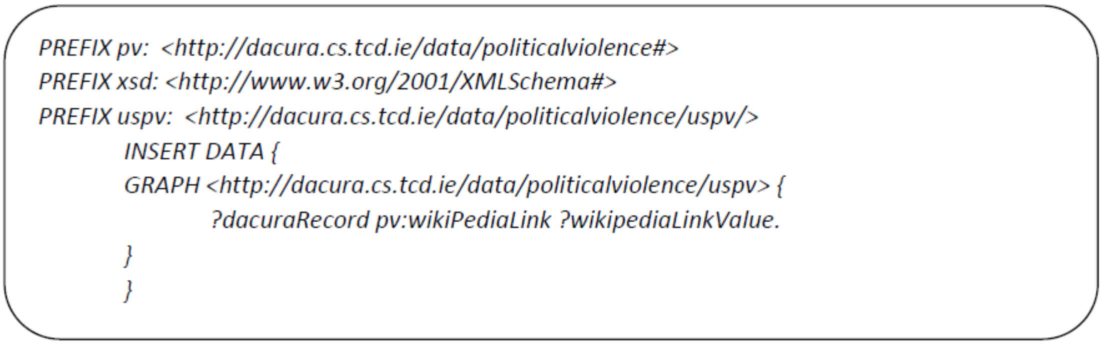
After this step, both duplicate and irrelevant entities are detected. These entities are skipped from the list of Dbpedia records. Duplicate entities are detected using string matching algorithm. This algorithm checks if Dbpedia entity’s location attribute contains Dacura’s unstructured location string. If it doesn’t, the record will be skipped from the list.

4.3.2 Visualization Phase

In this phase list of records are shown on the UI. Two interfaces have been used to display the results, Map and Table interface. Both the interfaces have inbuilt Google Places auto-complete API, using which user can search the records for different location. MapQuest geocoding API has been used for displaying records on the Map interface. This API returns latitude and longitude of each location in the list. These points are then used to display markers on Google Maps.

hen the user selects a record to link with Dacura data set, a SPARQL update query is run at the back-end, which adds wikipediaLink to the corresponding record. A new property ‘<http://dacura.cs.tcd.ie/data/politicalviolence#wikiPediaLink>' is created, which represents Wikipedia link of Dacura record.

SPARQL update query to update the corresponding record in the Dacura data set is shown in figure 4.7.



```
PREFIX pv:  <http://dacura.cs.tcd.ie/data/politicalviolence#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX uspv:  <http://dacura.cs.tcd.ie/data/politicalviolence/uspv/>
        INSERT DATA {
        GRAPH <http://dacura.cs.tcd.ie/data/politicalviolence/uspv> {
                ?dacuraRecord pv:wikiPediaLink ?wikipediaLinkValue.
        }
        }
```

Figure 4.7: Dacura Triple store update - SPARQL query

4.4 Summary

In this chapter, first the programming languages and libraries used are discussed. Then the technical architecture of the system is presented, with detailed explanation of each component. In the last the chapter describes Data Interlinking Task, with detailed explanation of each phase.

Chapter 5

Evaluation

This chapter presents the evaluation approach and the results of the developed system. A task-based user evaluation is conducted, with the aim of comparing the efficiency, effectiveness and usability of the system with both table and map interface. To measure these characteristics, the time spent by each user on the two different user interface was recorded. After the experiments, each participant was asked to fill a questionnaire.

The evaluation was conducted through two experiments. In the first experiment, the aim was to establish a baseline of usability by finding what issues users would face while using the system, how much time it would take them to complete the data interlinking task and which user interface they found most difficult to use. After completing the first experiment, changes were made to the existing system based on the user comments and the feedback. In the second experiment, the performance of two interfaces was compared, participants were asked to use only one of the user interfaces for the linking task. This chapter then compares the results of both the experiments.

As a part of the evaluation process, the following research objectives are evaluated:

RO1 - Determine how accurate the system is in terms of finding the relevant records.
To measure this objective, the number of records which users were able to link is counted. This information is inferred from the logs created at the back-end. Each time the participant selects a record to link, the information about the selected record is logged in the log file. The accuracy of score is calculated by dividing the number of records user actually linked by the total number of records user tried to link.

RO2 – Determine which user interface required more time to complete the interlinking task.
To measure this objective, time spent on each task is recorded. The system keeps track of the time spent on each record to create links. The metrics used to evaluate this research objective is the standard deviation, which is the amount of variation of time taken from the average time.

RO3 – Determine if the links created by participants are accurate.
To measure this objective, a manual approach is followed. From the collected data, all the linked records are checked manually to find if the records linked by participants are accurate. To check the accuracy of linked records, Wikipedia link of each updated record is checked.

5.1 First Experiment

The first experiment was conducted to get the initial feedback on the UI and performance of the system. Apart from evaluating research objectives, user satisfaction is also calculated based on the SUS (System Usability Scale) standards. The experiment does not require participants to have prior knowledge of any semantic web technologies. For the first experiment, participants from The Knowledge and Data Engineering Group (KDEG) of Computer Science department of Trinity College were recruited. All the participants performed experiments remotely on their machines. In the section first the methods of the experiment are discussed. Then the results obtained from evaluation are discussed. In the end, summary of the first experiment is presented.

5.1.1 Method

In this section, first the procedure of the experiment is presented. Then the data collected as a part of the experiment is discussed. Then the section describes analysis approach taken to evaluate the system.

Procedure

In the first experiment, the participants were recruited through email. In total six participants took part in the evaluation. Their task was to create links between the records in the Dacura application and places in Dbpedia. The whole process took 3 days to complete. Before the evaluation, all the participants were provided with an instruction document to help them understand the task. The participants were asked to create 50 links, 25 with the Map interface and 25 with the table interface. They were given the liberty to choose 50 records randomly from the total of 1600 records.

There were no gold standards to compare as the participants randomly chose to update the records. They were asked to leave the records for which the system couldn't find relevant records.

Some of the participants chose to complete the task in one go and some of the participants completed their task in parts at different time. However, not everyone completed the task. Only two out of six users completed the whole task.

Data

As part of this experiment, the participant's actions with the system were recorded in a log file on the different server. The system also recorded the time that each user took to link a record. At the end of the experiment each participant was asked to fill a questionnaire and provide feedback on the system. The entire set of data recorded during the experiment is provided in the DVD.

Analysis Approach

The performance of the system is calculated using the qualitative analysis of the time taken by the users to link the records. Accuracy of the system is evaluated from the data available in the log files. Accuracy of the links created is evaluated using manual approach. System Usability Scale (SUS) standard is used to compute the usability of the system.

5.1.2 Evaluation Results

This section is divided into three sub-sections. The first sub-section talks about the efficiency of the system. The second sub - section describes the effectiveness of the system. The third sub-section talks about the usability of the developed system. The entire data set of this experiment is provided in the accompanying DVD.

Efficiency

The efficiency of the system is measured on the basis of the time taken by the participants to link a record. This is the common method of evaluating the efficiency of any system. In total the participants tried to link 157 records; of which they could only link 116 records. For the remaining records, participants could not find any matched record. Here, the efficiency of the system is evaluated based on the number of records which were successfully linked.

Table 5.1 shows the number of records linked, total time taken to link those records using Map and Table interface, standard deviation and the mean time taken. The small value of standard deviation indicates that the time taken to link each record is close to mean value; large value of standard deviation indicates that there is a large amount of variation in the time taken to link each record.

$$\text{Standard deviation } \sigma = \sqrt{1/N \sum_{i=1}^{N} (x_i - \mu)^2}$$

x_i is the time taken for ith record to link. μ represent the mean time taken.

N is the total number of records linked. From the table 5.1, it can be inferred that for the table interface, there was not much variation in terms of time taken to link each record, whereas for the map interface there was a large amount of variation in terms of time taken to link each record.

Table 5.1: Time taken by users to link records

Interface Type	**Number of Records**	**Total time taken (in seconds)**	**Standard Deviation (variation from mean) σ**	**Mean (in seconds)**
Map	57	1071	29.775	18.789
Table	59	468	6.044	7.93

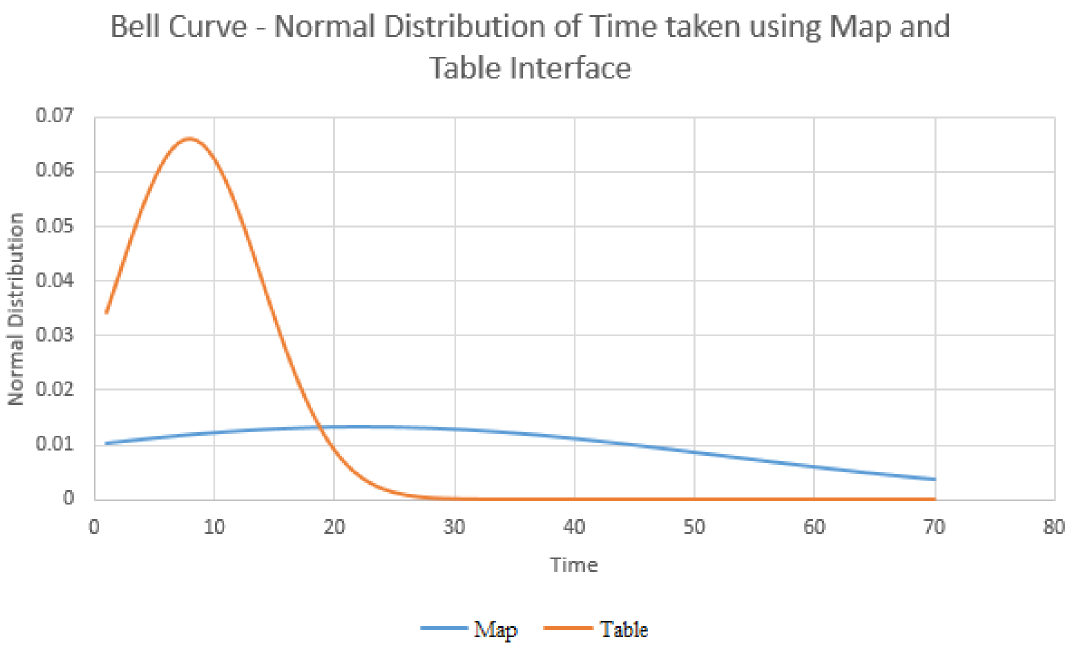


Figure 5.1: Normal Distribution of Time taken using Map and Table Interface

It is evident from the given data that the map interface took more time to link a record. Also, five out of six users said that Map interface took more time to complete the linking task, whereas only one user said that the Table interface took more time. The users also complained about the time the system took to find records from the Dbpedia. One of the comments from the users was *'Everything quite slow + a lot of switching back + forth involved.'*

Effectiveness

In this experiment, the effectiveness of the system is measured on the basis of number of links successfully linked by the participants. Here, the effectiveness of the system is measured using three metrics: Relevant records, Percentage of links generated and accuracy of the links.

Relevant Records - When asked users *'In your opinion, did the tool help you find relevant records?'* Three out of six users agreed with it. Two users were neutral and one of them disagreed with it. However, the effectiveness of the system also depends on the number of accurate results returned by the system. According to the user's response, three users felt that the tool helped them to find relevant records. However, two others were neutral and only one user disagreed with it.

Percentage of Links generated - It is found that 73.7% of the records were successfully linked using Table Interface and 74% of the records were successfully linked using Map interface. The overall effectiveness of the system is 74.1%. From the table 5.2, it can be inferred that the use of different interfaces had no effect on the effectiveness of the system.

Table 5.2: Effectiveness of the system

Interface Type	**Number of records linked**	**Number of records could not be linked**	**Effectiveness (in terms of percentage)**
Map Interface	57	20	74.02%
Table Interface	59	21	73.75%

Figure 5-2 shows the percentage of records linked by the participants. Some participants found table interface more useful for finding relevant records and some participants found the map interface more useful.

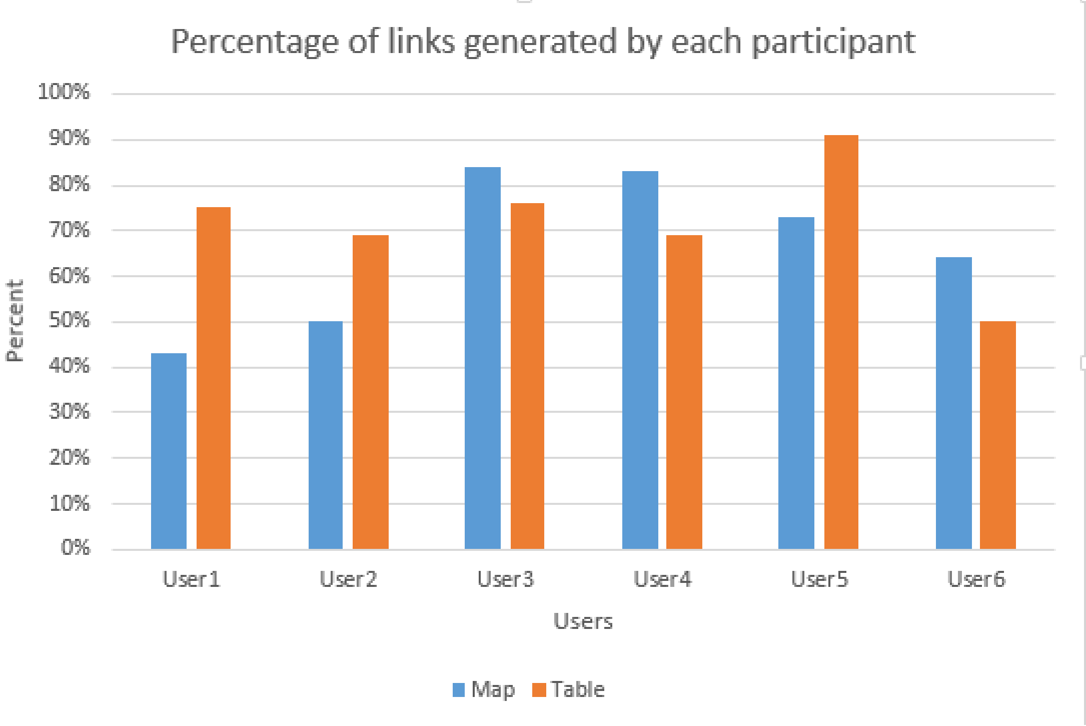


Figure 5.2: Percentage of links generated by each participant

When asked users *'What percentage, of the total number of records for the time period you* *examined, do you think you linked during the experiment?'*, three out of six users said that linked 80% of the total records, two others said 70% of the links and one user said 50% of the links.

Accuracy of links - The accuracy of the links generated by the users is measured manually. Each time the user updates the record, the information related to the link is saved in the log file. For every link created, the log file contains the URL of the Wikipedia page. To check the accuracy of the link, the Wikipedia link of the record is checked. If the Wikipedia page refers to the same location (Dacura's unstructured location), the link is considered as accurate. After going through all the information in the log file, it was found that 95% of the records were linked correctly. Also, some users commented that due to inaccurate records in the local data set, the linking system could not find relevant records from Dbpedia.

Usability

Usability of the system was calculated using the System Usability Scale (SUS) standard. SUS score is calculated on the scale of 0-100. On average the SUS score generally falls in the range of 60 and 70 [Lewis2009]. On the basis of the user's response, SUS score 49.20 was calculated. Most of the users complained that the system was not easy to use and functions in the system were not well integrated. Users also complained that the manual search option should be there even if the system found some results. One of the participants said that *'There are way too many wrong results and too many pop-up windows that need to be manually closed. That, plus the time it takes for the system to come up with results makes the task time consuming. When no results are found, the user can manually search for them, but if a wrong result is found by the system and it's obviously not the right one, the user cannot manually look for something else (or I couldn't find a way to).'* Another user commented *'Sometimes the system wasn't able to find the right place but returned some results, resulting in the inability of searching for new terms. The user should be able to use textual search even when the system return some results.'* Figure 5.3 shows the SUS diagram with interpretation of each score. From the figure 5.3, it can be inferred that the usability of the developed system was below marginal.

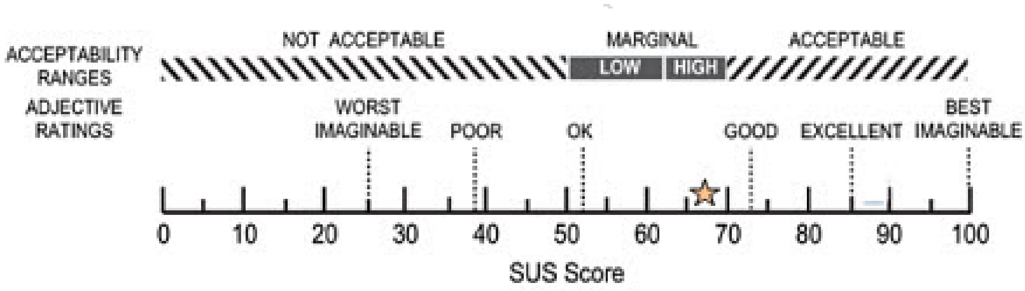


Figure 5.3: SUS diagram [uxmatters2013]

To find which user interface users found easy to use, the users were asked: *'Which user interface you found most difficult to use?'* Four users said that they found the Map interface difficult and two users found the Table interface difficult. One of the comments from the user was *'The map would be better with touch functionality, otherwise the table is much quicker.'*

5.1.3 Summary of the experiment

The section summarizes the findings of the first experiment. In the first experiment, the results revealed that the users took more time to link records using map interface than table interface. Table interface was more efficient than map interface. However, there was not much difference in the effectiveness of the system for both the interfaces. Almost all the users linked records accurately. However, users were not happy about the usability of the system. They found it generally difficult to use.

5.2 Experiment 2

The main aim of the second experiment was to find to what extent the system was able to address all the research objectives listed in the starting of this section. The feedback from the first experiment was used to improve usability, increase performance and eliminate bugs. At this stage, it was possible to be sufficiently confident that the major usability issues had been eliminated and it would now be possible to more accurately compare the two approaches. In this experiment the users are divided into two categories: Map users and Table users. This is done to evaluate which user interface is better for representing semantic information. This chapter is divided into three parts. First the methods of the experiment are discussed. Then the evaluation results of the experiment are discussed. In the last, summary of the whole experiment is presented.

5.2.1 Method

This section discusses how the experiment was done. In this section, first the procedure of the experiment is discussed. Then the data collected as a part of the experiment is discussed. Then the section gives details of analysis approach taken to evaluate the system.

Procedure

In the second experiment, the participants were recruited by emails. Since the experiment does not require participants to have prior semantic web knowledge, the evaluation was open to everyone. Invitation for evaluation was sent to post-graduate students and researchers of the KDEG group of Computer Science department of Trinity College. In total, 10 participants took part in the evaluation. Among them, four participants took part in the first experiment also.

Before the experiment, the participants were asked to respond with their time of availability. The experiment was conducted according to their timings. All the participants chose to do evaluation on their machines. During the experiment, the researcher was sitting along with the participants to help them. At the starting of the experiment, the researcher briefly explained the participants about their task. This time the participants were asked to use only one interface. Half of the participants were asked to use Map interface and rest half were asked to use Table interface. They were asked to link only 20 links in a specific period of decades. Since, the participants had the liberty of choosing the records randomly; no gold standards were there to compare the results. They were also encouraged to give feedback or comments during the experiment.

During the experiment, the researcher noted down all the issues that participants faced while performing the task. All the participants completed their task. At the end of the experiment, they were asked to fill the questionnaire. The questionnaire mainly focused on the usability aspects of the system. Most of the questions in the questionnaire were same as in the first experiment.

Data

As a part of this experiment, the following information was collected: a) Questionnaire at the end of each experiment to get user feedback on usability of the tool. b) Actions of the users with the tool were recorded in the log file. The entire set of data recorded during the experiment is provided in a DVD.

5.2.2 Analysis Approach

Map users are those who used only map interface during the evaluation. Table users are those who used table interface during the evaluation. The evaluation of the developed system is done by following the given steps: a) Performance of the system is evaluated using the qualitative analysis of the time taken by the users to link the records. b) Accuracy of the system is evaluated based on relevant records that the system finds from Dbpedia. c) Accuracy of the links created by the participants is evaluated based on the information in the log file for each updated record. d) System Usability Scale (SUS) standard is used to evaluate the usability of the system.

5.2.3 Evaluation Results

This section details the results that came out of this experiment. The section is divided into three sub-sections. The first sub-section talks about the efficiency of the two user interfaces. The second sub-section describes the effectiveness of the two interfaces. The third sub-section gives details on the usability of the user interfaces. The entire data set of this experiment is provided in the accompanying DVD.

Efficiency

Generally, the efficiency of the user interface is evaluated based on the time taken to perform a function. In this section, the efficiency of both the interfaces is evaluated based on the time taken to link a record. Here, the records for which no links were formed are not considered. In case of the table interface, 79 records were successfully linked out of 100 records, whereas in the case of the map interface, 86 records were successfully linked out of 100 records.

Table 5.3 shows the number of records linked, total time taken, standard deviation and mean time taken to link the records for both map and table interface. From the given data, it can be inferred that there was almost equal variation in terms of time taken to link a record for both map and table interface users. Also, the mean time taken to link records is almost same for both the interfaces.

Table 5.3: Time taken by the users to link records (Second Experiment)

Interface Type	**Number of records linked**	**Total time taken (in seconds)**	**Standard Deviation (variation from the mean) σ**	**Mean (in seconds)**
Map Interface	86	588	6.0	6.84
Table Interface	79	503	5.9089	

Figure 5.4 shows the distribution of time taken to link the records using both interfaces. The distribution of time taken for both the interfaces is almost same. It signifies that both the map and table users completed the task in almost the same time. However, in the case of map interface for some records the time taken to link records was quite high as users had to zoom in the map to see the places.

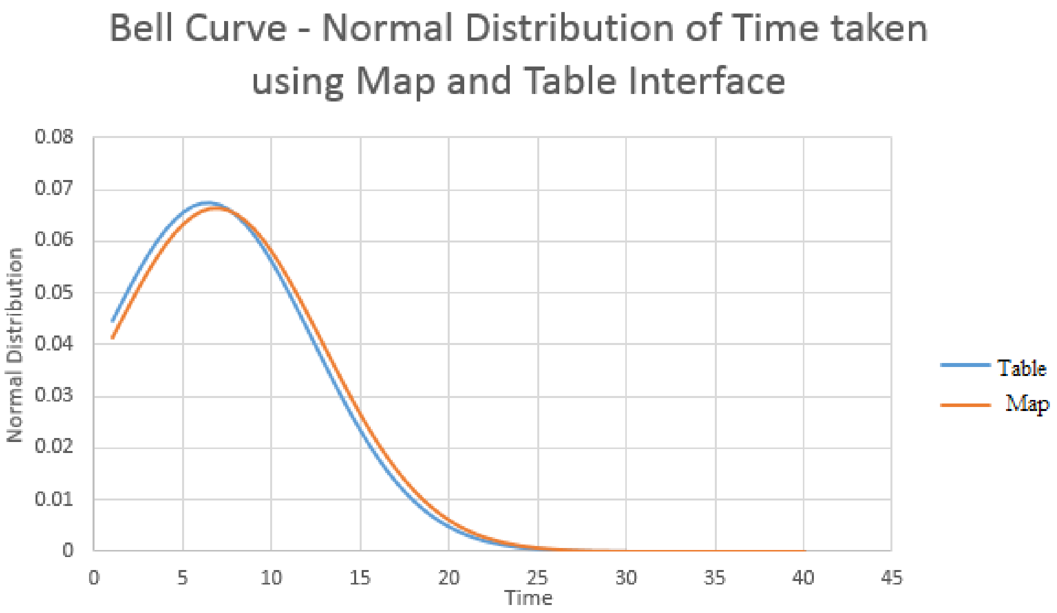


Figure 5.4: Normal Distribution of Time taken using Map and Table Interface

Effectiveness

Effectiveness of the system is evaluated on the basis of following metrics: a) *Relevant records* – Relevant records extracted from the DBpedia. b) *Percentage of links generated* – Number of links successfully linked by the users. c) *Accuracy of the links* – Accuracy of the links generated by the users.

Relevant records - When asked users ‘In your opinion, did the tool help you to find relevant records?’ seven out of ten users agreed with it, 2 users strongly agreed and one user was neutral about it. Another question which was asked to users was *‘Do you think the tool could have done better in terms of filtering irrelevant results?’* four out of ten users disagreed with it, two others were neutral and four users agreed with it.

Percentage of links generated – Table 5.4 shows the percentage of links generated by using both the interfaces. Map interface users linked 86% of the total records and table interface users linked 79% of the total records. However, the links generated by the users are greatly influenced by the ability of the system to extract relevant records from the DBpedia. It is difficult to say whether the users could not find relevant records because of the type of user interface they were using.

Table 5.4: Percentage of links generated by using both interfaces

Interface Type	**Number of records linked**	**Number of records could not be linked**	**Effectiveness (in terms of percentage)**
Map Interface	86	14	86%
Table Interface	79	21	73.75%

During the experiment, it was found that one of the main reason why the system could not find any relevant records for some records, is incorrect or misspelled location values in Dacura’s data set.

Figure 5.5 shows the user's response to the question *‘What percentage, of the total number of records for the time period you examined, do you think you linked during the experiment?’*

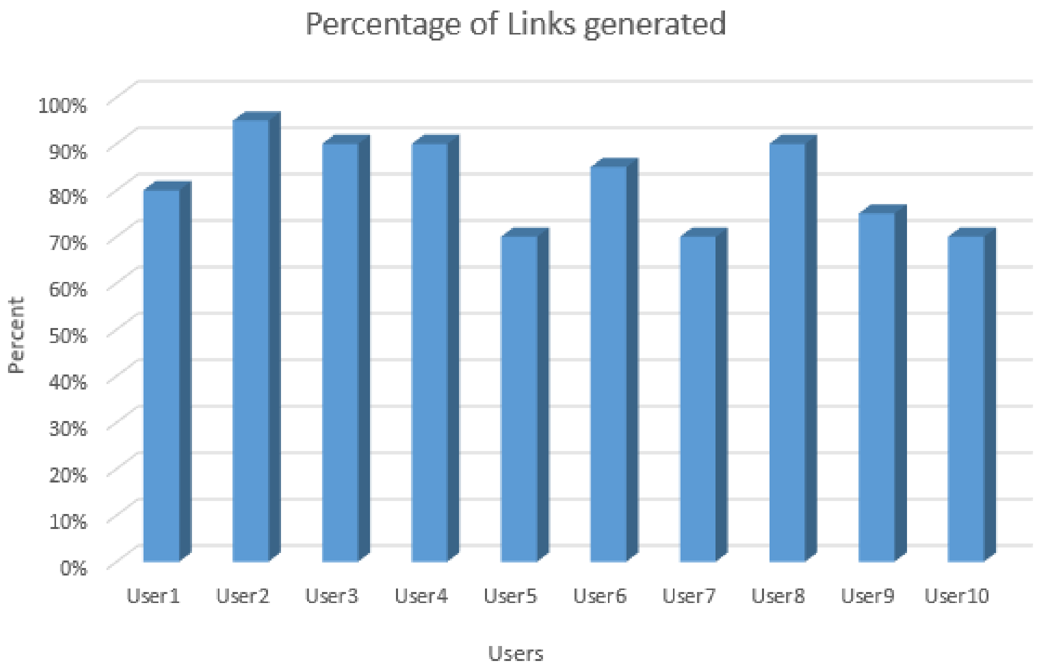


Figure 5.5: Percentage of links generated by each participant

Accuracy of the links – Accuracy of links is evaluated using the same approach as used in the first experiment. Each time the user update the record, the information related to the link is saved in the log file. To check the accuracy of the link, the Wikipedia link of the record is checked. After going through all the information in the log file, it was found that 95% of the records were linked correctly.

Usability

Usability of the system is evaluated based on the system usability score (SUS). On the basis of the user's response, SUS score 69.313 was calculated. Most of the users said that the system was easy to use and functions in the system were very well integrated. From the score, it can be inferred that the usability of the developed system was close to good.

One of the user commented that *'a lot of problems were fixed. Remaining problems are more database related than with the actual program (e.g. events without location, misspellings, multiple locations etc.)'*. Another user commented that *'system should be able to create multiple links for location attribute with multiple places'*.

5.2.4 Summary of the experiment

This experiment evaluates the research objectives listed at the starting of this chapter. The study analyses the efficiency, effectiveness and usability of the linking system using both UI's. In this experiment, the results revealed that both the user interfaces were equally efficient. Users of both categories generated links in almost equal time. The study also revealed that the effectiveness of the system majorly depends on the number of relevant records extracted from Dbpedia. However, due to inconsistencies in the database, the system's effectiveness could not be evaluated fully. Also, almost all the users linked the places correctly. This shows that the users did not face much problem in identifying potential matches.

In the next section of this chapter, detailed comparison of both the experiments is presented. The purpose of this comparison is to evaluate how the system improved from first to second experiment.

5.3 Comparison of two experiments

This section is divided into three parts: Efficiency, Effectiveness and Usability. In this section the comparison of two experiments will be presented based on these three parts. First the efficiency of the user interfaces in both the experiments is presented. Then the effectiveness of the whole system in both the experiments is presented and finally, usability of system is compared.

5.3.1 Efficiency

Table 5.5 shows the comparison of interfaces in two experiments. From the given data it can be inferred that the efficiency of the map interface drastically improved in the second experiment. Users took less time to link records than in the first experiment. However, there is not much improvement in the table interface. Users took almost equal time to link records in both the experiments.

Table 5.5: Comparison of interfaces in two experiments

Interface Type	**Experiment 1 (Standard Deviation)**	**Experiment 2 (Standard Deviation)**	**Experiment 1 (Mean time)**	**Experiment 2 (Mean time)**
Map Interface	29.775	6.0	18.789	6.84
Table Interface	6.044	5.9089	7.93	6.376

5.3.2 Effectiveness

Effectiveness of the system is compared on the basis of three metrics: Relevant records, percentage of links generated and accuracy of the links. Most of the participants in both the experiments agreed that the tool helped them in finding relevant records. After going through the user's response in both the experiments, it is found that the accuracy of the system in terms of finding records is increased. In the first experiment, majority of the participants said that the system found 50-70% of the accurate records, whereas in the second experiment majority of participants answered 70-90%. The percentage of records linked is increased in the second experiment. There is not much difference in the accurate links generated by the users. In both the experiments, users were able to correctly identify the potential matches between the records.

5.3.3 Usability

Usability of the system is improved in the second experiment. In the first experiment, almost all the users complained about the inconsistencies in the system, whereas in second experiment most of the users found various functions in the system very well integrated. Also, the usability score of the system also improved.

5.4 Summary of Evaluation

In this chapter, the evaluation of both the experiments was presented. First the research objectives were discussed and then both the experiments were discussed in detail. All the research objectives are answered in each experiment. On the basis of the results and data collected, it can be said that users using map interface took more time to link records even though, the time taken by the users using Map interface reduced significantly in the second experiment. The effectiveness of the system improved significantly after the second experiment. However, it is found that the use of different interfaces had no effect on the system's effectiveness. Both the interfaces were equally effective. Usability of the system improved significantly after the first experiment.

Chapter 6

Conclusion and Future Work

In this chapter, the first section discusses to what extent the research objectives are achieved. The second section presents the contributions achieved in this research. The third section discusses the future work. This section ends with the final remarks covered as a part of the fourth section.

6.1 Research Objectives

In chapter 5, three research objectives were introduced, see section 5.1. This section describes to what extent those research objectives are answered.

6.1.1 Accuracy of the implemented system

Objective: *Determine how accurate the system is, in terms of finding the relevant records.*

This objective was evaluated over two experiments. The accuracy of the system in the first experiment was significantly worse than it was in the second experiment. The reason behind this was the lack of support for the users in the first experiment. In the second experiment, users were provided with the manual search option in all the interfaces, which significantly improved the accuracy of the system. Overall, the findings in this dissertation reveal that the accuracy of the system gets better with a reduction of inaccurate records in the database.

6.1.2 Efficiency of user interfaces

Objective: *Determine which user interface requires more time to complete the interlinking task*

The efficiency of the user interface over two experiments was evaluated based on the time taken by the users to link records. Efficiency of the map interface in the first experiment was not as good as in the second experiment. The number of records returned by the system in the first experiment was significantly higher than in the second experiment. Due to this, the users took more time to go through all the records on the map interface. However, the efficiency of both interfaces was almost same in the second experiment.

6.1.3 Accuracy of the links

Objective: *Determine if the links created by participants are accurate*

Over the two experiments, it was found that the accuracy of the links generated by the users was almost same. The users were well aware of the places and system helped them identify potential matches.

6.2 Contributions

This section discusses the contributions of this research to the state of the art.

6.2.1 Improving the usability of Linked Data applications

Most of the approaches in the state of the art are concerned about improving the usability of single user interfaces. These approaches do not give many options to the user. The users are bound to use single interface. However, allowing the usage of the alternate user interfaces provides more insights into the expectation of the user. This allows building more usable and efficient user interfaces.

6.2.2 Improving the process of Interlinking

The approaches discussed in the state of the art provide various techniques to improve the interlinking process. However, not much attention is given to the user interface and the needs of the user of these applications. The iterative model approach discussed in this dissertation helps in identifying the difficulties and the required support to the users during interlinking.

6.2.3 Exploration of user interfaces in Linked Data

There has been no research on the examination of the alternate user interfaces in the field of linked data. This type of research is necessary to build user-friendly linked data applications. The approach taken in this dissertation revealed that the usability and the linking process improved significantly over two experiments. The research also revealed the interface that the users liked the most. The research investigation and exploration of user interfaces helps in finding most appropriate user interface for linked data applications.

6.3 Future Work

There are some limitations with the developed system. Currently, the system is only configured to find links of single location at one go. The system can be improved to handle the records which involve more than one location. More interfaces can be developed alternately, which adapts to the user actions. This will help in building more effective linking systems.

6.4 Final Remarks

Data-interlinking is a process of linking entities of different data sets, which represents the same individuals or places in the real world. However, linking millions of links is a cumbersome task and requires a lot of effort. This dissertation presents a new approach of improving interlinking in the web of linked data. The system is developed based on semi-automatic approach that involves users for validating links. This dissertation states the importance to explore design space in the context of the user interface of the system. The user interface of any application plays an important role in its success. In order to improve the data interlinking task, two different user interfaces, map and table, were used. This approach was followed to find how alternate user interfaces would help users in examining the records to be linked. Overall, this research demonstrates that the efficiency, effectiveness and usability of the applications can be improved by exploring alternate user interfaces.

Chapter 7

References

[Bizer2009] Christian Bizer, Tom Heath, and Tim Berners-Lee, Linked data-the story so far, In proceedings of International journal on semantic web and information systems, vol 5, no. 3, pp. 1-22, March 2009.

[Hsu2012] I-Ching Hsu, Hsu-Yang Lin, Lee Jang Yang, and Der-Chen Huang, Using Linked Data for Intelligent Information Retrieval, In Proceedings of the Joint 6th International Conference on Soft Computing and Intelligent Systems and 13th International Symposium on Advanced Intelligent Systems, pp 2172-2177, 20-24 November 2012

[Heim2011] Philipp Heim, Thomas Schlegel, and Thomas Ertl, "A model for human-computer interaction in the semantic web.", In Proceedings of the 7th International Conference on Semantic Systems, pp. 150-158, New York, NY, USA, ACM, 2011.

[Carlos2010] Carlos, Granell, Carlos Abargues, Laura Díaz, and Joaquín Huerta, "Interlinking geoprocessing services.", In Proceeding of the Second International Conference on Advanced Geographic Information Systems, Applications, and Services (GEOPROCESSING), pp. 99-104. IEEE, 10-16 Feb. 2010.

[Celino2012] Irene Celino, Simone Contessa, Marta Corubolo, Daniele Dell'Aglio, Emanuele Della Valle, Stefano Fumeo1, and Thorsten Kruger, Linking Smart Cities Datasets with Human Computation – The Case of UrbanMatch, In Proceedings of the 11th International Semantic Web Conference, pp 34-49 Boston, MA, USA, November 11-15, 2012.

[Hildebrand2012] Michiel Hildebrand and Jacco van Ossenbruggen, Linking User Generated Video Annotations to the Web of Data, In Proceedings of the 18th International Conference, pp 693-704, MMM 2012, Klagenfurt, Austria, January 4-6, 2012.

[Macias2012] Jos´e Antonio, Enhancing Interaction Design on the Semantic Web: A Case Study, IEEE Transactions on Systems, Man, and Cybernetics—part C: Applications and Reviews, vol. 42, no. 6, pp. 1365 – 1373, November 2012.

[Goldbeck2008] Jennifer Golbeck, Duane Degler, Abraham Bernstein, Lloyd Rutledge and m.c. schraefel, Semantic web user interactions: exploring hci challenges, In proceedings of CHI'08 Extended Abstracts on Human Factors in Computing Systems, pp. 3929-3932, ACM, 2008.

[Siorpaes2008] Katharina Siorpaes and Martin Hepp, Games with a Purpose for the Semantic Web, IEEE Intelligent Systems, vol 23 no 3, p.50-60, IEEE, May-June 2008

[Cuel2012] Roberta Cuel, Monika Kaczmarek and Elena Simperl, Making your Semantic Application addictive: incentivizing users!, In Proceedings of the 2nd International Conference on Web Intelligence, Mining and Semantics, page 4, ACM New York, NY, 2012.

[Saxena2010] Tripti Saxena, and Gabor Karsai, Towards a generic design space exploration framework, In proceedings of Computer and Information Technology (CIT), IEEE 10th International Conference on, pp. 1940-1947, 2010.

[Ma2000] Kwan-Liu Ma, Visualizing visualizations: User interfaces for managing and exploring scientific visualization data, Computer Graphics and Applications, IEEE volume 20, Issue no. 5, pp. 16-19, Sept/Oct 2000.

[Augenstein2012] Isabelle Augenstein, Sebastian Padó, and Sebastian Rudolph, Lodifier: Generating linked data from unstructured text, In the proceedings of 9th Extended Semantic Web Conference, ESWC 2012, Heraklion, Crete, Greece, May 27-31, 2012.

[Nikolov2008] Andriy Nikolov, Victoria Uren, Enrico Motta and Anne de Roeck, Integration of Semantically Annotated Data by the KnoFuss Architecture, In the proceedings of 16th International Conference, EKAW 2008, Acitrezza, Italy, pp 265-274, September 29 - October 2, 2008.

[Araujo2011] Samur Araujo, Jan Hidders, Daniel Schwabe and Arjen P. de Vries, SERIMI - Resource Description Similarity, RDF Instance Matching and Interlinking, CoRR, abs/1107.1104 (2011)

[Levenshtein1966] Levenshtein, Binary Codes Capable of Correcting Deletions, Insertions and Reversals, In proceedings of Russian—translation in Soviet Physics Doklady, vol. 10, Issue no. 8, pp. 707-710, 1966.

[Hamming1950] Hamming R. W., Error detecting and error correcting code, Bell System Technical Journal, 29 (2): pp. 147–160, April 1950.

[Schandl2010] Thomas Schandl and Andreas Blumauer, Poolparty: SKOS thesaurus management utilizing linked data, In the proceedings of 7th Extended Semantic Web Conference, ESWC 2010, Heraklion, Crete, Greece, pp 421-425, May 30 – June 3, 2010.

[Hassanzadeh2009] Oktie Hassanzadeh, Reynold Xin, Ren´ee J. Miller, Anastasios Kementsietsidis, Lipyeow Lim and Min Wang, Linkage Query Writer, In the proceedings of VLDB Endowment, Volume 2 Issue 2, pp. 1590-1593, August 2009.

[Lehmann2013] J. Lehmann, Q. Nguyen, and T. Ermilov, Can we create better links by playing games?, In proceedings of 7th IEEE International Conference on Semantic Computing, September pp. 16-18, 2013, Irvine, California, USA, 2013.

[Monge1996] Alvaro E. Monge and Charles P. Elkan, The Field Matching Problem: Algorithms and Applications, In the proceedings of second international conference of. Knowledge Discovery and Data Mining (KDD), pp.267 -270, 1996

[Fellegi1969] Fellegi, I. P. and Sunter, A. B., A theory for record linkage, Journal of the American Statistical Association, Vol. 64, No. 328, pp. 1183, Dec. 1969.

[Ahn2008] Luis von Ahn and Laura Dabbish, Designing games with a purpose, Communications of the ACM, vol 51 no 8, ACM, August 2008.

[Koutsomitropoulos2011] Dimitrios A. Koutsomitropoulos, Ricardo Borillo Domenech, Georgia D. Solomou, A structured semantic query interface for reasoning-based search and retrieval, In Proceedings of 8th Extended Semantic Web Conference, ESWC 2011, pp. 17-31, Heraklion, Crete, Greece, May 29-June 2, 2011.

[Heath2006] Tom Heath, John Domingue and Paul Shabajee, User interaction and uptake challenges to successfully deploying Semantic Web technologies, In proceeding of The 3rd International Semantic Web User Interaction Workshop SWUI2006 at ISWC2006, Athens, GA, USA (November 2006)

[García2006] Roberto García and Rosa Gil, Improving Human-Semantic Web Interaction: The Rhizomer Experience, In Proceedings of the 3rd Italian Semantic Web Workshop, CEUR-WS, vol. 201, pp. 57–64 (2006)

[Wolger2011] Stephan W¨olger, Katharina Siorpaes, Tobias B¨urger, Elena Simperl, Stefan Thaler and Christian Hofer, A survey on Data Interlinking Methods, Technical report. Semantic Technology Institute, Innsbruck, University of Innsbruck, March 2011.

[Partarakis2009] Nikolaos Partarakis, Constantina Doulgeraki, Asterios Leonidis, Margherita Anton1, and Constantine Stephanidis, User Interface Adaptation of Web-Based Services on the Semantic Web, In Proceedings of the 5th International Conference, UAHCI, Held as Part of HCI International, San Diego, CA, USA, pp. 711-719, July 19-24, 2009.

[Lewis2009] J. Lewis and J. Sauro, The Factor Structure of the System Usability Scale, In Proceedings of First International Conference, HCD 2009, Held as Part of HCI International, San Diego, CA, USA, pp 94-103, July 19-24, 2009

[uxmatters2013] http://www.uxmatters.com/mt/archives/2013/12/redesigning-a-knowledge-management-system-for-usability.php

[Alexander2009] Keith Alexander, Richard Cyganiak, Michael Hausenblas and Jun Zhao, Describing Linked Datasets - On the Design and Usage of voiD, the 'Vocabulary of Interlinked Datasets', in conjunction with 18th International World Wide Web Conference (WWW09, Madrid, Spain, 2009.

[Ferrara2011] Alfio Ferrara, François Scharffe and Andriy Nikolov, Data linking for the semantic web, International Journal on Semantic Web and Information Systems, volume 7, Issue no. 3 , pp. 46-76, July 2011.

[Hausenblas2008] Michael Hausenblas and Wolfgang Halb, Interlinking multimedia data, In the proceedings of Linking Open Data Triplification Challenge at the International Conference on Semantic Systems (I-Semantics' 08), pp. 2007-2008, 2008.

[Hausenblas2008a] M. Hausenblas and W. Halb, Interlinking of Resources with Semantics, In the proceedings of 5th European Semantic Web Conference (ESWC2008), Tenerife, Spain, 2008.

[Hausenblas2008b] M. Hausenblas, W. Halb, and Y. Raimond, Scripting User Contributed Interlinking, In proceedings of the 4th Workshop on Scripting for the Semantic Web, Tenerife, Spain, 2008.

[Ahn2009] Von Ahn, Luis, Human computation, In proceedings of Design Automation Conference, 2009, 46th ACM/IEEE, pp. 418-419, 26-31 July 2009.

[Volz2009] Julius Volz, Christian Bizer, Martin Gaedke and Georgi Kobilarov, Discovering and Maintaining Links on the Web of Data, In proceedings of 8th International Semantic Web Conference, ISWC 2009, Chantilly, VA, USA, pp 650-665, October 25-29, 2009.

[Turnbull2007] Douglas Turnbull, Ruoran Liu, Luke Barrington, Gert Lanckriet, A game-based approach for collecting semantic annotations of music, In Proceedings of 8th Intl. Conf. on Music Information Retrieval (Vienna), pp. 535--538, September 23-272007.

[Hsu2012] I-Ching Hsu, Hsu-Yang Lin, Lee Jang Yang, and Der-Chen Huang, Using Linked Data for Intelligent Information Retrieval, In Proceedings of the Joint 6th International Conference on Soft Computing and Intelligent Systems and 13th International Symposium on Advanced Intelligent Systems, pp 2172-2177, 20-24 November 2012

[Prud'hommeaux2006] E. Prud'hommeaux and A. Seaborne, SPARQL Query Language for RDF. W3C Candidate Rec. 6 April 2006, http://www.w3.org/TR/rdf-sparql-query/.

[P´erez2009] Jorge P´erez, Marcelo Arenas and Claudio Gutierrez, Semantics and complexity of SPARQL, ACM Transactions on Database Systems, Article No. 16, Volume 34, Issue 3, August 2009.

[Raimond2008] Yves Raimond, Christopher Sutton and Mark Sandler, Automatic Interlinking of Music Datasets on the Semantic Web, In Proceedings of the Linked Data on the Web (LDOW 2008) Workshop at 17th International World Wide Web Confernce (2008)

[Feeney2014] Kevin Feeney, Declan O'Sullivan, Wei Tai, Rob Brennan, Improving curated web-data quality with structured harvesting and assessment, International Journal On Semantic Web and Information Systems (IJSWIS), 2014

[Kobilarov2009] Georgi Kobilarov, Christian Bizer, Jens Lehmann, Soren Auer, Christian Becker, Richard Cyganiak and Sebastian Hellmann, DBpedia - A Crystallization Point for the Web of Data, In proceedings of Web Semantics: Science, Services and Agents on the World Wide Web 7 (3): 154–165, September 2009.

[Halb2008] Wolfgang Halb, Michael Hausenblas, Yves Raimond and Tom Heath, What is the size of the Semantic Web, In Proceedings of I-Semantics 2008, JUCS, Graz, Austria, pp. 9–16 2008.

[Nikolov2008] Andriy Nikolov, Victoria Uren, Enrico Motta and Anne de Roeck, Integration of Semantically Annotated Data by the KnoFuss Architecture, In proceedings of 16th International Conference, EKAW 2008, Acitrezza, Italy, pp 265-274, September 29 - October 2, 2008.

[Rowe2009] M Rowe, Interlinking distributed Social Graphs, In Proceedings of Linked Data on the Web Workshop, WWW 2009, Madrid Spain (2009)

Appendix A

Evaluation Material

A.1 Experiment 1

1. **Questionnaire related to developed system**

Table A.1: Percentage of records linked

Questions	User1	User2	User3	User4	User5	User6
What percentage, of the total number of records for the time period you examined, do you think you linked during the experiment?	80%	70%	80%	50%	70%	80%

Table A.2: User Interface

Questions	Map	Table	None
Which user interface you found most difficult to use?	4	2	

Table A.3: Time Taken - User Interface

Questions	Map	Table	None
Which user interface took a long time to complete the linking task?	5	1	0

Table A.4: Accuracy of the system

Questions	10-30%	30-50%	50-70%	70-90%	90-100%
In your opinion, how much accurate the system was in terms of finding records?	1		3	2	

2. **Table of Usability questionnaires and user responses**

Table A.5: Usability questionnaire and user responses

Questions	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
I thought the system was easy to use	1	3	1	1	
I found the system unnecessarily complex		2	2	2	
I thought there was too much inconsistency in this system	1	1	1	2	1
I felt very confident using the system	1	1	2	2	
I found the various functions in this system were well integrated	1	3	1	1	
I would imagine that most people would learn to use this system very quickly		2	2	2	
I needed to learn a lot of things before I could get going with this system:	1	2	3		
I think that I would need the support of a technical person to be able to use this system	1	3	1	1	
In your opinion, did the tool help you to find relevant records?		1	2	3	
Do you think the tool could have done better in terms of filtering irrelevant results?		1	2	2	1

A.2 Experiment 2

1. **Questionnaire related to developed system**

Table A.6: Percentage of records linked

Questions	**User1**	**User2**	**User3**	**User4**	**User5**	**User6**	**User7**	**User8**	**User9**	**User10**
What percentage, of the total number of records for the time period you examined, do you think you linked during the experiment?	80%	95%	90%	90%	70%	85%	70%	90%	75%	70%

2. **Table of Usability questionnaire and user responses**

Table A.7: Usability questionnaire and user responses

Questions	**Strongly Disagree**	**Disagree**	**Neutral**	**Agree**	**Strongly Agree**
I thought the system was easy to use	1		1	7	1
I found the system unnecessarily complex	1	7	2		
I thought there was too much inconsistency in this system	2	6	1	1	
I felt very confident using the system			1	9	
I found the various functions in this system were well integrated			3	6	1
I would imagine that most people would learn to use this system very quickly		1	1	5	3
I needed to learn a lot of things before I could get going with this system:	4	6	3		
I think that I would need the support of a technical person to be able to use this system	3	5	2		
In your opinion, did the tool help you to find relevant records?			1	7	2
Do you think the tool could have done better in terms of filtering irrelevant results?		4	2	4	1

Appendix B

Information Material

B.1 Information Sheet

In this experiment you will be asked to create links between records in the Dacura application and places in DBpedia. There are two different user interfaces: Map and Data Table. Your task is to create 50 links, 25 with the Map interface and 25 with the data table interface. Please make sure that pop-ups are enabled in your browser to see error messages.

Instructions

1. Go to http://dacura.cs.tcd.ie/Rajan/browse-uspv.html page
2. Go through 25 records and use the Map interface to link them to DBpedia places (records are already linked – have no links yet)
3. Go through 25 records and use the Table interface to link them to DBpedia places
4. Fill in the survey at: https://docs.google.com/a/tcd.ie/forms/d/1VlhXa9p5lH2gPDkk0NNOXn-QjCSkae6mIY8pSnW3kk/viewform

Notes

If the application could not find any relevant instances from DBpedia, a new window will be opened in a dialog box. If this happens, type in the name of the city or place and select the most appropriate record from the suggestions.

Data Table Interface:

1. Potentially relevant records (based on the location attribute of the record in the table) will be shown in a new data table. Please select the most appropriate option from the new data table.
2. If you are not sure which record to select, you can check the Wikipedia link of each record.
3. After updating the record, the icon of the button will change, indicating that the record has been successfully updated.

Map Interface:

1. When the Map Interface option is selected, the Map will open in a new tab where markers indicate the locations on the map. Each marker corresponds to a different location on the Map.
2. Click on the appropriate marker representing the correct location and click on the update button.
3. Before updating any record, you can visit the Wikipedia page of that record.
4. Once the record gets updated, you will be asked to confirm leaving that page. You may need to refresh the page to see updated records.

Your actions with the system may be recorded by the software. At the end of the experiment you will be asked to fill a short survey related to your experience with both interfaces. Your feedback will allow researcher to further improve the functionality as well as the user interface of the application.

B.2 Invitation Email

Hi All,

Thanks very much to those of you who completed Rajan's first experiment. He's processed the results and has cleaned up the major problems with the UI. As the final part of his experiment, he'd like to ask you to do one more task. This will involve another brief usage of the system - Rajan will come to you and explain what he needs you to do.

Rajan is CCed on this mail and he will contact you to try to arrange a time to come see you.

Thanks again for your help
Kevin

Appendix C

Table of Contents of the accompanying DVD

1. Application code.
2. User Evaluation Questionnaire responses in spread sheet.
3. Log files – User actions with the system.