THE UNIVERSITY OF
WAIKATO
Te Whare Wānanga o Waikato

User-guided Disambiguation for Semantic-enriched Search

Xinning Ren

This thesis is submitted in partial fulfillment of the requirements for the Degree of Master of Science (Research) at the University of Waikato.

July 2015

Abstract

In this thesis, we develop, implement and evaluate a strategy for user-guided disambiguation for semantic-enriched search. We use the prototype system Capisco that combines some features of traditional text-based search engines and semantic search engines. Unlike many semantic search approaches that require formulating queries in quite complex semantic languages, the Capisco system is a keyword-based search engine.

In this work, we focus on the user-guided disambiguation which helps to capture the users' information needs. The user-guided disambiguation developed in this thesis explores the semantic meanings for a user-provided keyword and displays these to the users for selection. We explored three different orderings in which the possible keyword meanings may be shown. Using our newly-developed disambiguation interface, the user selects one or several of these meanings from the list offered. The selected semantic meanings are then transferred to the search engine to identify matching documents.

This thesis provides an analysis of related work on search interfaces in semantic search, and in methods for semantic disambiguation. We developed the concept of user-guided disambiguation and implemented this concept in a prototype that explored three interface variations. We executed a user study on the three interface alternatives. Finally, we discuss the insights gained during this project, compare our approach to existing literature, and explore possible future work.

Contents

1. Introduction

Web search engines play an important role in providing users with useful information selected from the wealth of information available on the internet. Traditional search engines use keyword-based search that finds the documents that contain some or all of the user's input keywords (Bast, Bäurle & Buchhold, 2014). The keywords are matched to the documents literally, rather than understanding their semantic meanings. For example, when a user enters the keyword "Kiwi", the documents that contain the term "Kiwi "are displayed to the user. But the search engine does not know if the "Kiwi" really refers to New Zealand people, Kiwi fruit or Kiwi birds.

Semantic search engines aim to improve the accuracy of searching by analyzing the contextual meaning of the query and understanding the users' potential intent. However, many of the semantic search engines require their users to formulate the queries using a semantic query language such as SPARQL[1], which is very complicated to understand and manipulate, especially for users who are not technically proficient such as humanities scholars.

This thesis works with the semantic-enriched search engine Capisco, which is currently under development at the University of Waikato (Hinze, Taube-Schock, Bainbridge, Matamua & Downie, 2015). Capisco aims to provide a semantic-enhanced search that uses keyword-based queries. Using Capisco users do not need to have expertise in semantic query languages or syntax. Instead users input keywords, which are then assigned respective semantic concepts using the user-guided disambiguation developed in this thesis.

1.1 Focus and Aim of this thesis: user-guided disambiguation

Given the keyword-based queries used in Capisco, the intent and semantics of the user query is not as strongly defined as in semantic search engines. In semantic search, the input query terms need to follow a strongly defined semantic relating to RDF-S and OWL-based ontologies. In order to identify the semantics of search terms used for Capisco, this thesis will explore user-guided disambiguation. This requires the user to identify their intended meaning for a given keyword. In this thesis, three interface options are designed for Capisco.
This thesis introduces the concept of user-guided disambiguation, explains how this concept applies for the Capisco prototype and evaluates three disambiguation interfaces designed for the Capisco system. The three disambiguation interfaces are explored in a user study to identify which option best supports user-guided disambiguation.

---

[1] www.w3.org/TR/rdf-sparql-query/

1.2 Methodology and outline of the thesis

The purpose of this thesis is to examine users' information seeking behavior in semantic-enriched search engine Capisco with interfaces for user-guided disambiguation.
We first introduce both keyword-based and semantic search concepts. We study related work from these areas to identify advantages and disadvantages of these search interfaces. The thesis then introduces our semantic-enhanced search engine Capisco and the concepts of user-driven disambiguation. We introduce three ways of presenting the user with information for the manual disambiguation. We investigate how the three different disambiguation interfaces impact on users' search strategies. In a user study, 24 participants were invited to use the interfaces for disambiguation. Finally, we evaluate and compare the feedback for all three interfaces from the participants. The thesis concludes with a summary of the contributions and insights and a discussion of possible future work.

The outline of this thesis is as follows:

- Chapter 2 describes two search approaches: traditional keyword-based search and the semantic search, and introduces the semantic web briefly.
- Chapter 3 gives an overview of related work on interfaces for both keyword-based and semantic search. We provide a brief analysis and comparison of them.
- Chapter 4 introduces the Capisco system and explains its general architecture and concept.
- Chapter 5 introduces the concept of user-guided disambiguation, proposes three disambiguation interfaces and demonstrates how to use these interfaces. We also discusses the differences between our user interface and the related works.
- Chapter 6 describes the user study and evaluates the data obtained during the study and the participant feedback from the usage experiments and compare the three disambiguation interfaces.
- Chapter 7 discusses the wider impact of the outcome of the user study in the context of selected previous related work.
- Chapter 8 provides a conclusion and a discussion of possible further work directions.

2. Background

Search engines, such as Google,[2] Yahoo,[3] and Bing[4] are very popular and help users finding useful documents in massive online information (Bast, Bäurle & Buchhold, 2012). However, the traditional keyword-based search has shortcomings in fulfilling users' search requirements. Semantic web search and semantic-enriched search are addressing some of the shortcomings. In this section, we introduce the traditional keyword search (Section 2.1) and semantic search (Sections 2.2).

A scenario is used to help comparing the differences between the different approaches. In our scenario, a user wants to search for a person who is both painter and mathematician.

2.1 Traditional keyword-based search

In traditional search, such as full-text index search, the search engines mostly rely on the occurrence of keywords that appear in the text (Witten, Gori & Numerico, 2007). In order to do the full-text index search, the full-text index needs be to be built at first.

The first step lists all words existing in the text, along with the pointer to each appearance. For example, the text is "No pain, no gain", the beginning index is as shown in Figure 1(a). The pointer to each word is the number here. As in Figure 1(b), the number 1 and 3 beside "No" indicates that the word "No" appears as the first and third word in the text. Stop words, such as "the" are typically excluded from an index. Each word has as many pointers as the times it appears in the text, in other words, the total number of the pointers is the number of the words in the text. In our example, there are 4 words in the text and therefore it has 4 pointers. Figure 2 shows a simplified process of how a keyword-based index is built step by step.

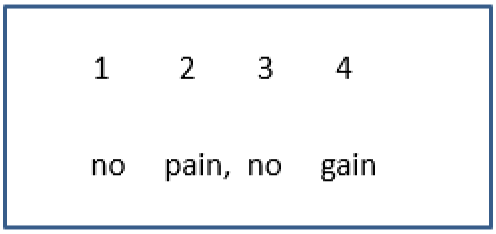


Figure 1(a): the text to be indexed

[2] www.google.com

[3] www.yahoo.com

[4] www.bing.com

Figure 1(b): the index (simplified)

Figure 1: This figure displays a simple example of a keyword index.

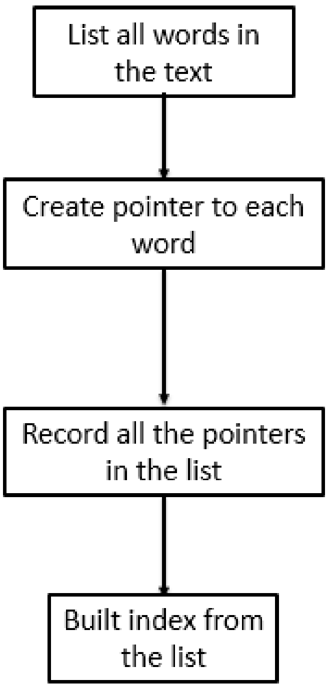


Figure 2: The diagram displays the proceeding that how full-text index is built.

By the given index, the users are enabled to do the search. For the one-word search, the keyword is located in the ordered list. The keyword's list of document and word numbers are extracted. For the query that contains several terms, all the terms are located firstly and then their related lists are extracted as mentioned in one-keyword search. Then, all the lists are scanned in order to get the more common document numbers (the documents contains all or some of the terms). Afterwards, the lists are merged and the result documents are displayed.

The diagram in Figure 3 shows the brief process of index search.

.

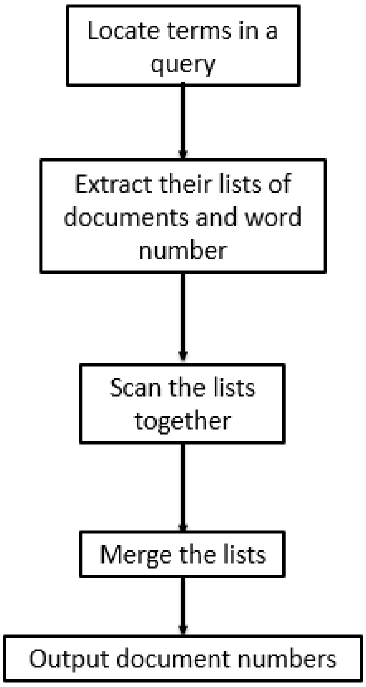


Figure 3: The diagram displays the proceeding that how keyword-based search work.

The documents which contains more query terms or the query terms occur more frequently, or contains less non-query terms, are considered as more relevant documents. Therefore, it is easy to search documents with specific words by using a given index.

However, the limitations of this kind of search are apparent, as the search literally provides the search results that contain some or all of the keywords (i.e., syntactic matching not semantic). As a consequence, it is difficult to answer some user questions. For example, if our user from the scenario wants to find people who are both painter and mathematician, she may enter the keywords “Painter” and “Mathematician” for a search (Figure 4). The results of traditional search would then be the documents that contain the words “Painter” or “Mathematician” (Figure 5), rather than documents about specific persons who are a painter and a mathematician. Figure 6 shows one document from search results, which is about the relation between mathematics and art.

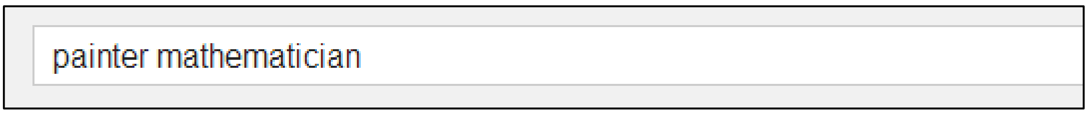
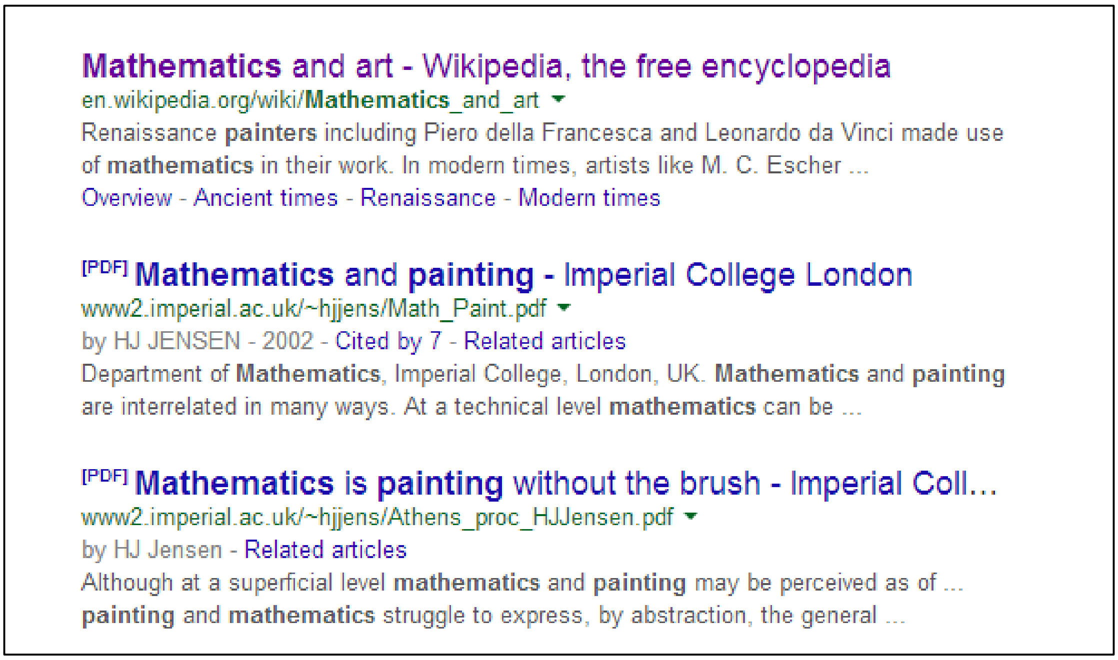


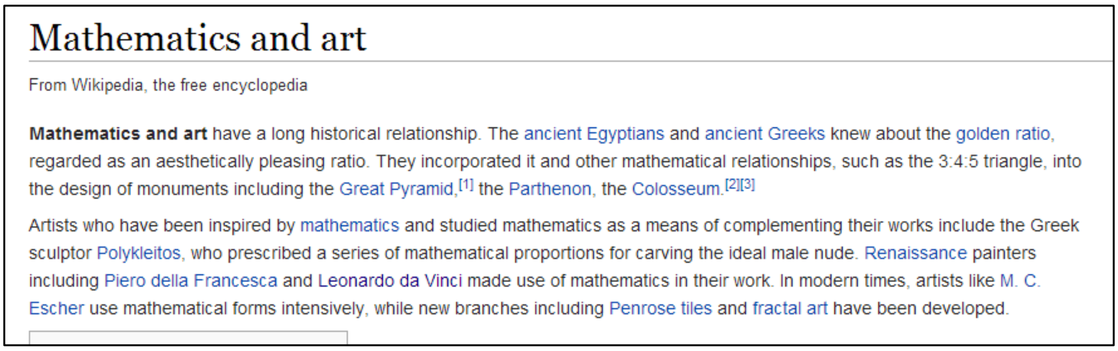
Figure 4: The interface for keyword input in semantic-enriched search


Mathematics and art - Wikipedia, the free encyclopedia
en.wikipedia.org/wiki/Mathematics_and_art
Renaissance **painters** including Piero della Francesca and Leonardo da Vinci made use of **mathematics** in their work. In modern times, artists like M. C. Escher ...
Overview - Ancient times - Renaissance - Modern times

[PDF] **Mathematics** and **painting** - Imperial College London
www2.imperial.ac.uk/~hjjens/Math_Paint.pdf
by HJ JENSEN - 2002 - Cited by 7 - Related articles
Department of **Mathematics**, Imperial College, London, UK. **Mathematics** and **painting** are interrelated in many ways. At a technical level **mathematics** can be ...

[PDF] **Mathematics** is **painting** without the brush - Imperial Coll...
www2.imperial.ac.uk/~hjjens/Athens_proc_HJJensen.pdf
by HJ Jensen - Related articles
Although at a superficial level **mathematics** and **painting** may be perceived as of ...
painting and **mathematics** struggle to express, by abstraction, the general ...

Figure 5: The search results of query “painter mathematician” using Google.


Mathematics and art

From Wikipedia, the free encyclopedia

Mathematics and art have a long historical relationship. The ancient Egyptians and ancient Greeks knew about the golden ratio, regarded as an aesthetically pleasing ratio. They incorporated it and other mathematical relationships, such as the 3:4:5 triangle, into the design of monuments including the Great Pyramid,[1] the Parthenon, the Colosseum.[2][3]

Artists who have been inspired by mathematics and studied mathematics as a means of complementing their works include the Greek sculptor Polykleitos, who prescribed a series of mathematical proportions for carving the ideal male nude. Renaissance painters including Piero della Francesca and Leonardo da Vinci made use of mathematics in their work. In modern times, artists like M. C. Escher use mathematical forms intensively, while new branches including Penrose tiles and fractal art have been developed.

Figure 6: The document to which the first search result from query “painter” and “mathematician” refers (shown in Figure 4).

A further problem is the disambiguation between the different meanings of search terms. For example, it may not be clear, if the term “painter” is meant to refer to an artist painter or a builder or house painter. Clarification of the semantics of search terms will be discussed later in Chapter 4. Examples such as the one given above and the described limitations motivated the development of semantic search.

2.2 Semantic web / Semantic Search

Before explaining semantic search, we give a brief introduction into the semantic web and related technologies.

2.2.1 Semantic Web / Document structure

The Semantic Web is the extension of the WWW (World Wide Web) which allows users to share and combine data across the boundaries of websites and applications. Based on semantic web technologies, users are enabled to create data and share them on the Web, build vocabularies, and write rules for handling data (W3C, 2013). The Semantic Web allows to specify the type of data that is stored, such as naming places and people. These data descriptions are implemented by technologies, such as RDF, SPARQL and OWL. Instead of mere text-based documents, the semantic web provides semi-structured information as explained below.

RDF (Resource Description Framework)

RDF is a standard model for semantic web, which is used as one scheme for description of concepts or modeling of information. It extends the linking structure of the web by using URIs to name the relationship between things and the two ends of the links, which is usually referred to as a "tripe". (W3C semantic web, 2014). W3C standard RDF serialization formant includes N-Triples, XML/RDF, Turtle, etc. The N-Triples and XML/RDF are explained in detail in the following. Figure 7 shows an RDF structure representing information about the Da Vinci who is a painter and mathematician.

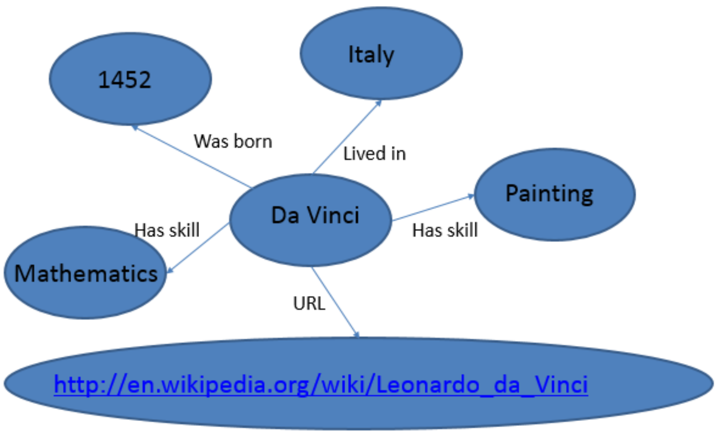


Figure 7: Displaying the data stored in RDF for example "Da Vinci".

N-Triple is a simple and line-based format for serializing RDF. A line in N-Triples is a sequence of RDF terms which represents the subject, predicate and object of an RDF Triple (see Figure 8(a)). Figure 8(b) is the N-Triples format example for "Da Vinci" as a graph structure.

```
<http://one.example/subject1> <http://one.example/predicate1> <http://one.example/object1>
_:subject1 <http://an.example/predicate1> "object1" .
_:subject2 <http://an.example/predicate2> "object2" .
```

Figure 8(a): Structure of N-Triples.

```
<http://www.w3.org/2001/sw/RDFCore/ntriples/><http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
        <http://xmlns.com/foaf/0.1/Document> .
 <http://www.w3.org/2001/sw/RDFCore/ntriples/> <http://purl.org/dc/terms/title> "N-Triples"@en-US .
 <http://www.w3.org/2001/sw/RDFCore/ntriples/> <http://xmlns.com/foaf/0.1/maker> _:art .
_:art <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://xmlns.com/foaf/0.1/Person> .
_:art <http://xmlns.com/foaf/0.1/name> "Da Vinci".
_:art <http://xmlns.com/foaf/0.1/country> "Italy".
_:art <http://xmlns.com/foaf/0.1/name> "Painting".
```

Figure 8(b): Displaying the data stored in RDF (N-triples) for example "Da Vinci".

RDF/XML is an XML-based syntax for RDF. RDF graph contains nodes and arcs which are represented as a set of RDF Triple (subject node, predicate, and object node). The nodes and predicate arcs are represented in XML terms — element names, attribute names, element contents and attribute values (W3C Recommendation, 2006). As can be seen, RDF/XML is a very verbose syntax that is difficult to read. Figure 9 is the same structure in XML/RDF format for example "Da Vinci".

```
<source lang="xml">
 <rdf:RDF xmlns="http://xmlns.com/foaf/0.1/"
        xmlns:dc="http://purl.org/dc/terms/"
        xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" >
  <Document rdf:about="http://www.w3.org/2001/sw/RDFCore/ntriples/">
   <dc:title xml:lang="en-US">N-Triples</dc:title>
   <maker>
    <Person rdf:nodeID="art">
     <name>Da Vinci</name>
     <country>Italy</country>
     <skill>Painting</skill>
    </Person>
   </maker>
  </Document>
 </rdf:RDF>
</source>
```

Figure 9: Displaying the data stored in RDF (XML) for example "Da Vinci".

OWL (Ontology Web Language)

An ontology formally represents knowledge as a hierarchy of concepts within a domain. Ontologies use shared vocabularies to denote the types, properties and interrelationships of those concepts (Thomas, 1993). It is a structural framework to organize information and are be used in semantic web.

OWL is a semantic web language, which is a family of knowledge representation languages. OWL is a computational logic-based language, by which that knowledge can be exploited by computer programs (W3C Semantic Web, 2014). Like RDF, it can be serialized in RDF/XML syntax as well. Compared with RDF, it allows users to specify more about the properties and classes and can be expressed in triples. For example, it can indicate that "If A isMarriedTo B" then this implies "B isMarriedTo A".

The OWL language family support many of the syntaxes. Here are the examples of OWL using OWL2 functional syntax and Manchester syntax to represent an equivalent class musician, which is equivalent to a person who compose music or perform music. Figure 10 shows the OWL 2 functional Syntax for the example.

```
EquivalentClasses(:People
    ObjectIntersectionOf(:Person
        DataSomeValuesFrom(:hasSkill
        DataRestriction(
            xsd:painting
            xsd:mathematics))))
          ))))
```

Figure 10: An example of the OWL Functional Syntax used to represent a class description.

Manchester Syntax is created to provide non-logicians with a syntax that makes it easier to write ontologies (Horridge *et al.*, 2006). The Manchester Syntax is a compact syntax which style is close to frame languages. It can be used as an entailment-based querying language and helps users to construct and submit queries to ontology document (Koutsomitropoulos *et al.*, 2011). An example using Manchester syntax is shown in Figure 11.

```
Class: People
    EquivalentTo: Person and (hasSkill some painting) and (hasSkill some methematics)
```

Figure11: Manchester OWL Syntax used to represent a class description.

2.2.2 Semantic Search

Semantic search is an application of semantic web which is used to do the search (Guha, McCool & Miller, 2003).It aims at improving the performance of traditional search results by analyzing the contextual meaning of keywords and understanding the user's information need. Semantic search uses the semantic query language (e.g., SPARQL) to find the target documents which has been semantically encoded by using specific technologies (e.g., RDF). Because input query terms follow a strongly defined semantic language query, the matched results are identified according to the semantic web data.

As the search takes into consideration the contextual meaning of keywords, the efficiency and accuracy of searching results can be substantially improved. Also, semantic search may take location, intent, variation, synonyms, generalized and specialized queries, concept matching and natural language queries into consideration while searching result (John, 2012). Currently, search engines such as Google and Bing, have taken some of these elements into their search algorithm (Cambridge Semantics, 2014).

SPARQL

SPARQL is one query language for RDF documents. It works on RDF documents that have been encoded in a database. It can retrieve and manipulate data that are stored in RDF format, by which the target documents are able to be found. For example, the SQARQL query (shown as Figure 12(b)) below returns the name of a person who has skill as painting and mathematics (shown as Figure 12(c)) from given information (shown as Figure 12(a)).

```
@prefix                          foaf:
<http://xmlns.com/foaf/0.1/> .
_:a   foaf:name    "Da Vinci" .
_:a   foaf:skill    "Painting"
_a:   foaf:skill    "mathematics"
```

Figure 12(a): Figure displays the data of given information for the example.

```
PREFIX                       foaf:
<http://xmlns.com/foaf/0.1/>
SELECT ?name ?skill
WHERE
   { ?x foaf:name ?name .
      ?x foaf:skill ?skill}
```

Figure 12(b): Figure displays the query for the example.

Name	Skills
"Da Vinci"	Painting
	Mathematics

Figure 12(c): Figure displays the result for the example.

We have discussed the difference between the traditional search and semantic search in Sections 2.1 and 2.2. However, the syntax and query language for semantic search are very complicated, especially for those users who do not have background knowledge of semantic web.

2.3 Summary

In this section, we discuss the differences between traditional keyword-based search and semantic search. Traditional search has severe limitations, which may be addressed through semantic search. However, semantic search is so complicated for users who do not have ontology or knowledge base background to understand and manipulate.

We explore interfaces for both keyword-based search and semantic search in the next chapter.

3. Related work on semantic search interfaces

Much research has been done on user interfaces for web search, and a number of related projects have explored semantic web search interfaces. However, most of these semantic search systems are more or less complicated, especially in the terms of creating queries. Some of them even require professional semantic knowledge or ontology background.

This chapter illustrates a number of approaches for semantic search interfaces.
The following aspects are explored to compare these systems and interfaces:

- Query interface
- Query interpretation
- Result interface

3.1 Broccoli: A User Interface for Semantic Full Text Search

This project presents a user interface for semantic full-text search that addresses the problem of user support for complex semantic queries (Bast, Bäurle & Buchhold, 2012). The interface allows users to find information by using semantic queries *incrementally* even if they do not have any knowledge about the underlying ontology. The four main elements of this user interface (Figure 13) are the (1) input field, (2) proposal boxes, (3) query panel and (4) hits area.

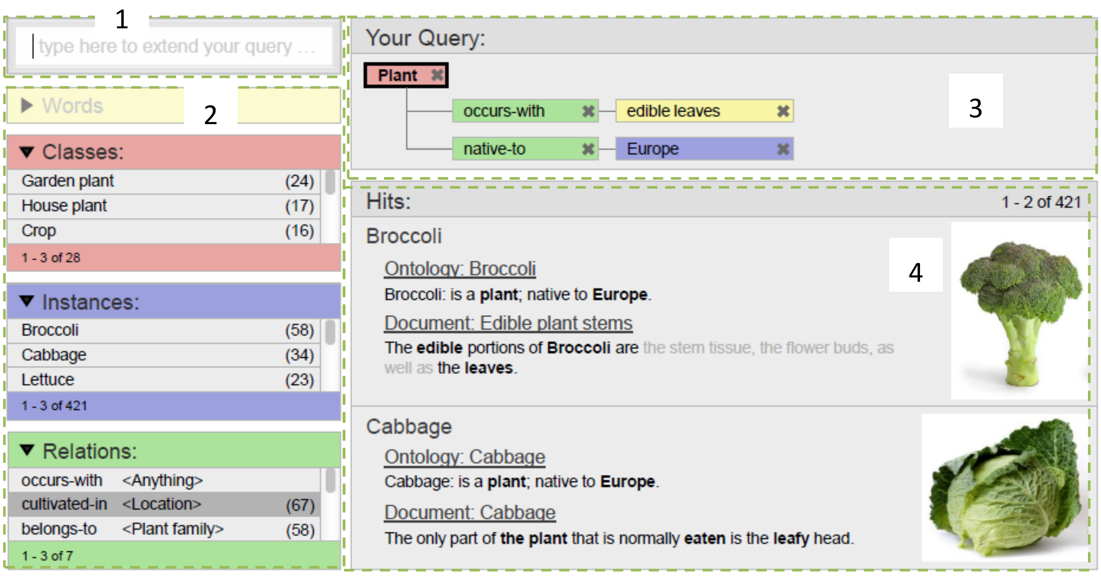


Figure 13: The interface of full-text semantic search for an example query (Bast et al., 2012).

Query Interface

The user types what they want to search in the single *input field* (element 1 in Figure 13, and larger in Figure 14, top). The possible *selection options* for *words*, *classes*, *instances (entity)* and *relations* are shown in the corresponding proposal box (element 2 in Figure 13, and larger in Figure 14 bottom).

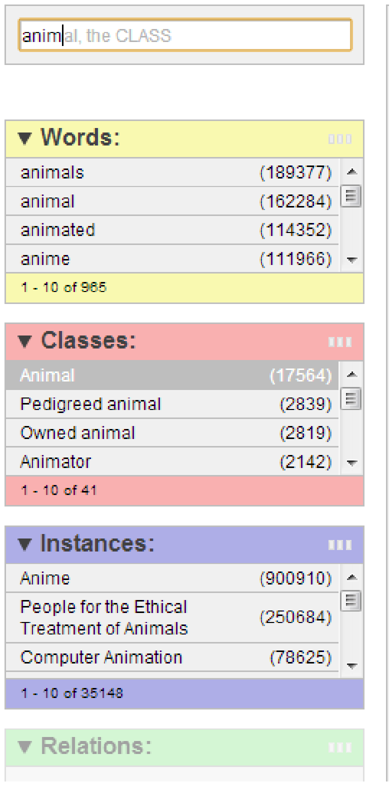


Figure 14: As soon as "anim" is entered, the interface updates by itself and displays appropriate proposals (Bast et al., 2012).

Query interpretation

For example, as the user types "anim", the proposal boxes automatically update their content according to the entered prefix (Figure 14). By clicking proposal 'Animal' in *class* box, user adds it as the root node of query (only class and instance nodes could be defined as root nodes). Similarly, "occurs-with" in the *relation* box and *word* "plant" are added to query as well.

In this way, a search query like a tree is built incrementally in query panel (Figure 15). Single nodes of the tree are the same color as the related proposal box, and they can be easily added, deleted or replaced.

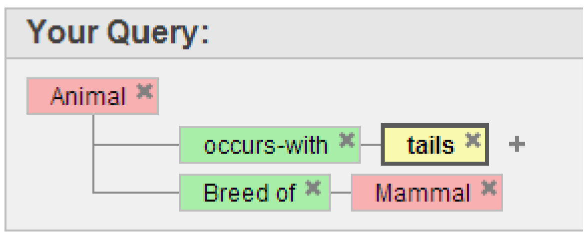


Figure 15: The query panel shows a possible formulation of the example query "Animals with tails and are breed of mammal" (Bast et al., 2012).

Result Interface

The hits for the search query are displayed automatically in the hits area (element 4 in Figure 13, and larger in Figure 16). Hits are referred to the root node of the query tree and grouped by the entity. For example, Figure 16 shows the hits that are grouped by entity "Cat". Every hit has an excerpt as evidence, in which all words that matched the query nodes are highlighted, see Figure 16 (Bast et al., 2012).

Figure 16: One hit group for example query "Animals with tails and are breed of mammal" (Bast et al., 2012).

Analysis

Even though Broccoli supports users in incrementally creating a query, in the end the query that is sent to the system assumes the existence of an ontology. Furthermore, the interface is still quite complex. For unprofessional users, it is complicated for them to formulate the query by using this interface.

3.2 A structured semantic query interface

This project shows a structured semantic query interface to help users to build and submit entailment-based queries to web ontology documents (Koutsomitropoulos, Domenech & Solomou, 2011).

Query Interface

Similar to Broccoli (see Section 3.1), this interface also guides users to construct correct and accurate queries. It does this by separating their intended query into several *atoms* (Figure 17) which will be combined and transformed to expressions that is based on Manchester Syntax.

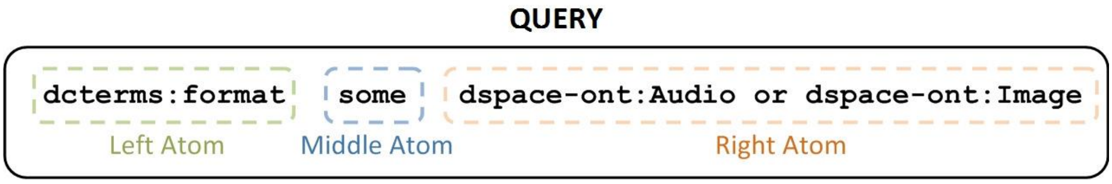


Figure 17: Figure displays three main *atoms* for building Manchester Syntax based query expression (Koutsomitropoulos *et al.,* 2011).

When the semantic search interface is loaded, there are three separate tabs: *Search* (default), *Advanced topics* (inactive) and *Options* (Figure 18).The elements that are required to build a query in Manchester Syntax are all in the *Search* tab. The *Advances topics* tab are not available currently. User can change the ontology and reasoners in the Options tab (Figure 19).

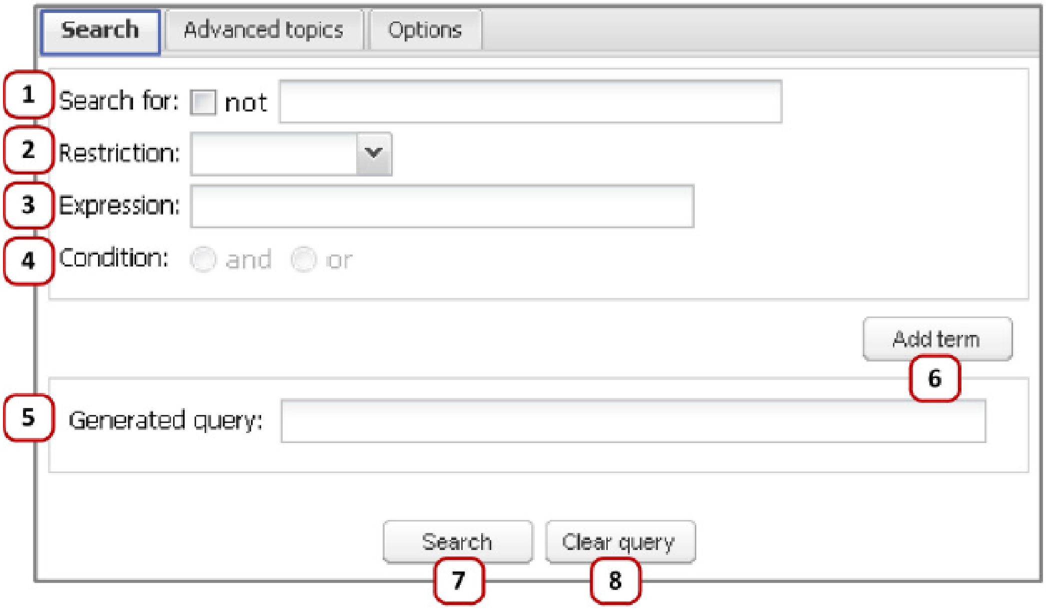


Figure 18: Search tab in semantic search interface (Koutsomitropoulos *et al*., 2011).

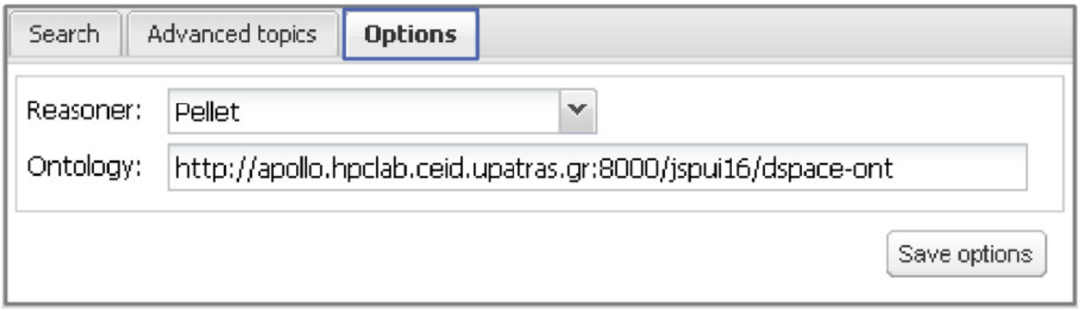


Figure 19: Options tab in semantic search interface (Koutsomitropoulos *et al.*, 2011).

1. **Search for** is related to the outmost first (left) *atom* of a Manchester syntax expression (Figure 17). It provides two groups of suggested values: Types (for classes) and Relations (for properties) (element 1 in Figure 18, larger in Figure 20).Additionally, the fields Restriction and Expression are inactive unless a property is selected.

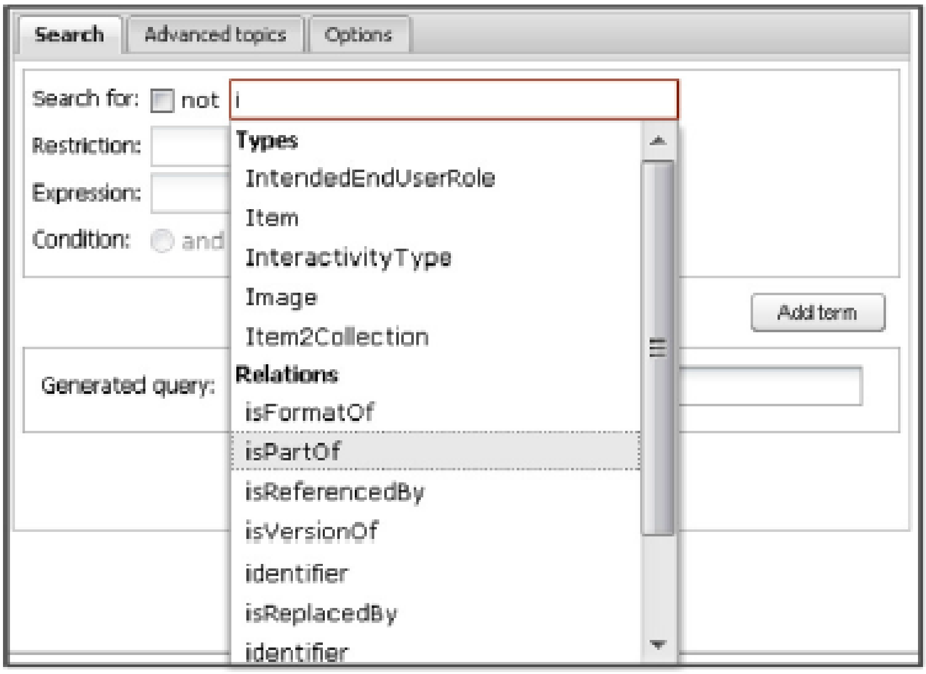


Figure 20:"Search for" can be either a property name or class name (Koutsomitropoulos *et al.,* 2011).

2. **Restrictions** correspond to the middle atom of the expression (element 2 in Figure 18, larger in Figure 21). It contains respective Manchester Syntax keywords.

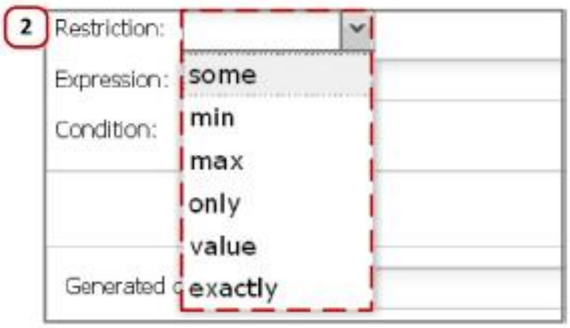


Figure 21: Options for restriction (Koutsomitropoulos *et al*., 2011).

3. **Expression** represents the outmost right (last) atom of query expression (Figure 17). It is

a free-text field (element 3 in Figure 18, larger in Figure 22 top).

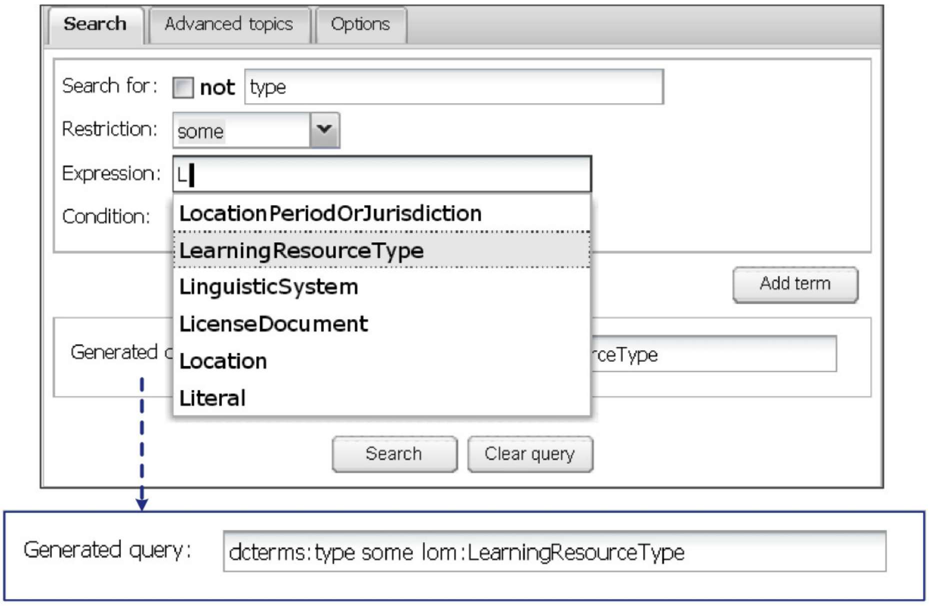


Figure 22: An auto complete facility is provided for class names (Koutsomitropoulos *et al.,* 2011).

4. **Condition** is set for users to express the type of logical connection: and&or (Element 4 in Figure 18).
5. **Generated Query** is the field that displays the query expression incrementally. Users can type directly in this input box (Element 5 in Figure 18, larger in Figure 22 bottom).
6. **Add Term** is the button that adds current completed query expression to Generated Query field (Element 6 in Figure 18).
7. **Search** is the button that starts evaluate the query expression. The result will be showed in the Generated Query field (Element 7 in Figure 18).
8. **Clear query** helps users to clear the search form and to be ready for the next search (Element 9 in Figure 18).

Query Interpretation

For example, the user wants to search all entities that have a type of `LearningResourceType`. As showed in Figure 20, "type" is selected in *Search for* field. In *Restriction* field, the user chooses "some". When user types "L" in *Expression* field, the class names are provided automatically. Finally, the query expression `dcterms:type some lom:LearningResourceType`iscreated.

Similarly, by clicking the "and" in *Condition* field, the more accurate requirement `dcterms:type value lom:Slide is` added to query expression (Figure 23).

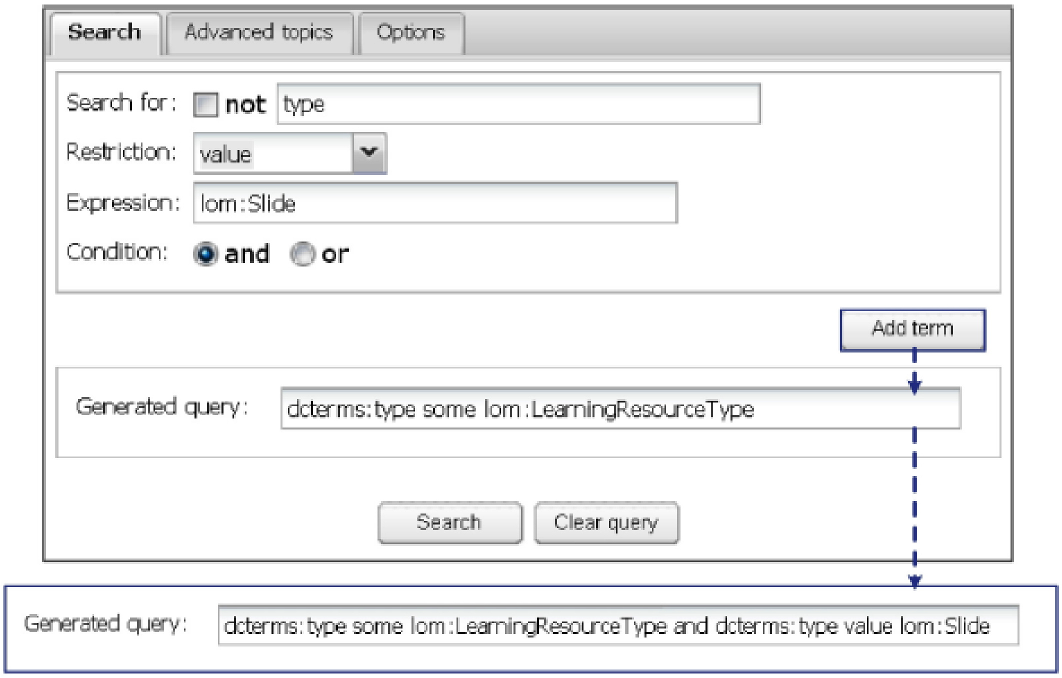


Figure 23: Combing query expressions by using Condition "and" (Koutsomitropoulos *et al.*, 2011).

Result Interface

No information is given about the way that the results are presented in this interface.

Analysis

Similarly to Broccoli, this interface is very complex and requires the users have the background of ontology and semantic languages. This interface would be appropriate for technical experts but not for scholars that are not well experienced with semantic search technology.

3.3 Microsearch: An Interface for Semantic Search

The project proposes a semantic search prototype called Microsearch (Mika, 2008). In order to solve the problems of sparsity and relatively low quality of embedded metadata in semantic search, embedded metadata is suggested to be brought to the surface of the web and be incorporated in the display of result.

Query Interface

The query interface of this semantic search is very simple, just a single input box (Figure 24). For example, "Ivan Herman" is entered in search box.

Query Interpretation

There is no query interpretation explained in the article.

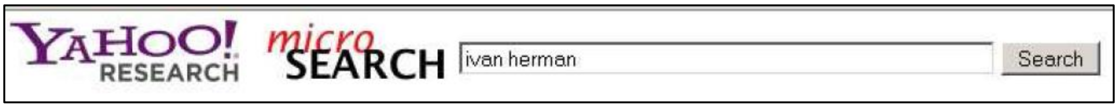


Figure 24: the query interface of Microsearch (Mika, 2008).

Result Interface

In the result page, there is a summary of metadata that is displayed as a part of abstract (Figure 25). Therefore, users can take further actions based on the semantics of information, such as sending email. For example, user are able to email Ivan Herman by the provided email address "mailto:ivan@w3.org". Also, the pages can be related through metadata and be visually grouped together (Figure 26). What's more, Microsearch also demonstrates the promise of semantic search when it comes to the aggregation of information across result pages, such as geographic relevance and timeline (Figure 27).

Figure 25: Metabox shows aggregated metadata (Mika, 2008).

Figure 26: Related pages are based on metadata (Mika, 2008).

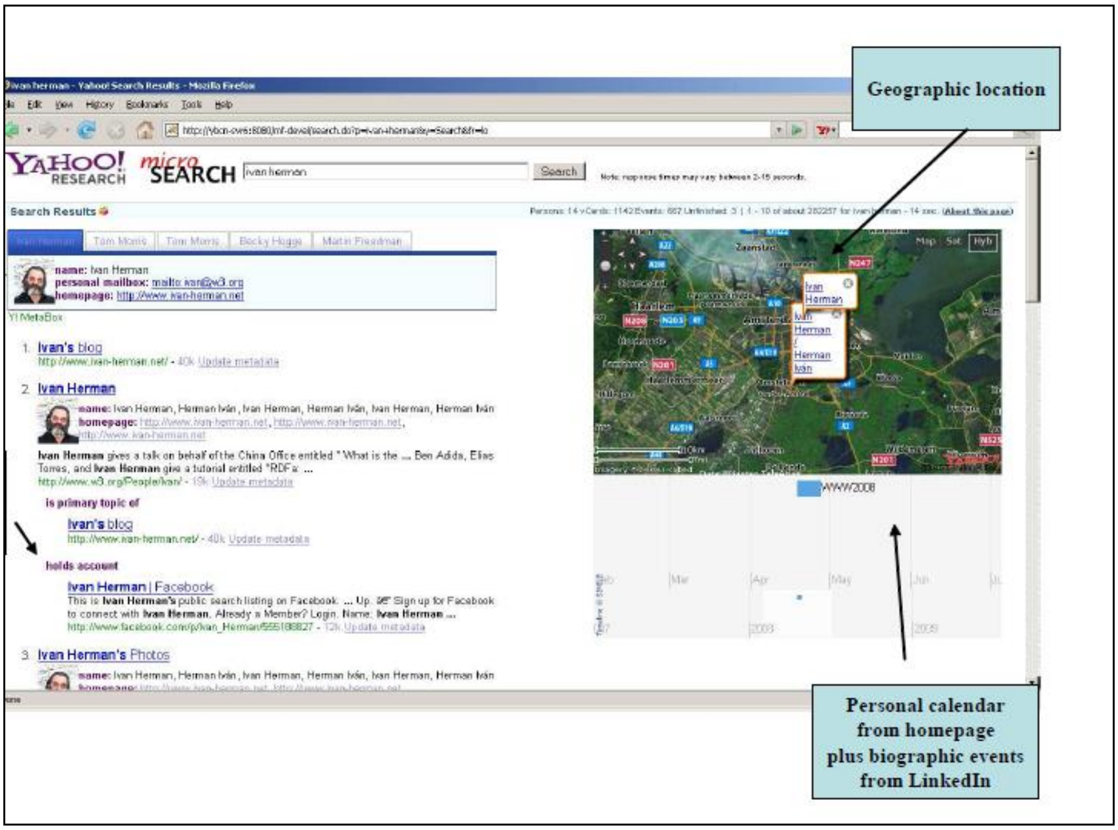


Figure 27: Interface for result display: "Ivan Herman" (Mika, 2008).

Analysis

Different from the last two related works which helps users to build the semantic query, the query interface of this prototype is quite easy to use, in which just types the keyword and search it. However, the result page contains too much information and looks a little messy. Also, it does not provide users a chance to refine their query.

3.4 User interaction with novel web search interfaces

The article compares the performances of three available web search engine interfaces: Carrot2 (C)[5], Middlespot (M)[6] and Nexplore(N)[7] (Ali *et al.,* 2009). In the study, 35 participants are given three navigational search topics and asked to find specific resources by using all three interfaces. The analysis shows that search completion time varies substantially in these interfaces. The mixture of textual and visual information leads to shortest completion time and least number of wrong answers.

5 http://www.carrot2.org

6 http://www.middlespot.com

7 http://www.nexplore.com

Query Interface:

There is no specific introductions for query interfaces of the three interfaces in this article. From the screenshots in the article, the query interface is similar to the regular search engine, such as Google.

Query Interpretation

In this article, it does not present how these three search engines interpret the input queries.

Results Interface

Carrot2 (see Figure 28 for interface) result interface display the search results in traditional method, which contains title, snippet and URL (part 2 in Figure 28). Additionally, it clusters the search result (part 1 in Figure 28), which is displayed in two ways: a hierarchical tree structure (see Figure 29) and the visualization (see Figure 30).

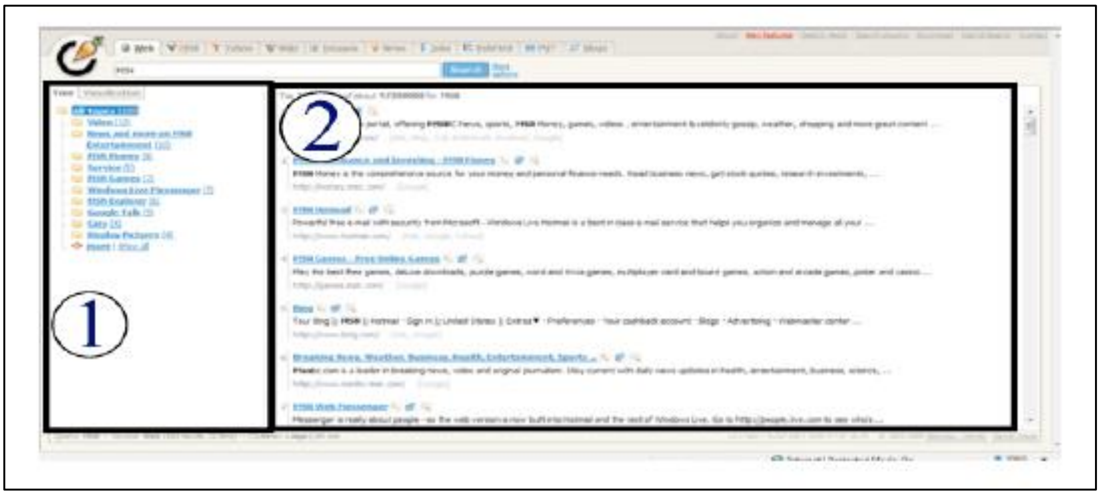


Figure 28: Figure displays the interface for Carrot2 (Ali *et al.,* 2009).

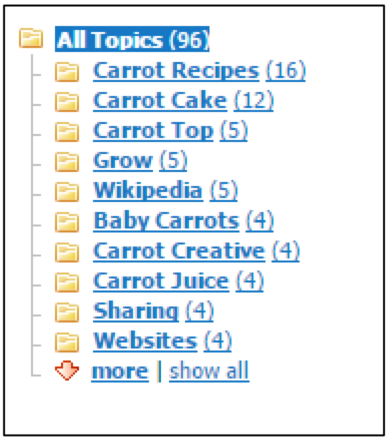


Figure 29: The results of example “carrot” are clustered in tree structure.

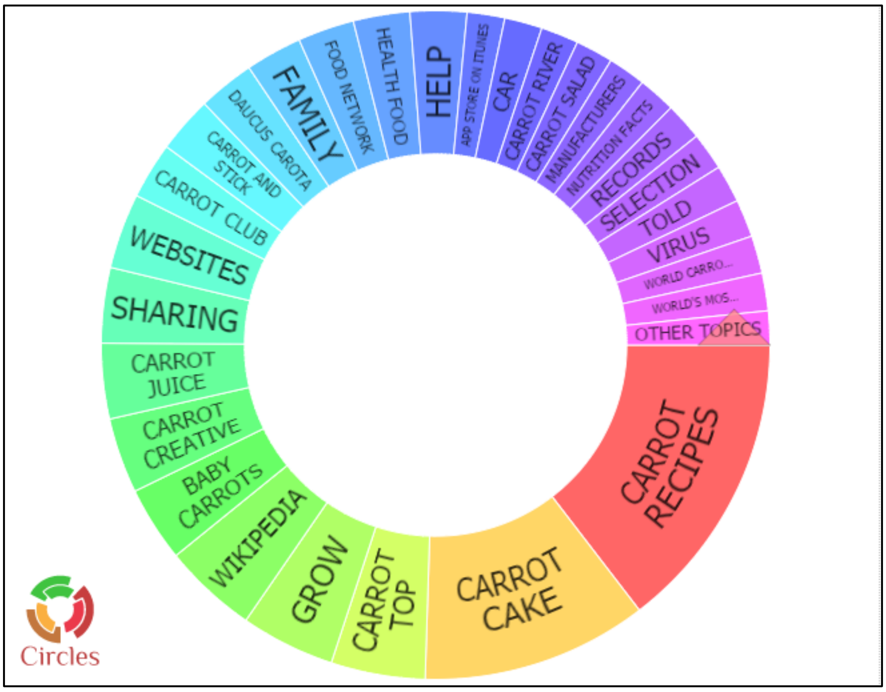


Figure 30: The results of example "carrot" are clustered in visualization.

Middlespot (see Figure 31) result interface contains two areas as well: text area (title, snippet and URL) (part 1 in Figure 31) and visual area (screenshots of web pages) (part 2 in Figure 31). Particularly, the proportion of visual area is about 70%. When the mouse is hover on a specific screenshot, the corresponding image is enlarged (see part 3 in Figure 31).

Figure 31: Interface for Middlespot (Ali *et al.,* 2009).

Nexplore (see Figure 32) result interface is divided into 4 areas: suggested queries, thumbnails of retrieved documents, traditional search results for retrieved documents (title, snippet and URL) and sponsored links.

Figure 32: Interface for Nexplore (Ali *et al.,* 2009).

Analysis

The result interfaces of the three search engines have some commons. Each of the result interface includes the traditional display way (title, snippet and URL). However, they all provide some special features separately. In Carror2, the hierarchical tree structure is very clear for users to select the specific category before browsing all the result documents. In Middlespot, it provides the screenshots for the web pages. The user-driven enlarging screenshot is a very special feature. In Nexplore, it is able to provide users the suggested queries, which is lacked in another two search engines. Also, it contains more result areas than Carrot2 and Middlespot.

3.5 SemSearch: A Search Engine for the SemanticWeb

The search engine SemSearch (Lei, Uren & Motta, 2006) aims to hide the complexity of semantic search from the end user and make it easy to use. It offers user the Google-like query interface which supports the multiple keywords.

Query Interface

The SemSearch query interface contains two parts: subject and keywords. The subject shows the expected type of search results. Operator: is used to capture the subject. *And* and *Or* are used to combine different keywords. Different from some semantic-based keyword search that only accept one keyword, this query interface supports the specification of multi keywords. The query in SemSearch looks like “Subject: keyoword1 or/and keyword2…”. For example, user inputs query “news: PhD students” if he wants to search the news about PhD students.

Query Interpretation

The search engine tries to find semantic entities matches for keyword. For example, user want to search news about PhD students. The result is supposed to be news entity in which PhD students are contained.

Result Interface

The search results are self-explanatory. For example, the related PhD student, the relations between student and news entity must be retrieved (Figure 33, bottom). Also, in each search result, the matched subject and matched keyword are displayed after title (Figure 33, top).

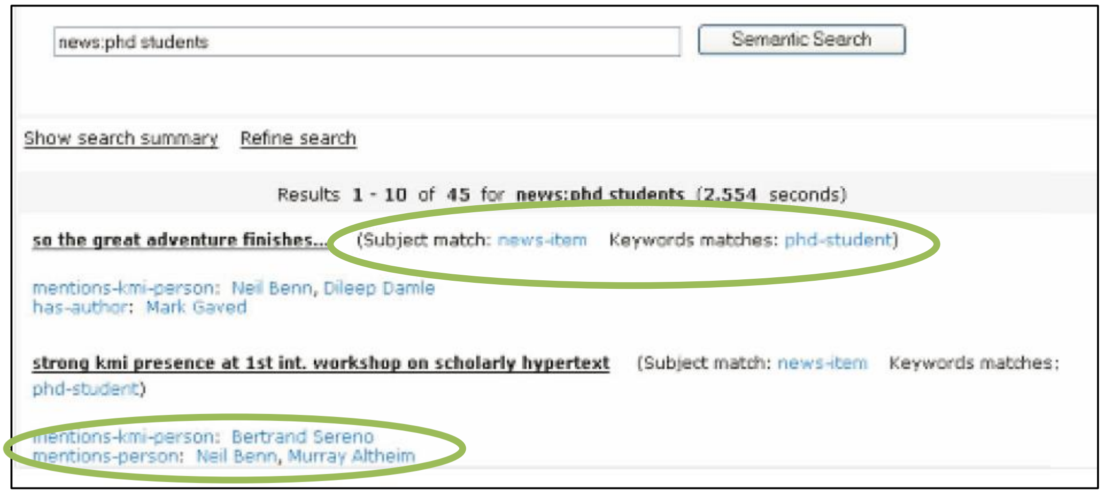


Figure 33: A screenshot of the search results of the query example news:phd students (Lei *et al.,* 2006).

Analysis

Compared with the first two interfaces introduced in this section, this user interface is much easier for ordinary users to use even they do not have any background knowledge. However, the creating of query and the contents in result page are not easy to understand for users who is not familiar with semantics and ontologies.

3.6 A Goal-Oriented Web Browser

The project introduces a web browser that is based on the knowledge base of semantic information, is able to provide users with a goal-oriented web browsing experience. (Faaborg & Lieberman, 2006). The Web Browser contains two parts that work together: Creo and Miro. Creo is a programming by Example system for the web that allows users to train their web browsers to interact with their favorite sites.

Miro is a Data Detector that turns plain text to a form of hypertext.

We now discuss the Miro interface. The interface has a toolbar for the Internet Explorer to match sematic content of a page to a high-level goal. For example. User is viewing the recipe for Blueberry Pudding Cake. The browser notices a pattern of foods on this page and provides user with two options: 1) Order Food (How to create this action is mentioned above) and 2) Nutritional info. If user click on “Order Food”, each food in the recipe will be converted into a hyperlink for that food at the user’s favorite online grocery store. Also, users can browse the nutritional info about each of the foods in their favorite sites.

Query Interface

There is no query interface introduced in this paper. The Miro reads the normal web pages and create a toolbar for it.

Query Interpretation

Creo allows users to train their Web browser to interact with a page by demonstrating how to complete the task. For example, the user click the *Start recording* button, and it starts to capture the user’s action of navigating to FreshDirect.com (Figure 34). Then, the user starts search “Diet Cook”. Clicking on the *Ask->Food brand* link (Figure 35) and clicking on the Scan tab. Use can see the page showed in Figure 36. By checking and un-checking items, users can directly control Creo’s generalizations. Here, user generalizes the concept to “Food Brand”. Finally, the user can finish recording and name this action as “Order Food”. By using the method introduced above, the user’s favorite online store are recorded in example of “Blueberry”.

According to the large knowledge base of ConceptNet and TAP knowledge base, the data detector Miro enables to recognize the useful words in text and convert the text into form hypertext. The hypertext of nutritional information in “Blueberry” example is created from the knowledge base.

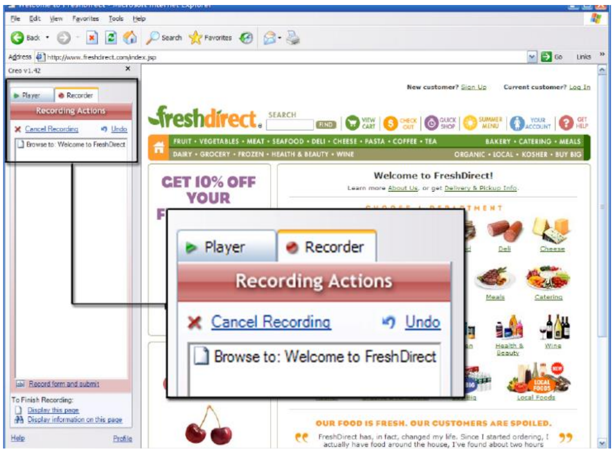


Figure 34: Creo learns how to interact with a Web site by watching the user’s demonstration

(Faaborg *et al.,* 2006).

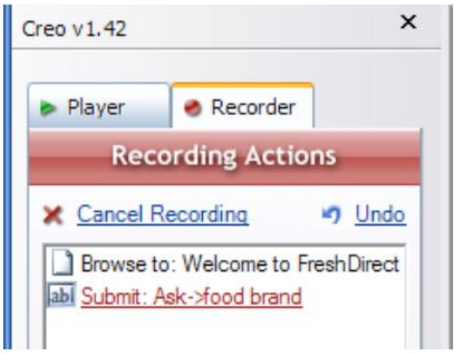


Figure 35: Creo automatically generalizes the user's input (Faaborg *et al.,* 2006).

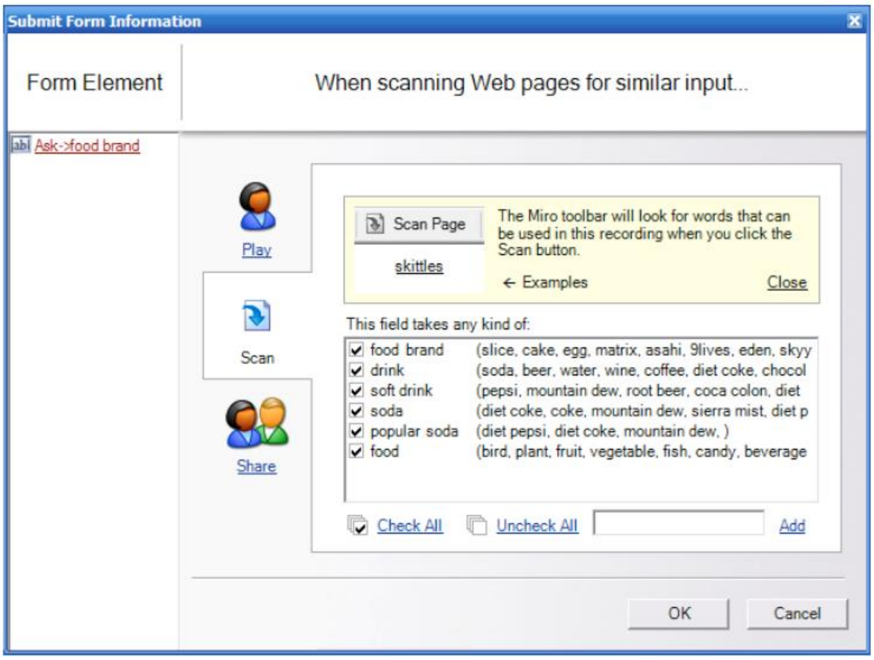


Figure 36: The user can control which generalizations are active with check boxes (Faaborg *et al.*, 2006).

Results Interface

The result interface is displayed as Figure 37 which is similar to the normal web page. The only difference of this web page from normal web page is the auto created toolbar. This toolbar indicates the user's potential search goal.

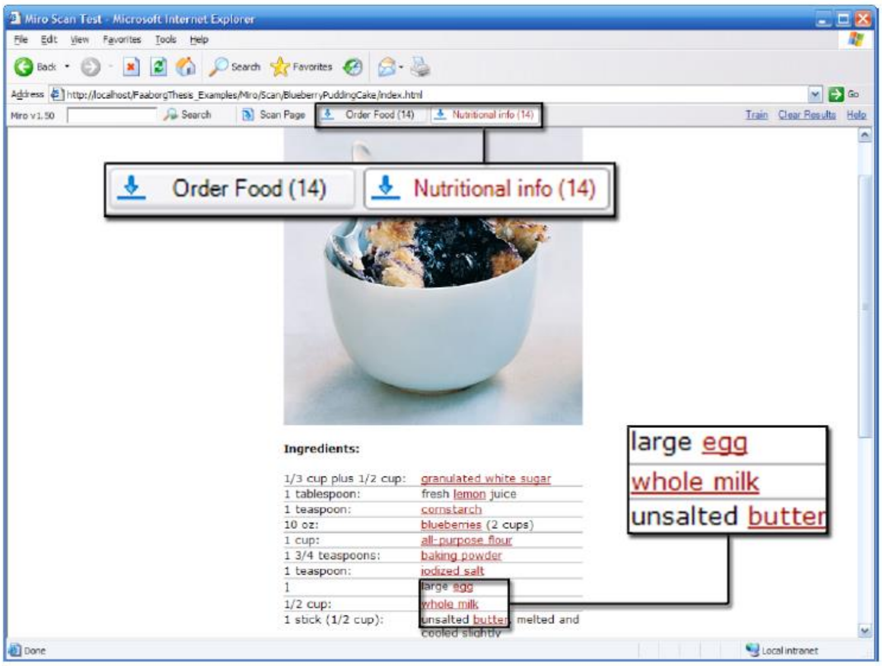


Figure 37: Automatic association of a user's high-level goals with the content of a Web page (Faaborg *et al*., 2006).

Analysis

This is not a keyword or query based search interface as we discussed before. This project detects and analyzes the users' potential intent automatically when user is browsing the web page. Users need to pre-record the actions at first. Also, the process of creating an action is kind of complex for the users.

3.7 Summary

The table below (Table 1) represents the aggregate data among the related works based on the criteria. Most of the systems use an underlying ontology, rather than knowledge base (only Web Browser uses the knowledge base). The user interfaces in related works mostly contains both query interface and result interface.

In most of the query interfaces, they are very complicated to understand and use, even they are claimed to be simple for end-users. Some query interfaces assume that the user is technically experienced and has prior knowledge about semantic web technology (e.g., Structured Interface). The goal-oriented web search browser even needs users to record the actions beforehand. The recording process is complicated as well.

For displaying the results, the interfaces of the result in theses system are mostly similar to the traditional result interface. Therefore, it is simple and clear to read the search results. Although, some systems in related works (e.g., structured interface) are designed for non-professional users and the purpose of them is aimed to reduce the complexity of the semantic search, but the difficulty of building the query still makes the whole searching processing complicated. For the system which does not need users to build a query, the quality of searching result is not very satisfied. The three result interfaces introduced on Section 3.4 brings up some new idea of displaying results.

Related Work	Semantic Search	Use Ontology	Use Knowledge Base	Has Query Interface	Has Result Interface	Designed for Non-IT User	Is Easy use
Broccoli	+	+		+	+	+	
Structured Interface	+	+		+			
Microsearch	+	+			+	+	+
Novel Interface				+	+	+	+
SemSearch	+				+	+	
Web Browser	+	+	+	+	+	+	

Table 1: **+** means the requirement is met, an empty cell means the requirement is not met. The related works listed here corresponds to the works discussed from Section 3.1 to 3.6.

4. Semantic-Enhanced Search & user-driven Disambiguation

Faced with the shortcomings of the two search approaches described in Chapter 2 (keyword based and semantic search), we describe now the concept of semantic-enriched search. This third concept learns from the advantages of both earlier approaches. As in keyword search, the user only needs to input keywords. The keywords are then identified to related concepts by user-guided disambiguation. Then, these concepts are searched by using concept index. According to the matched concepts, the selected documents are displayed to the users. This search approach is very simple for users and augment the efficiency of searching results as well.

This chapter introduces the concept of semantic-enhanced search in Section 4.1. It then introduces the prototype system Capisco in Section 4.2, and the concepts of user-guided disambiguation (Section4.3). Section 4.4 discusses related work for disambiguation and Section 4.5 gives an overview of the concept of user-driven disambiguation.

4.1 Concept of semantic-enhanced search

Like traditional keyword-based search, semantic-enriched search proposed in this thesis uses text-based documents rather than semantic web documents in which data are stored as RDF/XML. Also, the target documents are text-based as well. The main idea of semantic-enhanced search is to analyze the documents for keywords indicating concepts, which are then build into concept index (Hinze *et al*., 2015). Afterwards, the users' keywords are identified into concepts which are used to find matching document by the built concept index (see Figure 38). Thus this approach acknowledges that most documents available online are still lacking semantic markup, and that most users are not familiar with semantic query languages.

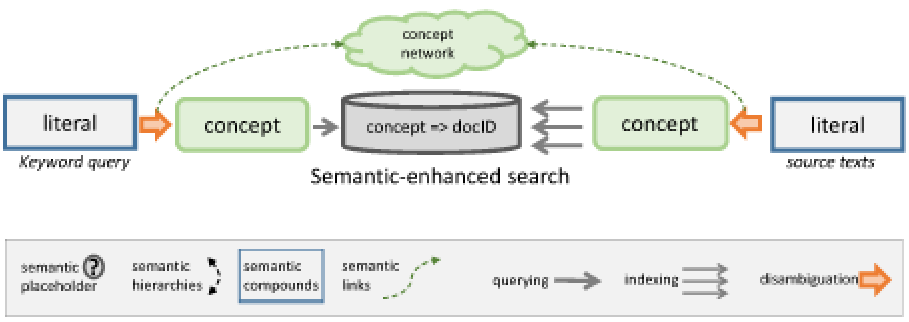


Figure 38: The figure displays the text-based keyword and target document of semantic-enhanced search (Hinze *et al.,* 2015).

The first step of semantic-enriched search is to extract the semantic concepts from each document (see example in Figure 39). Each document refers to a number of concepts, which may be used for the search. To recognize the concepts contained in each document, an ontology or knowledge base, as used in the semantic web, would be employed. Then, instead of the keywords, the search can be performed based on these concepts.

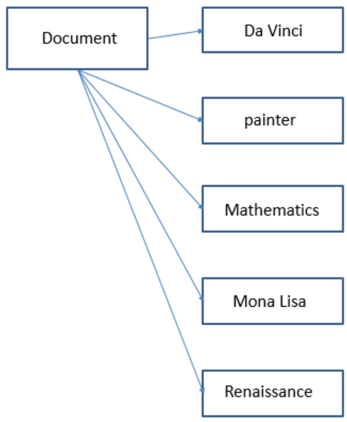


Figure 39: The example of displaying the list of extracted concepts from a document.

Secondly, the concept index is required to be built. A similar way to that which was used to build the keyword-based index (introduced in Section 2.1) can be used here to build a concept index. The concepts that extracted from the document are listed and the pointers are created to point to each of them (see Figure 40). Each concept that occurred in the document is listed in the index with pointers to its occurrence. The concept index contains all the concepts from the documents and also pointers to each of the documents containing a concept.

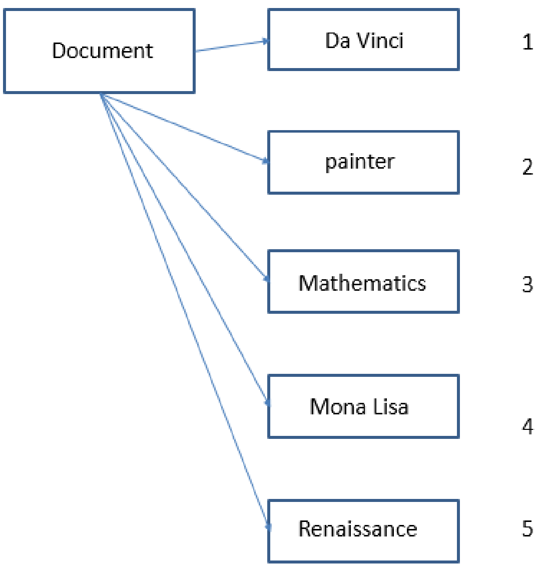


Figure 40: The pointers are created for the concepts in a document.

The last step is deal with the search keyword. According to the input keywords, the corresponding concepts are identified and added to the query (i.e., as query expansion) or used instead of the original keyword-based query (i.e. query replacement). The act of identifying corresponding concepts for each keyword is called *disambiguation* (Zhang, Deng & Li, 2009). Then, these concepts are used to identify the matching documents by using concept index.

4.2 Capisco Prototype System

In this section, we describe the prototype of a semantic-enhanced search system called Capisco. Before describing the system structure, the important component of Capisco Concept-in-Context network are introduced at first (in Section 4.2.1). The structure of the Capisco system are explained then in detail in this Section 4.2.2.

4.2.1. Concept-in-Context network (CiC)

As explained in Section 4.1, the semantic-enriched search systems use an ontology or knowledge base to identify matching concepts. In our semantic-enriched search system Capisco, the semantic concepts and their relationships are identified by the Concept-in-Context (CiC) network, which plays the crucial role in Capisco. In Concept-in-Context network, the initial knowledge base is built by analyzing Wikipedia articles. Wikipedia is the largest and fastest growing encyclopedia currently, which contains two million articles and thousands of contributors. As it is a massive repository for available knowledge, it is considered as an excellent information resources to integrate the Wikipedia knowledge to this system.

Unlike some other systems, i.e. based on the Wikipedia category hierarchy, the approach in this system is to use the concepts extracted from the corresponding Wikipedia articles, and the relationships between symbols and their meanings which are extracted from Wikipedia links. In each article from Wikipedia, the concepts are extracted automatically by the machine which contains the links that the corresponding article is linked to or those are linked to this article. Each article contains several or even substantial numbers of concepts. Some articles may have one or more concepts in common.

For example, as we can see from Figure 41, both of “Dog” and “Cat” contain so many links to different concepts and some concepts are overlapped which means they are related. However, the article contains a lot of irrelevant concepts as well, which indicates that are not related. So, it is obvious that sometimes it is difficult to identify those meaningful links from irrelevant links, which contains many irrelevant results.

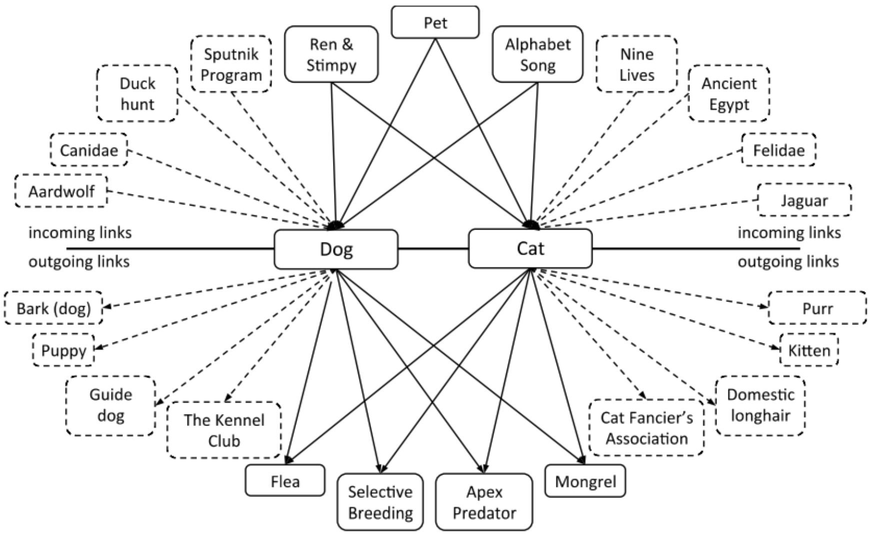


Figure 41: Comparing the article about dog with another article about cat and measure the relatedness between them (Milne & Witten, 2013).

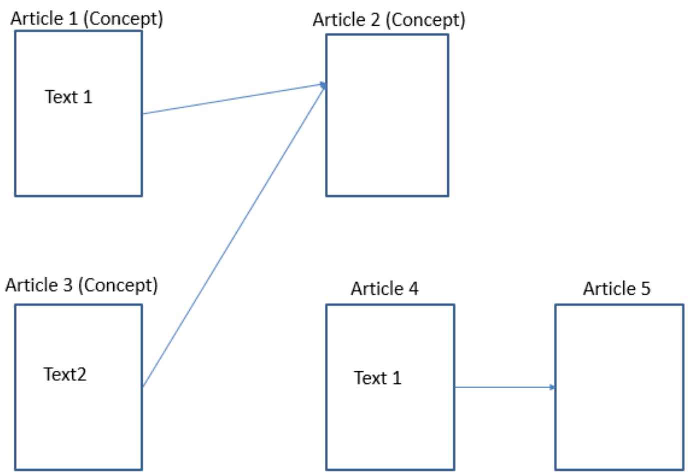


Figure 42: Disambiguation of text and articles.

In different articles, the different texts are likely to link to the same article. However, sometimes the same text in different articles are linked to totally different articles (Figure 42). In this condition, the concepts need be disambiguated. We can use the article the text is belonged to as the context to help fulfill disambiguation and get the disambiguated concept.

4.2.2 Structure of Capisco system

Figure 43 presents the conceptual structure of the Capisco system. The structure contains three main components: 1) the Wikipedia seeding, 2) semantic analysis and indexing, and 3) the semantic-enhanced search. In this section, these three components of the structure are described in detail.

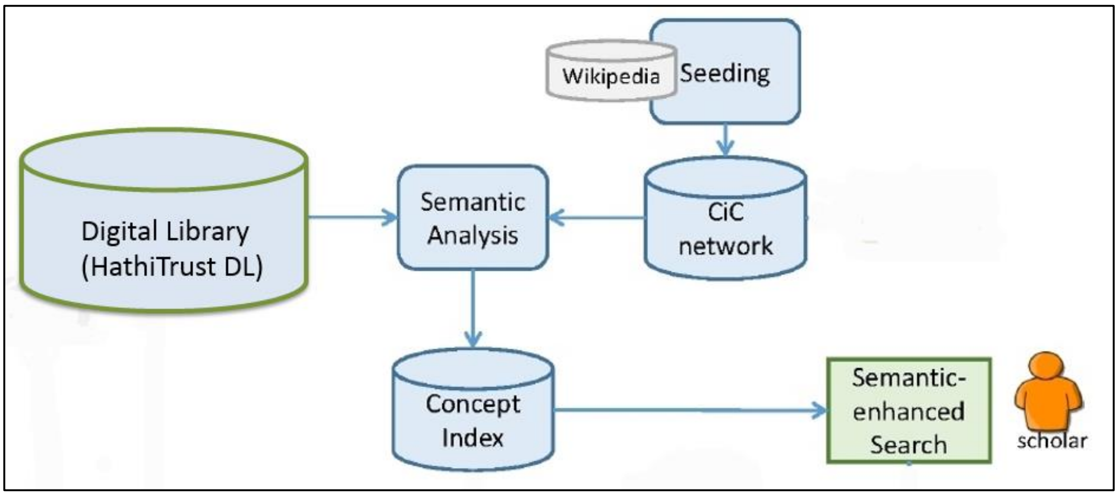


Figure 43: Figure displays the conceptual structure of Capisco system, adapted from (Hinze *et al.,* 2015).

CiC network seeding

As explained above, the relationships between symbols (the textual value assigned to a link included in a Wikipedia article) and their meanings are extracted from the links between Wikipedia articles (see Figure 44). In each Wikipedia article, it has a unique ID. Every ID points to the title of the article, which is regarded in Capisco as a concept. In regards to the links between the articles, the starting side for the link is IDC (id for the context article, which means the article uses this article) and the destination side is IDM (id for the meaning article, which means the article uses the concept as name. Thus, the link is considered as the triple of symbol, IDC and IDM. IDC is the start point of the link. IDM is the destination of the link and symbol is the textual value used for the link.

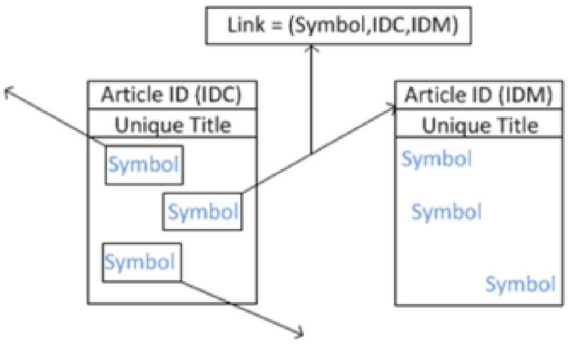


Figure 44: The relationships between IDC, IDM and symbol.

Semantic analysis and indexing

The concepts derived from CiC network are used to do the semantic analysis for the documents in the digital library. In the given text, the terms that appear together are considered as related, so one term is able to be disambiguated by their related terms. In reality, these terms are might be ambiguous themselves, which refer to many concepts in CiC network. However, some of them might be unambiguous in the context when they only have one meaning in the given context. Therefore, these unambiguous terms are treated as anchor pointers for disambiguation, by which the documents are semantically analyzed. The concept index is built by the semantic analysis, which handling the links between the documents in the digital library and the concepts between the CiC network.

Semantic-enhanced search

With the help of the Capisco, the text-based search interface is provided to the users. According to the terms that entered by the users, the mapping concepts are automatically listed to the users, which leads to the user-guided disambiguation (see Section 4.5). Based on the selection result from the users, the matching documents are returned to the users.

4.2.3 System access and functionality

In order to access to the Capisco system and manipulate the search process, there are several context commands of manipulating data in Capisco are required to be introduced.

The context command (Context (symbol) ->IDC) will return all IDCs that a symbol is used in article (see Figure 45). For example, the symbol 'english' is used in the articles 'Language' and 'United Kingdom', so the context command would return the IDs for at least these two articles. This command can be used to learn in which context a symbol is used.

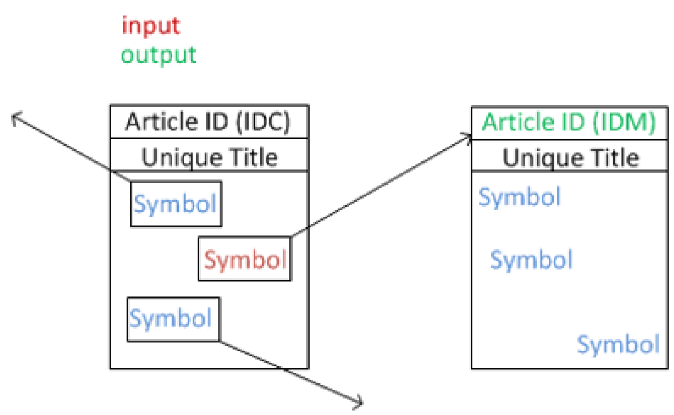


Figure 45: The structure of Contexts Command.

Similarly, the context command (Senses (symbol) ->IDM) will return all IDMs that a symbol is linking to. For example, the symbol 'cancer' might return the IDMs for the animal cancer, the constellation cancer, the astrological sign cancer and so on. This command can be used to learn what different meanings a word can have.

According to the given meanings of the word, the context command (Artid(name) -> ID) will return the ID for a given article title. On the contrary, the context command (Artname(id) ->name) will return the title for a given article ID.

4.2.4 Search Processing

From the CiC network, all potential meanings of the keyword can be retrieved. In this section, an example is used to demonstrate how the Capisco system support the semantic search. We use "Kiwi" as the given keyword for the example.

The first step is to extract the concept of keywords that would be utilized as concept index (symbol). Each term is likely be related to multiple concepts (polysemy). From the CiC network, all the relevant concepts of keyword can be identified. Thus, the keyword "Kiwi" is found to be used to refer to Kiwi people (New Zealander), Kiwi bird (New Zealand national bird), Kiwi fruit and another 11 concepts (Figure 46).

```
0.3389830508474576,Kiwi
0.23163841807909605,Kiwi (people)
0.10734463276836158,New Zealand national rugby league team
0.0847457627118644,2008 Rugby League World Cup squads
0.05649717514124294,Kiwifruit
0.04519774011299435,New Zealand
0.03389830508474576,Kiwi (shoe polish)
0.02824858757062147,Kiwi (horse)
0.02824858757062147,Kiwi (store)
0.01694915254237288,List of Code Lyoko characters
0.011299435028248588,Kiwi (band)
0.005649717514124294,Culture of New Zealand
0.005649717514124294,New Zealanders
0.005649717514124294,Cui (Dragon Ball)
```

Figure 46: The keyword "Kiwi contains 14 concepts by the system search results".

We face another problem for disambiguation, which is synonymy, in which a concept is known by multiple names. For example, the word kiwi (bird) is referred to by words such as "Kiwi", "kiwi", "kiwi bird" or "Kiwi bird" (Figure 47). Thus, no matter "kiwi bird" or "Kiwi bird" should be linked to the same article.

kiwi
Kiwi
Apterygidae
Apteryx
Kiwis
kiwis
Apterygiformes
kiwi bird
national bird
Brown kiwi
Kiwi bird
Kiwi Bird
North Island brown kiwi
apteryx
bird of that name
brown kiwi
native bird of that name
Brown Kiwi
New Zealand national rugby league team
flightless bird
Kiwi (Bird)
Kiwi (bird)
Liittle Spotted Kiwi

Figure 47: The words that the concept “kiwi (bird)” are referred.

Although the Capisco CiC allows the lookup of concepts for a keyword, the actual intent of the user is still ambiguous at this stage. Therefore, a further refinement from user is necessary. With participation from users, we can move from the possible concepts to the intended ones. The next subsection explores the requirements of the process.

4.3 Requirements on user-guided disambiguation

From the example above (Section 4.2.4), it is obvious that the same symbol might refer to different concepts (Kiwi refers to kiwi bird, New Zealander and so on). In many cases, the symbol refers to a number of concepts. In this condition, the way to distinguish these different concepts is to refine the user’s information need. The user-guided disambiguation is an effective way to understand the user’s intent and provide users with more accurate search results.

In semantic search, the input query terms need to follow a strongly defined semantic query language such as SPAROL. Also the documents are stored as semantic web data, such as RDF. Thus, the semantic search engine are able to supply accurate search results by analyzing the semantic query language. However, in the Capisco, the semantic-enhanced search deals with the given keyword. In order to get accurate concepts of the keyword, the further disambiguation from the users is required to identify the intent meaning of keyword.

AS explained in Section 4.2, the Capisco system is able to identify the related concepts for a given input keyword. These possible meanings for a keyword are displayed to a user to support the manual semantic disambiguation. The user-driven disambiguation helps constrain the possibilities of the target concepts and augment the efficiency of semantic-enhanced search. The user-guided disambiguation is described in detail in Section 4.5 after first discussing related approaches to disambiguation.

4.4 Related work on disambiguation

There are several approaches to disambiguation. In this section, we introduce and analyze the disambiguation approaches in the following related works according to our requirements. Our analysis concentrates on the process, the performance and any involvement of users in the approach.

4.4.1 Concept based query expansion using WordNet

Concept based query expansion using WordNet is a query expansion technique (Zhang *et al.,* 2009). It utilizes WordNet, which is a large manually-created thesaurus, and its semantic relatedness module. Zhang *et al*. observe that in a query sentence, the meaning of each word is not only determined by its own definition, but also affected by the context of the whole query sentence (based on the work of Hoeber & Xue, (2007)). They pre-process sentence queries as to split those sentences with punctuation and other elements, into separate queries (so-called “clean” sentences).

Disambiguation Process

They process each of the queries through the WordNet::SenceRelate module. For each term in the query, all synonymous concepts are identified in WordNet, and stored into a synset. All synsets related to a query are combined into a 2-dimenional concept array. This array is then used for query expansion in the following way: New queries are formed by selecting a synonymous term for each of the original terms (i.e., a query is a combination of terms in which each terms is from a different synset). Instead of using all permutations of terms, suitable queries are selected using the following process: For each of the newly created candidate queries, another process of synonym identification is started and another concept array is identified. Then the original concept array and the newly created concept arrays are compared. Only candidate queries that have the same concept array as the original query are accepted as valid query expansions.

Disambiguation Performance

Zhang *et al*. conducted experiments in which the top 10 results of the original queries were compared with those of their valid query expansion queries (using Google). Shorter queries were excluded from the experiment as it was found that this query expansion does not work on short queries. The authors report an improved query precision of 7% between expanded and original queries, judged by human participants. The authors note the slow performance of the disambiguation process.

User Disambiguation

There is no user disambiguation introduced in this article.

Analysis

Zhang *et al*. follow a similar approach to our project in identifying synonyms from a given knowledge base (WordNet). Their approach then uses an automatic second step to identify appropriate synonyms. Their approach does not directly involve users and their information need. Furthermore, as they observed, their method is not appropriate for short queries (no information about the number of query terms was given). It also appears as if Zhang *et al*. assumed that queries would be sentences, not collections of keywords; however, the implications of this assumption are unclear. The results achieved seem poor for an approach that has slow performance.

4.4.2 SemSearch

SemSearch is a search engine designed by Lei, Uren and Motta (Lei *et al*., 2006) which aims to decrease the complexity of semantic search and enable the ordinary users without knowledge of ontology can manipulate it as well. Its user query contains two part: the queried subject which indicates the type of intent results and the combination of keywords.

Disambiguation Processing

In the process of searching, the first step in SemSearchis is to figure out the semantic meaning of the user queries. The Text Search Layer in the engine finds out the explicit semantic meaning of the keywords. In this way, in order to find out all the matched semantic entities of each keyword, Lei *et al* use labels of semantic entities as the main search source, which represents the understandable meaning of the semantic entities. All the semantic entities in the data repository are indexed firstly, and then the engine searches the indexed repository and g*et al*l the matched entities. These matched semantic entities are the potential meaningful candidates for keywords.

Next, the engine translates user queries into formal queries by classifying entered queries into two types: simple queries (contains two keywords) and complex queries (contains more than two keywords). In simple queries, Lei *et al*. build a set of templates to help retrieve the relations between two entities of semantic entity combination. Also, the relation between entities are needed for searching result. According to the corresponding combination of matched entities, the template is to be initialized as the formal query. The keyword is likely to match more than one semantic entity, thus there are more than one query needs to be constructed. Specifically, if the subject keyword matches N semantic entities and the other keyword has K matches, therefore totally there are N*K queries that need to be constructed. For complex queries, the semantic entities of each matched keyword are needed to be combined together and then construct queries for each of the combinations. In some cases, the

Disambiguation Performance

Lei *et al* note that, in reality, the number of the combination for complex query could be huge and takes quite long time to get the result. Hence, Lei *et al* just use the simple query to find the matches for keyword in order to get quick result.

User Disambiguation

There is no user disambiguation involved in this article.

Analysis

It is not clear how long it would take to get the results for complex quires if a large number of of queries need to be constructed. Also, there is no experiment section in this article, so we are not sure of the performance of this engine in terms of simple quires and complex queries. Moreover, there is no user participant involved.

4.4.3 Librarian Agent Query Refinement

Librarian Agent Query Refinement (Stojanovic, Studer & Stojanovic, 2004) is a comprehensive approach for ontology-based queries refinement. It adjusts a query to the demands of the user step by step and includes the interaction with the users as well by analyzing the user behavior. The list of results by using this approach is able to decrease the irrelevant results and increase the relevance of the results.

Disambiguation Process

At first, the potential ambiguities of the query are discovered. The user's information needs and their queries are always not matched, which is defined as the query ambiguity. The ambiguity of the query is defined to two types: the semantic ambiguity and the content-related ambiguity. In terms of semantic ambiguity, the specific formula is applied to calculate the query and output the query variable that causes the highest ambiguity in the query. This discovered variable will be refined in the next step pf progress. In content-related ambiguity, the results of the query can be used to discovery the potential ambiguity. For example, if two queries return two same results, these two queries are defined as equivalent queries. Therefore, a list of equivalent queries are treated as the indicators of content-related ambiguity. In this way, the content-related ambiguity can be presented by comparing the results of the queries.

Then, the meaning of the query needs to be found out according to users' needs. Stojanovic *et al*. use an approach called implicit relevance feedback to get the user's preferences. Specifically, the resource that the user selects from the retrieved results is treated as the correspondence of the user information need. Consequently, more relevant information are considered as the intent of current query. According to the given users' information need, the ambiguities collected from last step can be interpreted.

Lastly, regarding the relevance for fulfilling user's information need, the recommendations of query refinements are ranked to the users.

Disambiguation Performance

There is no user experiments introduced in this article, therefore, feedback on this query refinement is not available.

User Disambiguation

The query refinement method introduced in this article involves the user participation for disambiguation (see step 2 in *disambiguation process*). The users are able to select a resource from the retrieved results. The resource that are selected by the user, to some extent, refers to the users' intent. By analyzing the resources that the user pick up. By analyzing the common attributes in the results which are selected by the users to view, the intent of the user are discovered.

Analysis

Stojanovic *et al*. (Stojanovic *et al.,* 2004) do not include the experiment section in the article, so the actual performance of this new approach in query refinement is not clear so far. Also, there is no proper result lists displayed in the article, so we do not know what it looks like.

4.4.4 Summary of related work on disambiguation

The disambiguation approaches introduced in this section mainly use the query expansion, combination of semantic entities and query refinement respectively, to disambiguate the input user quires. In these three disambiguation approaches, the user queries are all translated into specific generated quires. By analyzing these constructed quires, the more accurate results/concepts are identified from the candidates. However, the performance of these disambiguation approaches are not very impressive and satisfied. Concept based query expansion only improves 7% of accuracy and the performances of another two disambiguation are not clear from the given articles.

Only in the Librarian Agent Query Refinement, the user refinement are involved to do the disambiguation. The users are provided with a list of refinements. According to the users' preference, the ambiguities of queries are interpreted in a further step.

4.5 Concept of user-driven Disambiguation

The related work introduced in last section identifies one approach for refinement of user queries with the help of user feedback. As explained in Section 4.3, automatic disambiguation in Capisco is not suitable to be applied for keyword-based search as there is no context available. We suggest instead to use user-guided manual disambiguation. In this section, we explain our user-guided disambiguation approach in detail.

The principle of user-guided disambiguation is to make use of a searcher's information need as the source to disambiguate the existing relevant concepts of the given keyword. We explained before how the concepts of a keyword are identified by the Capisco system's CiC network. However, it is rare that the keyword belongs to only one concept. A keyword potentially refers to a number of concepts (that is, it is ambiguous). Thus, the user-driven disambiguation is the process to find out the correct concept according to user's preference of the concepts and information need.
Using Capisco, a user could use the keyword "Kiwi" in a search. The information about possible senses can be retrieved from the CiC network in Capisco. This requires to lookup all senses for term "Kiwi" (using the commend "Senses Kiwi", as shown in Figure 48). This then retrieves the *ids* of the senses (Figure 49). The command "*artname* artid" (eg: artname 509080) then retrieves the name of sense one by one.

Figure 48: The syntax of typing keyword in Capisco system.

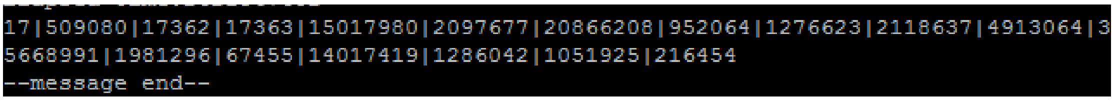


Figure 49: The results of the optional senses in Capisco system.

To support this process, we design the user interface for the Capisco system. With the help of our search interface, the users without the semantic web or concept are allowed to manipulate the system visually. The searching steps are simplified and clear. The users type the keyword in the box directly, like any other search engine. The names of all the optional senses are displayed to the users, rather than the ID numbers.

Obviously, both automatic semantic disambiguation and user-driven disambiguation is complicated to manipulate by using the text commands, especially for the scholars that do not have professional background. Therefore, a simple and user-friendly interface for this system is necessary.

4.6 Summary

In this section, we present the concept of semantic-enhanced search which take advantages from traditional text-based search and semantic search. The Capisco system and its structure are explained in detail. According to the introduced functionality and search processing, the users who have Linux knowledge are able to do semantic-enhanced search effectively by using Capisco. However, our aim is to enable normal users without specific technologies to manipulate the semantic-enriched search. Thus, the user-friendly interface is needed to be designed for the end-users.

5 User-guided disambiguation: interfaces

In this chapter, we present the user-guided disambiguation strategy through three interface prototypes that we designed. The next section describes the general layout of the user interface in with the help of screenshots. In Section 5.2, the differences of three variations for user-disambiguation interface are introduced. The example of manipulating the search by using the interface is demonstrated in Section 5.3.

5.1 Interface for user-guided disambiguation

The users of the semantic-enhanced search are acknowledged as scholars that are not technically versed rather than experts who have technical knowledge of semantic web technology. Thus, one task is to create the user-friendly interface that the users are familiar which easy for them understand and use. Also, in order to increase the efficiency and accuracy of searching, apart from the system disambiguation, the interface of user-driven disambiguation is crucial as well.

Within the Capisco architecture, our disambiguation interface is part of the semantic-enhanced search component. The user interface and its interactions support users in disambiguating the search keyword step by step. In the end, the matched documents are displayed to the users according to the disambiguated keyword.

The three disambiguation interfaces offered to the users follow the same principle layout and the differences of the three variations are explained in detail later. Here, we introduce the general layout at first. Our semantic-enhanced search interface contains two main part: keyword inputting interface and the user-guided disambiguation interface.

Figure 50 and Figure 51 show the overview of the layout of the user interface. As the Capisco project was executed in collaboration with the HathiTrust, the interface uses the Hathitrust logo and name. The interface contains the following 4 main elements:

1) Search box (number 1 in Figure 50): the input box for the keyword. The users type the keyword that they want to search here. Then click the search icon, the system starts searching. This is query interface for the semantic-enhanced search.
2) Senses result area (number 2 in Figure 51): the list of all the optional senses. All the possible senses are displayed in this area. Users can go to next page and view other options by clicking the pager (number 3 in Figure 51). This is area where user-guided

disambiguation occurs.

3) Picked up senses box (number 4 in Figure 51): the selected senses will be displayed in the box here. By selecting the optional senses displayed above (number 2 in Figure 51), the selected senses will be displayed in this box. Then, the user can get the documents of the selected senses by clicking the icon. Element 2 and element 3 are the user-guided disambiguation step.

4) New search (number 6 in Figure 51): Start a new search. User gest back to the search interface (Figure 50) by clicking the "New Search".

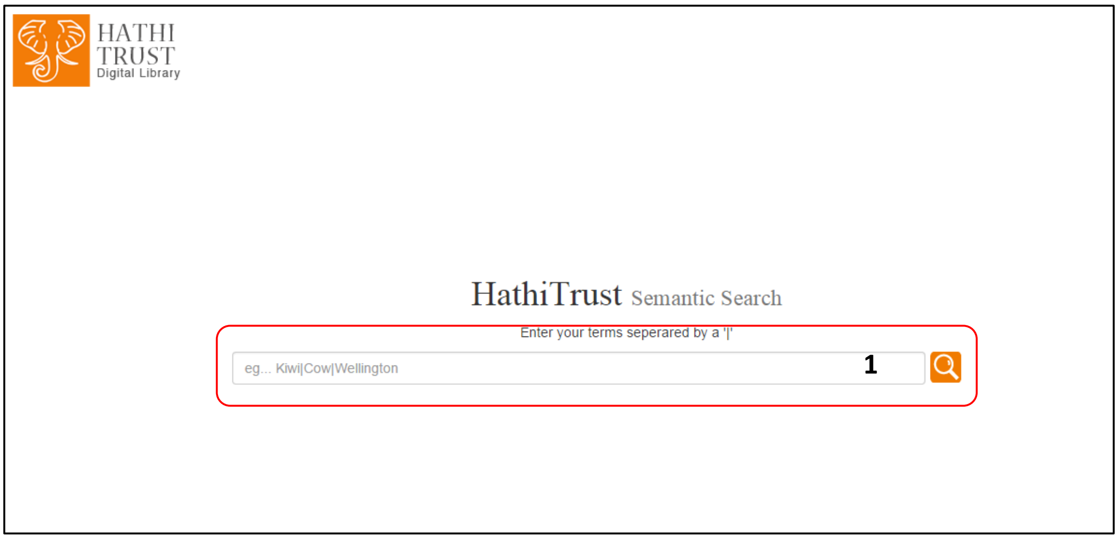


Figure 50: The search interface for the Capisco system: number one shows the search box.

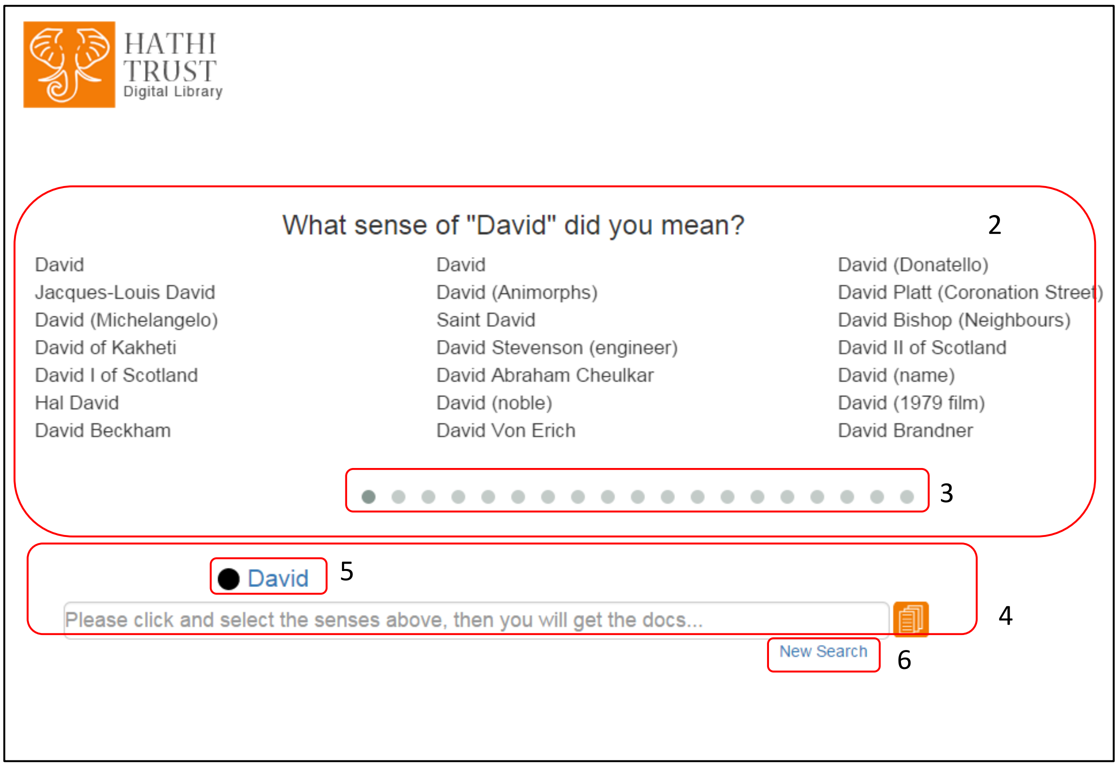


Figure 51: The interface for the result page of optional senses: 2 is senses result area; 3 is the pager of senses result area; 4 is the picked up senses box; 5 is the new search link.

5.2 Three disambiguation interfaces

For the user interface, it is not only the design of the query interface is important, but also the interface for user-disambiguation is crucial as well. In our user interface, we design three different disambiguation interfaces for the users: 1) interface sorts the senses of keyword according to the alphabet; 2) the interface ranks the senses of keywords by the decreasing the amount of the corresponding documents; 3) the interface orders the senses of the keywords by their likelihood. These three disambiguation interfaces are designed to investigate how they influence users' searching strategies.

The example of the given keyword "David" is used here to display the differences of three disambiguation interfaces. Regarding to the disambiguated concepts from Capisco, the first result page of all the optional concepts in three disambiguation interfaces are displayed separately in Figure 52, Figure 53 and Figure 54 below. It is obvious that the optional senses (concepts) that shows in first page vary dramatically according to the different ranking methods. We will do the user study for all the three interfaces and compare the feedbacks of them in the next chapter.

Figure 52: The results of the interface that sorts the senses of keyword according to the

alphabet.

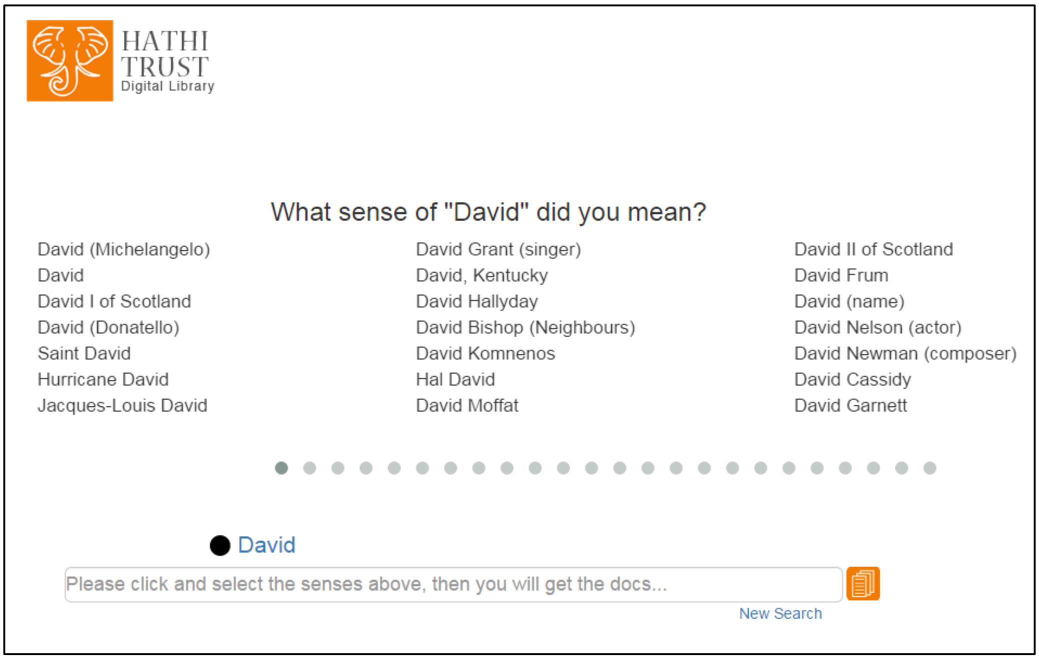


Figure 53: The results of the interface that sorts the senses of keywords by the decreasing the amount of the corresponding documents.

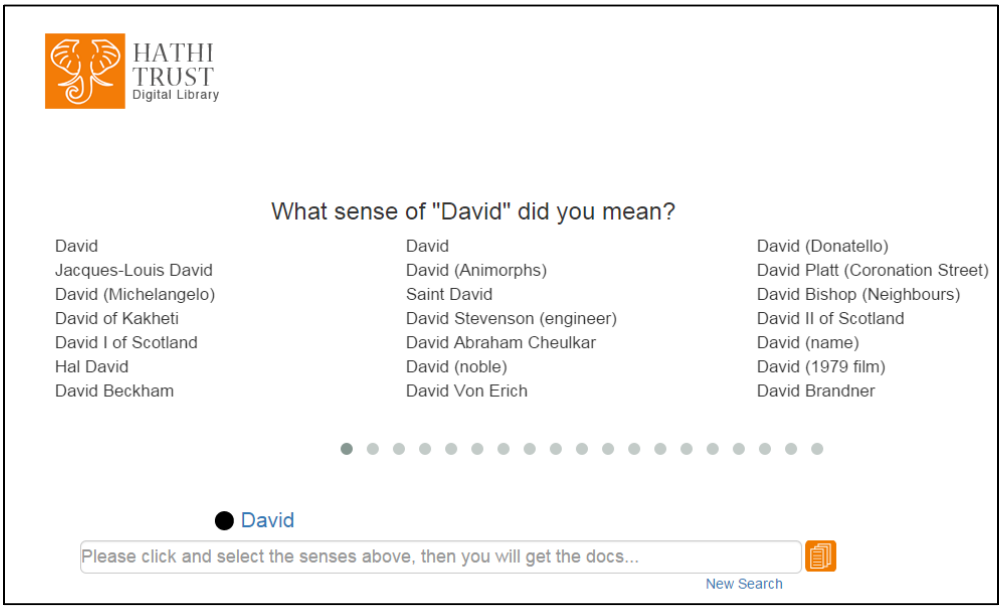


Figure 54: The results of the interface that sorts orders the senses of the keywords by their likelihood.

5.3 A disambiguation example

The previous sections we explain how the user interface work and introduces three disambiguation interfaces. In this section, we will demonstrate how to use this interface to do the search by using the example. In this example, we use the interface that sorts the senses of keyword by alphabet.

We want to search the information about the famous movie “Harry Potter. We type the keyword “Harry” into the search box and click search icon (Figure 55). Then, the result page of senses are displayed in the screen (Figure 56). In the first result, we do not find sense that are related to the “Harry Potter”, so we click the pager and continue looking the senses in the next page (Figure 57).By keeping doing this, we find the sense “Harry Potter” which is possibly the target sense in the fourth page (Figure 58).

Unlike some of the complicated semantic interfaces we described in Chapter 3, the search steps in this interface are simple to use and easy to understand. Even for the unprofessional users, it is quick for them to learn how to use this system. The users’ performances using the three disambiguation interfaces are introduces and analyzed in detail in the next chapter.

Figure 55: The example of typing the keyword “Harry” in the search box.

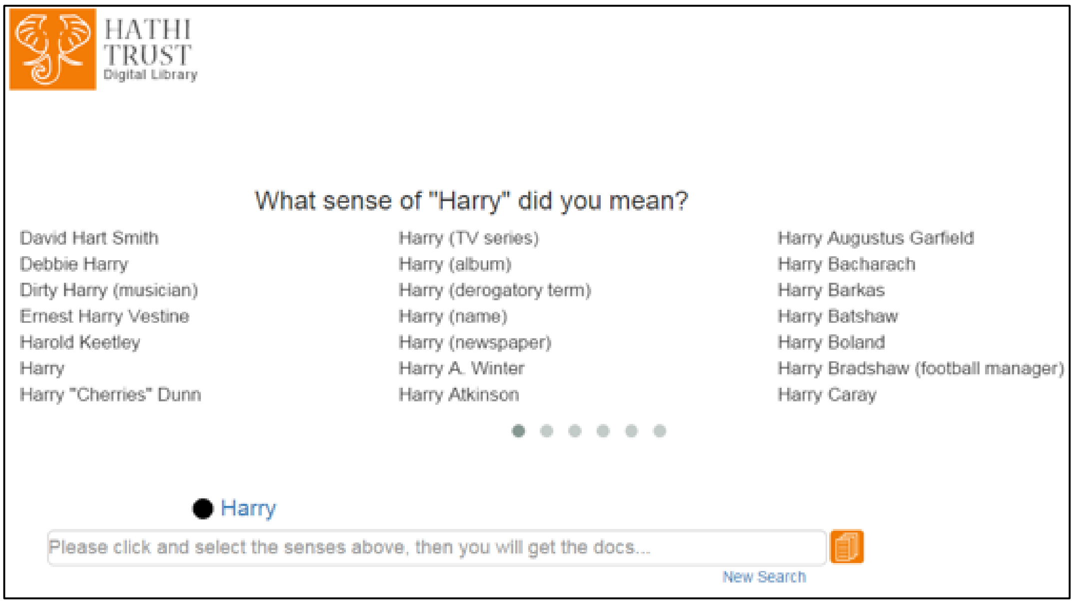


Figure 56: The Figure displays the first result page of the example keyword “Harry”.

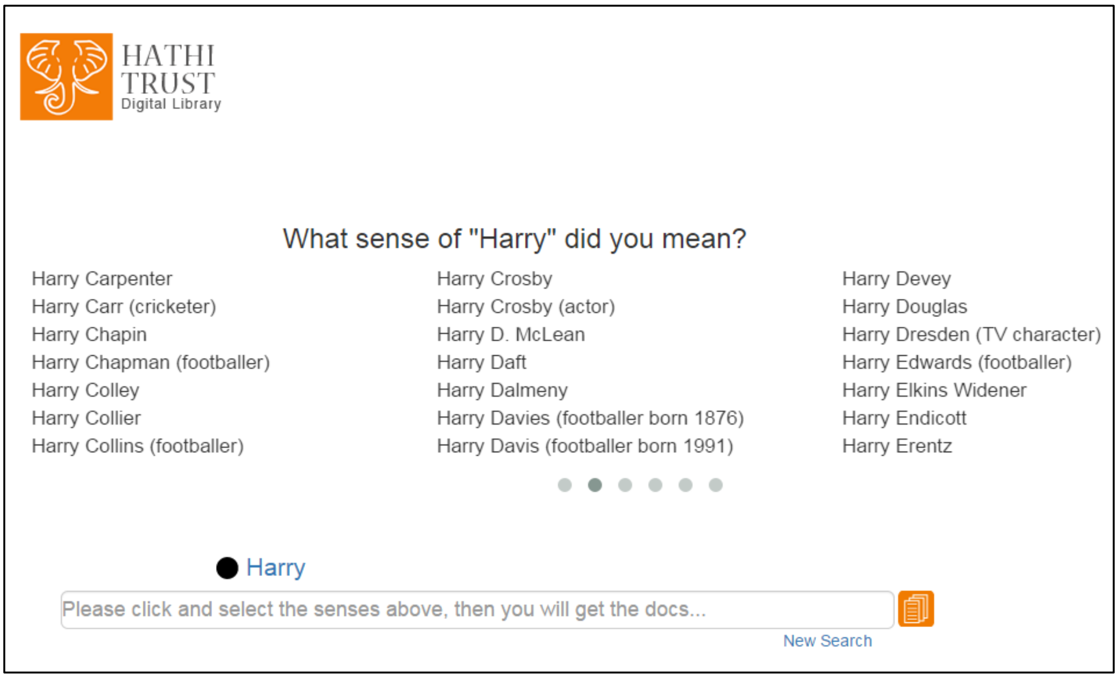


Figure 57: Second result page of the example keyword “Harry”.

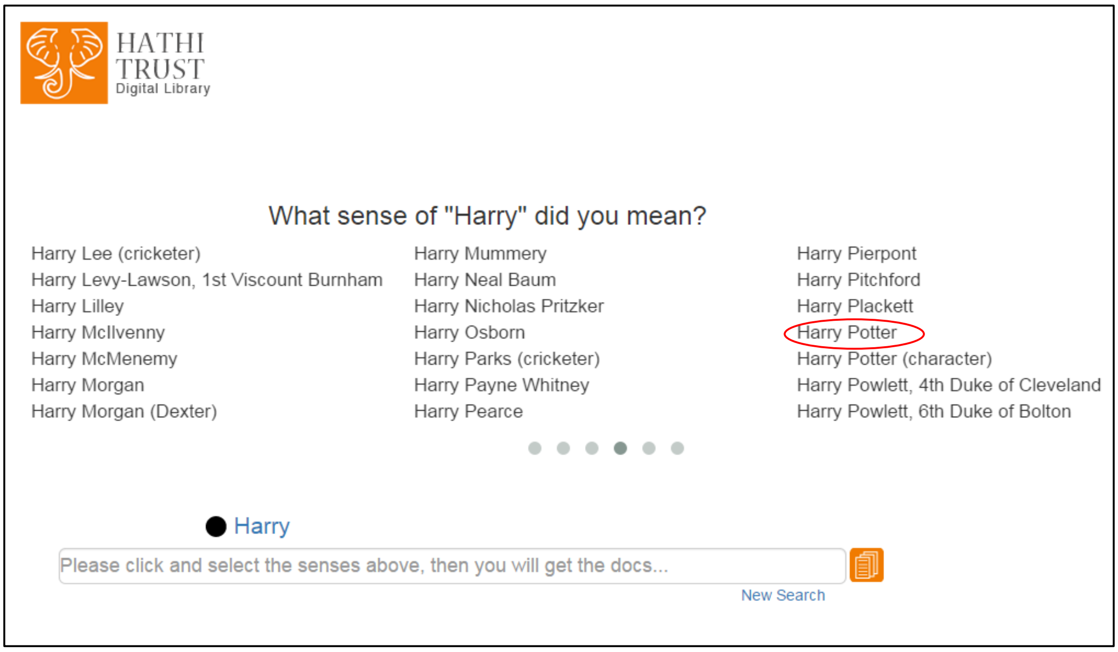


Figure 58 Content of the fourth result page which contains the target sense of the example keyword “Harry”.

5.4 Summary

As the overview of the interface, the interface of Capisco currently includes two main part:

query interface and user disambiguation interface. According to the example implementation in Section 5.3, it is easy and effective for users to understand and manipulate the designed interface, even they do not have professional semantic background. The feedbacks from the user study and related evaluation are described in the next chapter.

6 Evaluation

We introduced the three variations for the disambiguation interface for Capisco, see Chapter 5. The first interface sorts the senses of keyword according to the alphabet (Interface A – alphabetical). The second interface ranks the senses of keywords by the decreasing the amount of the corresponding documents (Interface B - #documents). The last interface orders the senses of the keywords by their likelihood (Interface C – likelihood of sense): the more popular the sense is for a given keyword, the higher ranking it has.
In this chapter, the user feedbacks for each interface are presented, analyzed and evaluated. Also, user feedbacks of three disambiguation interfaces are compared.

6.1 Methodology

We designed 16 experiments for the user study. In each experiment, the participants are asked to find the target concept according to the given keyword and target description. After doing the search, there are some questions needs to answer. In this user study, users do not need to search for the documents. Our emphasis is to observe how three disambiguation interfaces influence users' search strategies and seeking behavior.

24 participants were invited to complete the experiments. The 16 experiments were divided into 4 groups: very easy, easy, medium and complicated. In order to evaluate the performance of all three different interfaces equally, all the 16 experiments are be tested by using these three interfaces separately (Interface A, Interface B and Interface C). For each participant, there are 3 experiments that are picked up from each group (totally 4 groups). The three experiments from each group are set up in three interfaces separately. Therefore, each participant is totally asked to finish 12 experiments.

We did not restrict the time for the participants to find a target sense of the keyword, so the participants can keep searching until they find the satisfied keyword senses. Afterwards, the participants are allowed to mark the questions about processing in terms the alignment of understanding, ease of use and positive feedback. In each experiment, the questions that the participants are asked are the same (see the example in Figure 59). There is a Likert scale that is built for each question (Table 2). From the score 1 to 5, the higher the score is, the more negative the feedback is.

Likert scales from 1 to 5:

a) Did the system behave in the way that you expected it to be?
   1) exactly as I expected
   2) pretty much as I expected
   3) kind of as I expected
   4) quite different as I expected
   5) totally different as I expected

b) What do you think about the searching processing in this system?
   1) Very easy to use
   2) Easy to use
   3) Medium to use
   4) Hard to use
   5) Very hard to use

c) Are you satisfied about the searching results?
   1) very satisfied
   2) satisfied
   3) Medium
   4) Not satisfied
   5) not satisfied at all

Figure 59: The questions in user study that are based on Likert scale.

Score	Likert Scale
1	Strongly Agree
2	Agree
3	Neutral
4	Disagree
5	Strongly Disagree

Table 2: Table displays the relation between the scores and their related Likert scale.

The ranking of the target sense for each experiment has a crucial influence on the participant's user experience feedback. In order to receive the comprehensive feedbacks and get the comparison results effectively and objectively, the ranks of the target sense in three interfaces are all needed to be taken into consideration. In this condition, the selected target sense is needed to ranks differently in three interfaces. An example of the experiment and the rank of target concept in the three disambiguation interfaces are given in the next section.

6.2 Study Setup

The total 16 experiments are divided into 4 groups. The degree of difficulty is increasing from

Group 1 to Group 4. The experiments in Group 1 are the simplest. The options of the senses are not more than 7 which are displayed in just one column. The experiments in Group 2 are set to be easy, which contain between 8 and 21 senses. The senses are displayed in one result page, but they might have 2 columns or 3 columns. In Group 3, the difficulty scale for the experiments is medium. The options are between 22 and 42, which are contained in two result pages. The experiments in the fourth group are the hardest. The options of the results are more than 42, and the result pages are at least 3 pages. Sometimes, it could be 10 pages or more.

In each experiment, the search keyword and target sense are offered to the users. Here, we take experiment 15 as the example: You are interested to find out about Harry Potter. You enter the search term "Harry".

By using the keyword "Harry", the system offers 125 concepts for disambiguation. The target sense is set to be "Harry Potter". The ranks of the target sense "Harry Potter" in three differences vary greatly (see Table 3).

Interface	**Rank**
Interface A	Rank:81,Page 4
Interface B	Rank:97,Page5
Interface C	Rank:2, Page 1

Table 3: The ranks of target sense "Harry Potter" in three interfaces (Rank refers to the ranking of the target sense in all options and Page refers to the page number that the target sense is displayed).

It is obvious that, due to different ranking method, the ranks of target concept in three different disambiguation interfaces are substantially different. Assumedly, the influence of different ranks of target concept in three disambiguation interfaces on the users' search strategies and search behaviors are varied according to different degree of difficulty.

6.3 Results

In this section, we present the results of participants' experiments in detail. In each group, the aggregated results are analyzed and compared based on different disambiguation interfaces at first. At the end, the whole results are compared and analyzed by group and different disambiguation interfaces separately.

6.3.1 Group 1 (very easy – about 1 to 7 options)

The first group (experiment 1- experiment 4) are the simplest. The number of the disambiguated senses from Capisco is not more than 7. The options are displayed in just one column (three columns in each result page, 7 disambiguated concepts in each column) and very easy to have a look of all of them.

Experiment 1 ("pie")

Setup: You are interested to find out about the food Pie. Please enter the search term "Pie".

Interface A

Table 4: Results for interface A (alphabetical) for experiment 1(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:1	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P1	Y	4	3	Y	3	Y
P5	Y	1	3	Y	3	
P11	Y	1	1	Y	1	
P13	Y	3	3	Y	4	Y
P17	Y	2	2	Y	2	
P21	Y	2	1	Y	2	

Interface B

Table 5: Results for interface B (by #documents) for experiment 1 (Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:1	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P3	Y	1	1	Y	1	
P6	N	3	2	Y	4	
P10	Y	1	1	Y	1	

P15	Y	2	2	Y	3	
P19	N	4	3	N	4	Y
P23	Y	2	2	Y	2	

Interface C

Table 6: Results for interface C (by likelihood) for experiment 1((Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:1	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P2	Y	2	2	Y	3	Y
P7	Y	5	2	N	5	
P9	Y	2	2	N	4	
P14	Y	1	1	Y	2	
P18	Y	1	1	Y	1	
P22	Y	3	2	Y	3	Y

Experiment 2 ("sad")

Setup: You are interested to find out about the village called Sad which in Podlaskie Voivodeship, Poland. You enter the search term "Sad".

Interface A

Table 7: Results for interface A for experiment 2 (Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:1	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P4	Y	3	2	Y	3	Y
P2	Y	1	1	Y	1	
P10	Y	2	1	Y	1	
P16	N	4	3	N	4	Y
P20	Y	2	2	Y	3	

P24	Y	2	1	Y	2	

Interface B

Table 8: Results for interface B for experiment 2(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:1	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P2	Y	1	2	Y	2	
P5	Y	1	2	Y	1	
P9	Y	1	1	Y	1	
P14	Y	1	1	Y	1	
P18	Y	3	3	Y	3	
P22	Y	1	2	Y	1	

Interface C

Table 9: Results for interface C for experiment 2(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:4	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P1	Y	1	2	Y	1	
P8	N	5	5	N	5	
P2	Y	3	3	Y	3	Y
P13	Y	1	1	Y	1	
P17	Y	2	2	Y	2	
P21	Y	2	1	Y	2	

Experiment 3("window")

Setup: You are interested to find out about the Windows Computing System. You enter the search term "Window".

Interface A

Table 10: Results for interface A for experiment 3(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. “Is First” means whether this is the first experiment that the participant does).

Rank:1	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P3	Y	2	2	Y	2	Y
P7	Y	4	2	Y	4	
P9	Y	2	1	Y	2	
P15	Y	3	2	Y	3	Y
P19	Y	1	2	Y	2	
P23	Y	2	2	Y	2	

Interface B

Table 11: Results for interface B for experiment 3(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. “Is First” means whether this is the first experiment that the participant does).

Rank:1	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P1	Y	1	2	Y	2	
P8	Y	1	1	Y	1	
P12	Y	1	2	Y	2	
P13	Y	1	1	Y	1	
P17	Y	3	2	Y	3	Y
P21	Y	1	1	Y	2	

Interface C

Table 12: Results for interface C for experiment 3(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. “Is First” means whether this is the first experiment that the participant does).

Rank:3	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P4	Y	1	2	Y	2	
P5	Y	2	2	Y	1	
P11	Y	3	2	Y	3	
P16	Y	2	2	Y	2	
P20	Y	2	2	Y	2	
P24	Y	3	3	Y	3	Y

Experiment 4(metal)

Setup: You are interested to find out about the TV series called Mental. You enter the search term “Mental”.

Interface A

Table 13: Results for interface A for experiment 4(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. “Is First” means whether this is the first experiment that the participant does).

Rank:1	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P2	Y	1	2	Y	1	
P8	Y	1	1	Y	1	
P4	Y	1	2	Y	1	
P14	Y	2	2	Y	2	Y
P18	Y	1	1	Y	1	
P22	Y	1	2	Y	1	

Interface B

Table 14: Results for interface B for experiment 4(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. “Is First” means whether this is the first experiment that the participant does).

Rank:1	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P4	Y	1	2	Y	1	
P7	Y	3	2	Y	3	
P11	Y	1	2	Y	1	
P16	Y	2	2	Y	2	
P20	Y	3	2	Y	2	Y
P24	Y	2	1	Y	2	

Interface C

Table 15: Results for interface C for experiment 4(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:1	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P3	Y	1	1	Y	1	
P6	Y	3	1	Y	3	
P10	Y	3	2	Y	2	Y
P15	Y	1	2	Y	2	
P19	Y	2	2	Y	2	
P23	Y	2	2	Y	2	Y

6.3.1.1 Aggregation of data and Analysis of results

This section aggregates the available data for group 1. The feedback data of three disambiguation interfaces are aggregated and compared according to three different Likert scale. In total 72 study cases, there are 3 participants who failed finding target sense. According to the Likert scale score table above (Table 2), the average score of alignment of understanding, ease of use and positive experience is 1.97,1.89 and 2.14 (score 3 means medium, the smaller the number is, the better the feedback is).

Aggregated results

Figure 60 shows an overview of the alignment of understanding between system interaction and user expectation. I compare all three interfaces A, B and C. For each, the chart shows if the participants managed to find the answer ("yes") or not ("no). The colors indicate the participants' feedback to the question "Did the system behave in the way you expected it?". From Figure 60 it can be seen that overall interface B received the best feedback, while interface C seemed receive less positive feedback than others.

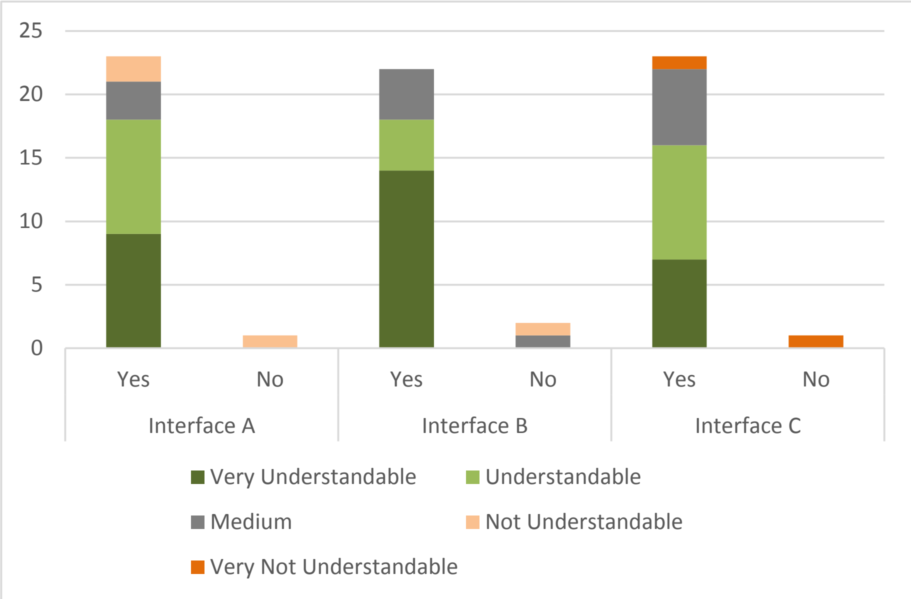


Figure 60: Figure displays the chart for alignment with understanding for group 1.

Figure 61 provides an overview of users' feedback about the ease of use for the system. I compare all three interfaces A, B and C. For each, the chart shows if the participants managed to find the answer ("yes") or not ("no). The colors indicate the participants' feedback to the question "what do you think about the search process?" From Figure 61 it can be seen that overall interface B performs the best, while interface C seemed perform the worst.

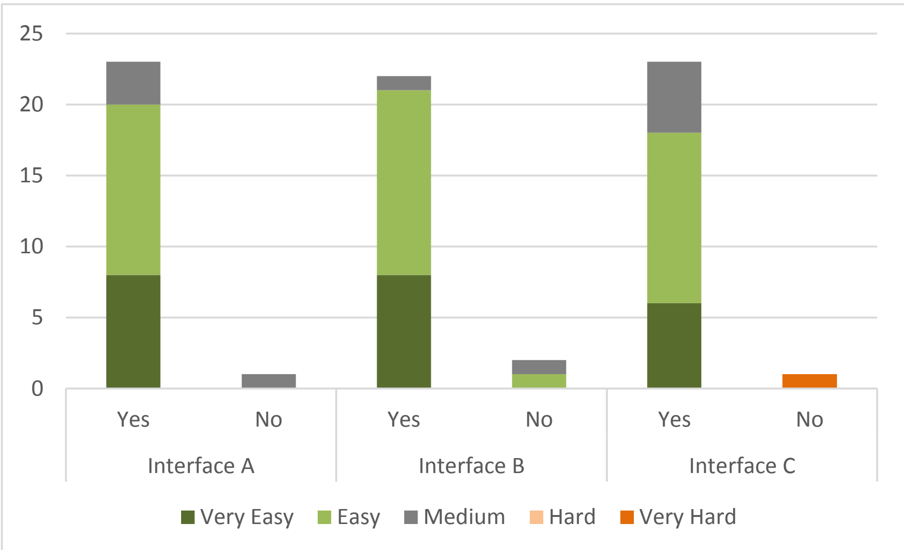


Figure 61: Figure displays the chart for ease of use for group 1.

Figure 62 displays an overview of positive feedback from the participants. I compare all three interfaces A, B and C. For each, the chart shows if the participants managed to find the answer ("yes") or not ("no). The colors indicate the participants' feedback to the question "Are you satisfied with the offered senses?" We can tell from the figure 62 that overall interface B receives the most positive feedback, while interface C receives seemed the most complicated.

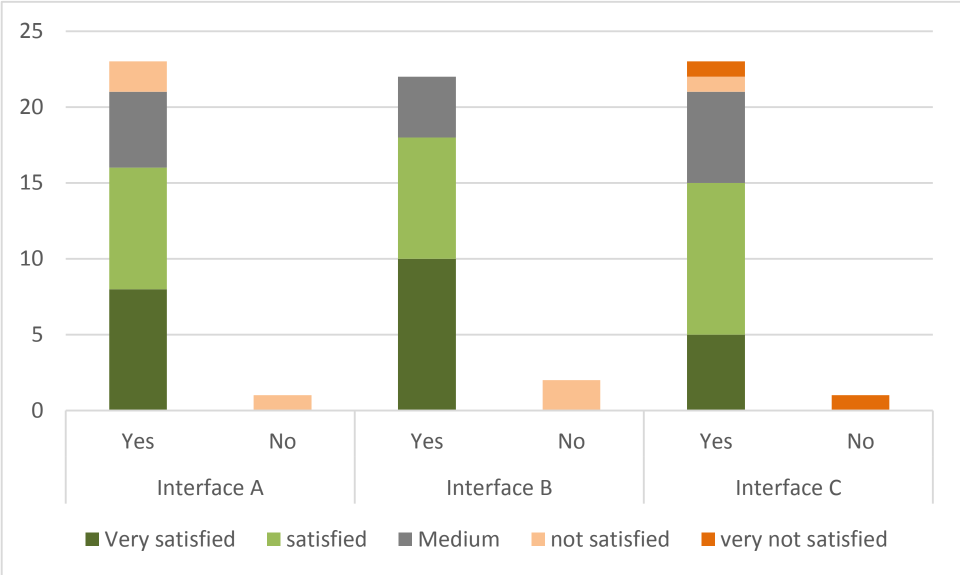


Figure 62: Figure displays the chart for positive experience for group 1.

Analysis of group 1 results

The first group (experiment 1- experiment 4) is set to be very simple and the options of the senses are not more than 7. The results will be displayed in just one column and very easy to have a look of all of them. I therefore assumed that the feedback of first group would be very positive and the different disambiguation interfaces will not influence the user searching process too much.

However, we can tell from the data that the feedback of group 1 is not as unanimous as expected and some participants even failed to find the correct sense. For example, the alignment with understanding varied from 18 positive for interface B to only 15 positive for interface C. I believe the reason for this is that at this point the participants are not yet familiar with this searching system and little understanding yet how it works when they use it at the first time. From their feedback we can see that many of the first-time users thought this searching system would work like Google (13 participants). However, the participants' feedbacks are becoming more positive after doing the first experiment in group 1 (xxx be precise where the data is from).

Because we aimed to get feedback of first time users, the participants were not allowed to interact with the system before encountering group 1 study. Therefore study 1 described the feedback of total novices with this interface.

6.3.1.2 Use feedback on doing group 1 study and suggested improvements

Apart from the collected data above, the users also provide the feedbacks and suggestions for improving the user interface. As we introduced before, the idea of the query interface is similar to the traditional keyword-based search engine, which makes the non-professional users feel familiar and easy to use. According to the feedback from P3 (Participant 3), P4, and P5 etc., they agreed that the query interface is simple and user-friendly. Also, they thought the searching processing and result interface are similar to the regular search engine like Google before executing the experiments. However, some users (e.g. P3, P13, and P19) feel confused when they see the disambiguation interfaces and do not know what they need to do. Also, there are some participants, such as participant 1, participant 6, participant 7 etc., ignored the requirement that the first letter of keyword must be capital.

Suggested Improvements:

In order to address the problems proposed above, some improvement are required. First of all, an instruction or user guide of the searching process should be displayed to the first-time users. Besides, the notice of capital compulsory needs to be highlighted in the query interface. Alternatively, the input box is able to convert the first letter of keyword to uppercase automatically.

6.3.2 Group 2 (easy – about 8 to 21 options)

The second group is the experiments that are from 5-8. The numbers of the options are between 8 and 21. The senses will be displayed in one result page, but they might have 2 columns or 3 columns.

Experiment 5("base")

Setup: You are interested to find out about the military base. You enter the search term "Base".

Interface A

Table 16: Results for interface A for experiment 5(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:9	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First

P1	Y	1	1	Y	1	
P5	Y	1	2	Y	1	
P10	Y	2	2	Y	2	
P5	Y	1	1	Y	1	
P17	Y	2	2	Y	2	
P21	Y	2	1	Y	2	

Interface B

Table 17: Results for interface B for experiment 5(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:1	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P2	Y	1	2	Y	2	
P6	Y	1	1	Y	1	
P11	Y	1	1	Y	1	
P14	Y	1	1	Y	1	
P18	Y	1	2	Y	2	
P22	Y	2	2	Y	2	

Interface C

Table 18: Results for interface C for experiment 5(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:4	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P3	Y	1	1	Y	1	
P7	N	5	1	N	5	
P9	Y	1	1	Y	1	
P15	Y	1	2	Y	2	
P19	Y	2	2	Y	2	

P23	Y	2	2	Y	2	

Experiment 6("ktv")

Setup: You are interested to find out about Florida Channel Kids & Teens TV. You enter the search term "KTV".

Interface A

Table 19: Results for interface A for experiment 6(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:6	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P3	Y	1	1	Y	1	
P7	Y	4	2	Y	4	
P9	Y	1	1	Y	1	
P15	Y	2	2	Y	2	
P19	Y	1	2	Y	1	
P23	Y	2	2	Y	2	

Interface B

Table 20: Results for interface B for experiment 6(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:8	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P1	Y	1	1	Y	1	
P5	N	4	4	Y	4	
P12	Y	1	1	Y	1	
P13	Y	1	1	Y	1	
P17	Y	2	2	Y	2	
P21	Y	2	2	Y	2	

Interface C

Table 21: Results for interface C for experiment 6(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:6	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P4	Y	2	2	Y	2	
P6	Y	1	1	Y	1	
P11	Y	2	2	Y	2	
P16	Y	2	2	Y	2	
P20	Y	2	1	Y	2	
P24	Y	2	3	Y	2	

Experiment 7("ginger")

Setup: You are interested to find out about the band called Ginger. You enter the search term "Ginger".

Interface A

Table 22: Results for interface A for experiment 7(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:3	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P2	Y	1	2	Y	1	
P6	Y	1	1	Y	1	
P12	Y	1	2	Y	1	
P14	Y	1	2	Y	2	
P18	Y	1	1	Y	1	
P22	Y	2	2	Y	2	

Interface B

Table 23: Results for interface B for experiment 7(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:14	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P4	Y	2	2	Y	2	
P8	Y	2	1	Y	2	
P9	N	1	1	Y	1	
P16	Y	2	2	Y	2	
P20	Y	3	2	Y	2	
P24	Y	2	2	Y	2	

Interface C

Table 24: Results for interface C for experiment 7(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:3	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P1	Y	1	2	Y	1	
P8	Y	1	1	Y	1	
P10	Y	2	2	Y	2	
P13	Y	1	2	Y	1	
P17	Y	2	2	Y	2	
P21	Y	2	1	Y	2	

Experiment 8("rat")

Setup: You are interested to find out about Brown Rat. You enter the search term "Rat".

Interface A

Table 25: Results for interface A for experiment 8(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First"

means whether this is the first experiment that the participant does).

Rank:2	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P4	Y	1	2	Y	1	
P8	Y	1	1	Y	1	
P8	Y	1	1	Y	1	
P16	Y	2	2	Y	2	
P20	Y	1	2	Y	1	
P24	Y	2	1	Y	2	

Interface B

Table 26: Results for interface B for experiment 8(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:2	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P3	Y	1	1	Y	1	
P7	Y	3	2	Y	3	
P10	Y	1	2	Y	1	
P15	Y	1	2	Y	2	
P19	Y	2	3	Y	3	
P23	Y	2	2	Y	2	

Interface C

Table 27: Results for interface C for experiment 8(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:4	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P2	Y	1	2	Y	1	
P5	Y	2	3	Y	3	

P12	Y	2	2	Y	2	
P14	Y	1	1	Y	2	
P18	Y	1	1	Y	1	
P22	Y	2	1	Y	2	

6.3.2.1 Aggregation and Analysis of results

This section aggregates the available user case data for group 2. In this group, there are 3 failed participants (one less than last group), and the average scores of three questions is lower than Group 1, which means Group2's performance is better than Group 2. They are 1.61 in alignment of understanding, 1.67 in ease of use and 1.72 in positive experience. The average scores are all lower than score 2 in three aspects, which are very positive feedbacks.

Aggregated results

Figure 63 indicates an overview of the alignment of understanding between system interaction and user expectation. I compare all three interfaces A, B and C. For each, the chart shows if the participants managed to find the answer ("yes") or not ("no). The colors indicate the participants' feedback to the question "Did the system behave in the way you expected it?". From Figure 63 it can be seen that overall interface A received the best feedback, while interface B seemed the most complicated.

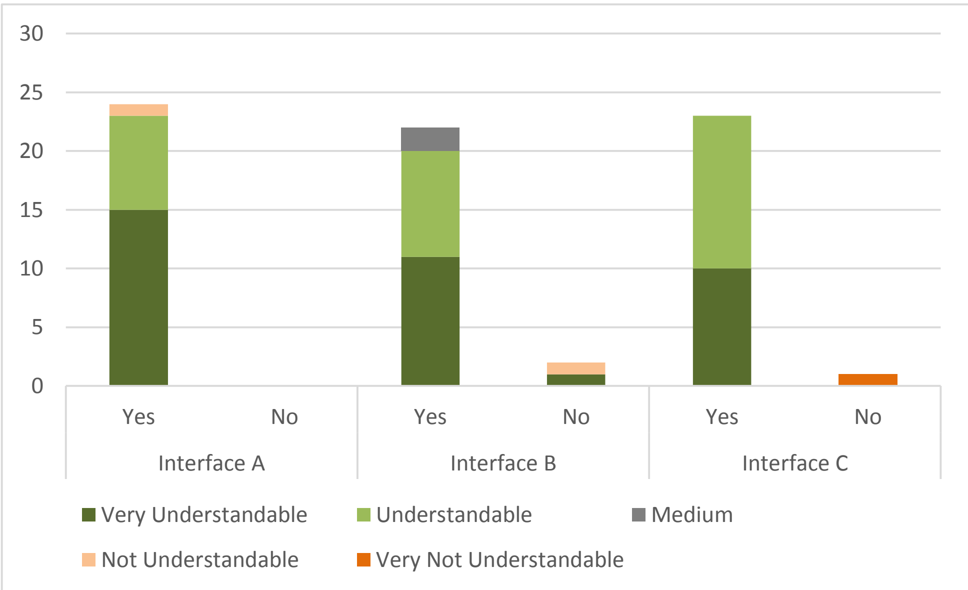


Figure 63: The Figure displays the chart for alignment with understanding for group 2.

Figure 64 provides an overview of users' feedback about the ease of use for the system. I compare all three interfaces A, B and C. For each, the chart shows if the participants managed to find the answer (“yes”) or not (“no). The colors indicate the participants’ feedback to the question “what do you think about the search process?” From Figure 64 it can be seen that overall interface A performs the best, while interface B seemed perform the worst.

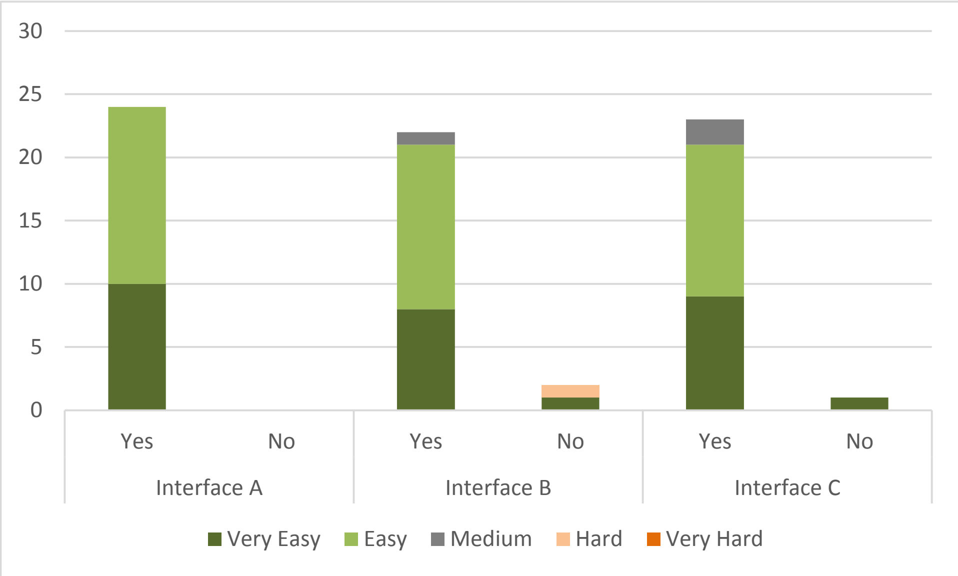


Figure 64: The figure displays the chart for ease of use for group 2.

Figure 65 displays an overview of positive feedback from the participants. I compare all three interfaces A, B and C. For each, the chart shows if the participants managed to find the answer (“yes”) or not (“no). The colors indicate the participants’ feedback to the question “Are you satisfied with the offered senses?” We can tell from the figure 65 that overall interface A receives the most positive feedback (23), while interface B receives less positive feedback (20).

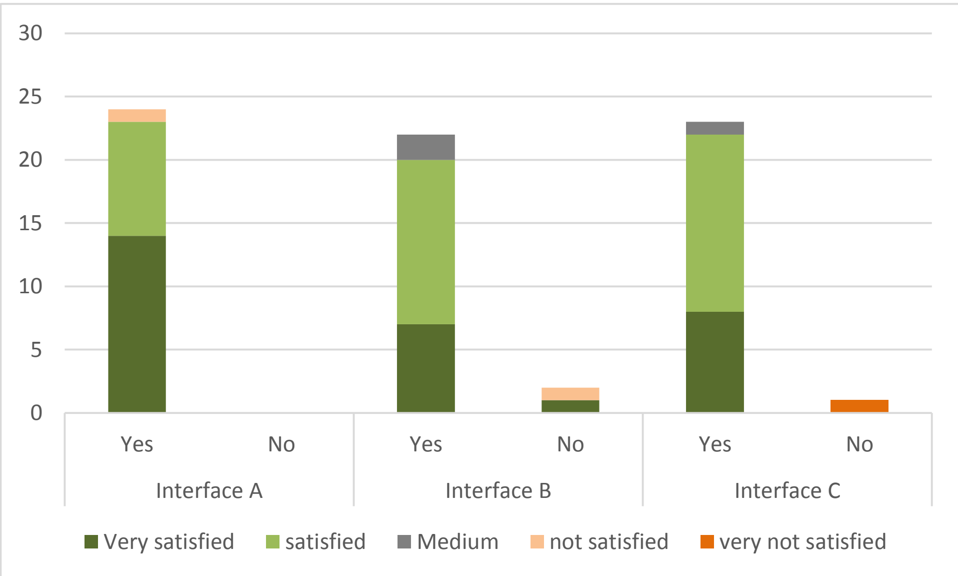


Figure 65: Figure displays the chart for positive experience for group 2.

Analysis of group 2 results

The second group is the experiments that are from 5-8. The amount of the options are from 8 to 21. One result page can afford them, but they might have 2 columns or 3 columns. Assumedly, the influence from different interfaces is still slight and the feedbacks are quite positive as group 1.

Comparing the Figure 5 in this group with equivalent figure in Group 1 (i.e., Figure 1), it is obvious that the feedback of the group 2 is more positive than Group 1. The feedbacks from the successful participants are generally very positive (only one participant gives the negative feedback). The average Likert scale of three questions are all about 1.5. In my opinion, I believe this is because that they participants are more confident and familiar with this system after finishing the experiments in Group 1. It is interesting to see that, one participant (participant 9) gives very positive feedback even he failed. This is because he thought he found the right sense. As expected, the differences between three different interfaces are slight in this stage.

6.3.2.2 Use feedback on doing group 2 study and suggested improvements

In Group 2, there is a problem that has been proposed by quite few participants. P7 (Participant 7), P15 and some other participants said that the meaning of some disambiguation concepts are so difficult to understand and sometimes they are still not sure which sense they are looking for after reading all the possible senses. They want to see more explanation for the senses which can help them to understand the literal meaning of them.

Suggested Improvements:

Regarding to deal with this problem, it would be reasonable that there are explanations for each concept that present after each sense.

6.3.3 Group 3(medium – about 22 to 42 options)

The third group ranges from experiment 9 to experiment 12. The difficult scale for this group is medium. The options are between 22 and 42, and are contained in 2 result pages.

Experiment 9("band")

Setup: You are interested to find out about the band which refers to an idempotent semigroup in math. You enter the search term "Band".

Interface A

Table 28: Results for interface A for experiment 9(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First"

means whether this is the first experiment that the participant does).

Rank:2	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P1	Y	2	2	Y	3	
P5	N	2	1	Y	1	
P8	Y	1	1	Y	1	
P13	Y	2	1	Y	2	
P17	Y	2	2	Y	2	
P21	Y	1	1	Y	2	

Interface B

Table 29: Results for interface B for experiment 9(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:4	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is Firs
P3	Y	2	2	Y	2	
P7	Y	3	2	Y	3	
P12	Y	1	2	Y	1	
P15	Y	2	2	Y	2	
P19	Y	2	2	Y	3	
P23	Y	2	2	Y	2	

Interface C

Table 30: Results for interface C for experiment 9(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:19	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P4	Y	2	2	Y	2	
P8	Y	1	1	Y	1	

P9	Y	2	2	Y	2	
P16	Y	2	3	Y	2	
P20	Y	2	2	Y	3	
P24	Y	2	3	Y	2	

Experiment 10("company")

Setup: You are interested to find out about film called Company. You enter the search term "Company".

Interface A

Table 31: Results for interface A for experiment 10(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:5	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P2	Y	1	2	Y	1	
P6	Y	1	1	Y	1	
P9	Y	1	1	Y	1	
P14	Y	1	2	Y	2	
P18	Y	1	2	Y	1	
P22	Y	2	2	Y	2	

Interface B

Table 32: Results for interface B for experiment 10(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:26	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P1	Y	2	2	Y	2	
P8	Y	1	2	Y	1	
P11	Y	2	2	Y	2	

P13	Y	2	1	Y	2	
P17	Y	2	2	Y	2	
P21	Y	2	2	Y	2	

Interface C

Table 33: Results for interface C for experiment 10(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:4	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P3	Y	1	1	Y	1	
P5	Y	3	3	Y	4	
P10	Y	2	2	Y	2	
P15	Y	2	2	Y	2	
P19	Y	2	2	Y	2	
P23	Y	2	2	Y	2	

Experiment 11

Setup: You are interested to find out about an English TV series called Skin. You enter the search term "Skin".

Interface A

Table 34: Results for interface A for experiment 11(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:23	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P4	Y	2	3	Y	2	
P7	N	4	2	N	4	
P10	Y	2	2	Y	2	
P16	Y	2	2	Y	2	

P20	Y	1	2	Y	2	
P24	Y	2	1	Y	2	

Interface B

Table 35: Results for interface B for experiment 11(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:1	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P2	Y	2	2	Y	2	
P5	N	5	5	Y	5	
P9	Y	2	2	Y	3	
P14	Y	2	1	Y	2	
P18	Y	2	2	Y	3	
P22	Y	2	2	Y	2	

Interface C

Table 36: Results for interface C for experiment 11(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:4	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P1	Y	1	1	Y	1	
P6	Y	1	1	Y	1	
P12	Y	2	2	Y	2	
P13	Y	1	2	Y	1	
P17	Y	2	2	Y	2	
P21	Y	2	1	Y	2	

Experiment 12("fuji")

Setup: You are interested to find out about Fuji Apple. You enter the search term "Fuji".

Interface A

Table 37: Results for interface A for experiment 12(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank3	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P3	Y	1	1	Y	1	
P8	Y	1	1	Y	1	
P12	Y	1	2	Y	1	
P15	Y	2	1	Y	2	
P19	Y	1	2	Y	1	
P23	Y	2	2	Y	2	

Interface B

Table 38: Results for interface B for experiment 12(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:23	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P4	Y	2	3	Y	2	
P6	Y	1	1	Y	1	
P10	Y	2	2	Y	2	
P16	Y	2	3	Y	2	
P20	N	3	3	N	5	
P24	Y	2	2	Y	2	

Interface C

Table 39: Results for interface C for experiment 12(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2).

Rank:9	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P2	Y	1	1	Y	1	
P7	Y	3	1	Y	1	
P11	Y	2	2	Y	2	
P14	Y	1	2	Y	1	
P18	Y	1	1	Y	2	
P22	Y	2	2	Y	2	

6.3.3.1 Aggregation and Analysis of results

This section aggregates the available data for Group 3. With the increasing of difficulty, the score is incrementing as well. In this group, 4 users did not find the target sense, which is as same number as Group 1. Specifically, the average score of alignment of understanding is 1.75, the score of ease of use is 1.85 and the average score for positive experience is 1.93. They are all higher than the score in Group 2, but lower than Group 1 in separate aspects.

Aggregated results

Figure 66 displays an overview of the alignment of understanding between system interaction and user expectation. I compare all three interfaces A, B and C. For each, the chart shows if the participants managed to find the answer ("yes") or not ("no). The colors indicate the participants' feedback to the question "Did the system behave in the way you expected it?". From Figure 66, it can be seen that overall interface A performs a little better than another 2 interfaces (1 more positive feedback than C and 2 more than B), while interface B seemed the most complicated.

Figure 67 provides an overview of users' feedback about the ease of use for the system. I compare all three interfaces A, B and C. For each, the chart shows if the participants managed to find the answer ("yes") or not ("no). The colors indicate the participants' feedback to the question "what do you think about the search process?" From Figure 67 it can be seen that overall interface A performs the best which even gets positive feedback from the failed participants. While interface B seemed to the most complicated. In interface A, both of the participants who fail to find the sense still give the positive feedback (one thought he successfully found the sense).

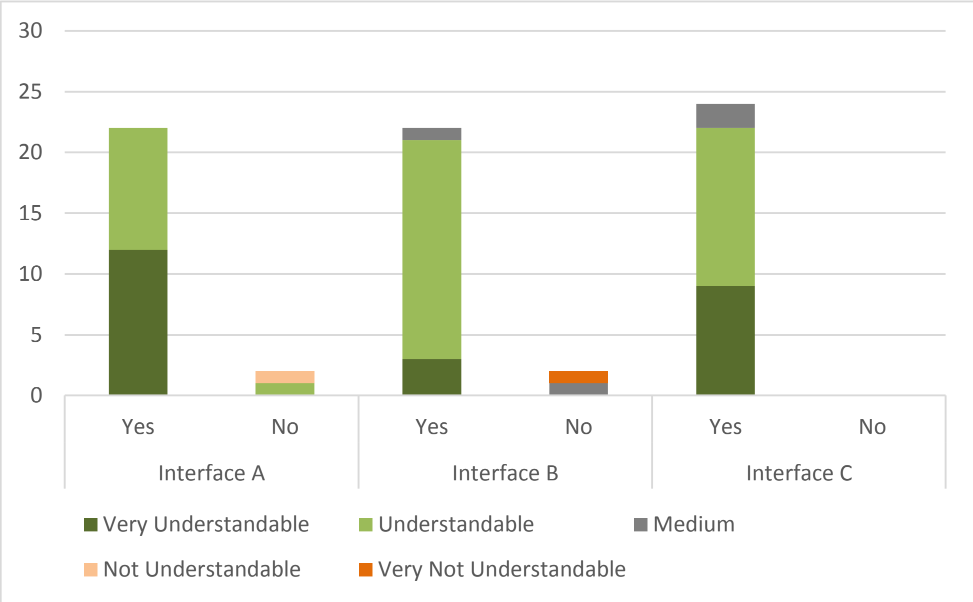


Figure 66: Figure displays the chart for alignment with understanding for group 3.

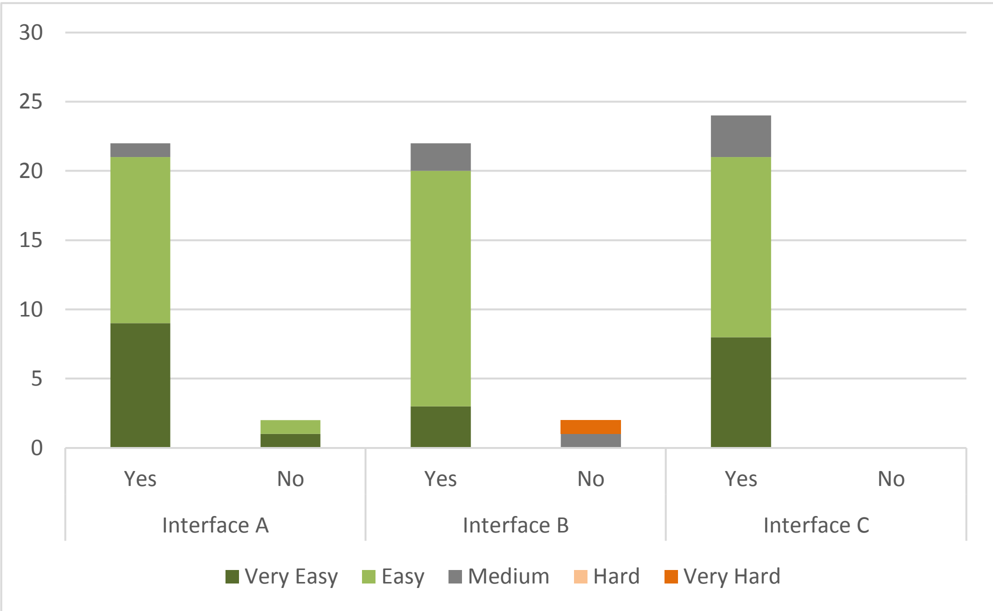


Figure 67: The figure displays the chart for ease of use for group 3.

Figure 68 displays an overview of positive feedback from the participants. I compare all three interfaces A, B and C. For each, the chart shows if the participants managed to find the answer ("yes") or not ("no). The colors indicate the participants' feedback to the question "Are you satisfied with the offered senses?" We can tell from the Figure 68 that overall both interface A and interface C receive the most positive feedback (22), while interface B performs the worst (18 positive feedbacks) .

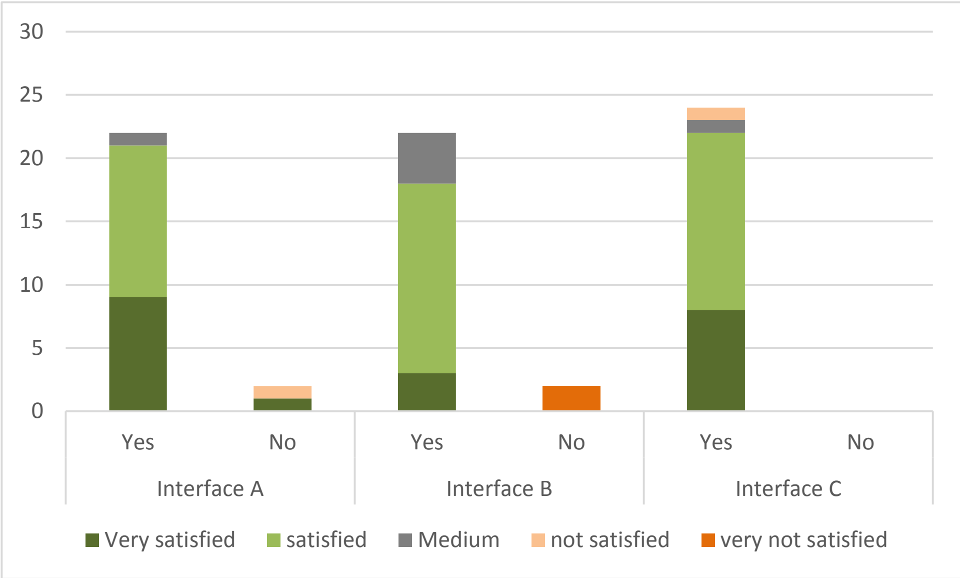


Figure 68: Figure displays the chart for positive experience for group 3.

Analysis of group 3 results

In this group, the options are between 22 and 42, and are contained in 2 result pages. Therefore, I assumed that the influence of the ranking in three interfaces will be much bigger than Group 2. Sometimes, it would cost more time finding the target sense of keyword. Also, the feedback would not be that positive as Group 2 due to difficult scale of this group.

Actually, the basic feedbacks from the participants are better than expected. Generally, more than three quarters of the participants give the positive feedbacks in three questions. Especially for the participants who successfully finding the senses, the feedbacks are mostly satisfied. It is surprise to notice that, the participant 7 still think this system is easy to use even he failed. In this group, the difference between these three interfaces are still not very clear.

6.3.3.1 User feedback on Group 3 study and suggested improvements

In Group 3, the concepts for disambiguation are listed more than one page. It is interesting to find that, several participants (e.g. P7) did not notice the pagination (grey dot) in concept result interface or they did not know the grey dot is clickable. Therefore they did not click it to go to next concept page. Also, I notice that most of the participants did not realize that the disambiguated concepts are sorted differently in three interfaces.

Suggested Improvement

It is necessary that we need to change the pagination color from grey to bright color. Also, there are more than one participant didn't realize the pagination is clickable at the beginning. One participant (Participant 9) suggested that it is better to change the pager from dot to the page number.

6.3.4 Group 4(hard- more than 42 options)

Group 4 contains the last four experiments and they are the most difficult. The options of the results are more than 42, and the result pages are at least 3 pages. Sometimes, it could be 10 pages or more.

Experiment 13("lucy")

Setup: You are interested to find out about the elephant-shaped building in Margate called Lucy the elephant. You enter the search term "Lucy".

Interface A

Table 40: Results for interface A for experiment 13(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:60	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P1	Y	1	2	Y	3	
P5	Y	1	3	Y	2	
P10	Y	2	3	Y	4	
P13	Y	2	2	Y	3	
P17	N	3	3	N	4	
P21	Y	2	2	Y	2	

Interface B

Table 41: Results for interface B for experiment 13(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:71	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P2	Y	1	4	Y	1	
P6	Y	3	3	Y	3	
P13	N	2	2	N	3	
P14	N	3	3	N	5	
P18	Y	2	2	Y	3	

P22	Y	2	3	Y	3	

Interface C

Table 42: Results for interface C for experiment 13(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:25	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P4	Y	2	2	Y	3	
P8	Y	2	3	Y	2	
P9	N	5	2	N	5	
P16	Y	2	3	Y	2	
P20	Y	2	2	Y	3	
P24	Y	3	3	Y	3	

Experiment 14("tiger")

Setup: You are interested to find out about a European attack helicopter called Eurocopter Tiger. You enter the search term "Tiger".

Interface A

Table 43: Results for interface A for experiment 14(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:6	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P3	Y	1	1	Y	1	
P7	Y	4	2	Y	4	
P9	Y	3	3	Y	2	
P15	Y	2	2	Y	2	
P19	Y	1	2	Y	2	
P23	Y	2	2	Y	2	

Interface B

Table 44: Results for interface B for experiment 14(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. “Is First” means whether this is the first experiment that the participant does).

Rank:60	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P1	Y	1	1	Y	1	
P8	Y	2	2	Y	1	
P10	Y	3	4	Y	3	
P13	Y	2	2	Y	2	
P17	Y	2	2	Y	2	
P21	Y	2	2	Y	2	

Interface C

Table 45: Results for interface C for experiment 14(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. “Is First” means whether this is the first experiment that the participant does).

Rank:11	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P2	Y	1	2	Y	1	
P6	N	4	2	N	3	
P11	Y	2	2	Y	2	
P14	Y	2	2	Y	2	
P18	Y	1	2	Y	2	
P22	Y	2	2	Y	2	

Experiment 15(“harry”)

Setup: You are interested to find out about Harry Potter. You enter the search term “Harry”.

Interface A

Table 46: Results for interface A for experiment 15(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:81	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P4	Y	2	3	Y	3	
P8	Y	1	1	Y	1	
P11	Y	2	2	Y	3	
P16	Y	2	3	Y	2	
P20	Y	2	2	Y	2	
P24	Y	2	2	Y	2	

Interface B

Table 47: Results for interface B for experiment 15(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:97	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P3	Y	1	1	Y	1	
P5	Y	4	4	Y	4	
P9	Y	3	3	Y	2	
P15	Y	2	2	Y	3	
P19	Y	3	2	Y	3	
P23	Y	3	2	Y	2	

Interface C

Table 48: Results for interface C for experiment 15(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:2	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P1	Y	1	1	Y	1	
P7	Y	1	1	Y	1	
P12	Y	2	1	Y	1	
P13	Y	1	1	Y	1	
P17	Y	2	2	Y	2	
P21	Y	2	1	Y	2	

Experiment 16("david")

Setup: You are interested to find out about a film in 1979 called "David". You enter the search term "David".

Interface A

Table 49: Results for interface A for experiment 16(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank:14	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P2	Y	2	3	Y	1	
P6	Y	1	1	Y	1	
P12	Y	2	2	Y	2	
P14	Y	1	2	Y	2	
P18	Y	2	2	Y	2	
P22	Y	2	2	Y	1	

Interface B

Table 50: Results for interface B for experiment 16(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2. "Is First" means whether this is the first experiment that the participant does).

Rank: Last 2 Page	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P4	Y	3	4	Y	3	
P7	N	5	5	N	5	
P11	N	3	4	N	5	
P16	Y	3	4	Y	3	
P20	Y	3	4	Y	2	
P24	N	2	4	N	5	

Interface C

Table 51: Results for interface C for experiment 16(Y means meet the requirements and Y means does not meet the requirement. The score corresponds to the Likert scale in table 2).

Rank:20	Successfully assigning concept	Alignment with understanding	Ease of use	Successfully finding concept	Positive experience	Is First
P3	Y	1	1	Y	1	
P5	Y	3	4	Y	4	
P10	Y	2	2	Y	2	
P15	Y	3	2	Y	2	
P19	Y	2	3	Y	2	
P23	Y	2	2	Y	2	

7.3.4.1 Aggregation and Analysis of results

This section aggregates the available data for group 4. The experiments in Group 4 are set to be the hardest. As expected, the performances of the participants in this group are the worst. Totally, there are 8 users who did not successfully find the target concept, which is double than another three groups. Similarly, the scores of three questions are much higher than other groups (2.15 in alignment of understanding, 2.35 in ease of use and 2.38 in positive experience). The average score in three aspects are all over score 2.

Aggregated results

Figure 69 displays an overview of the alignment of understanding between system interaction and user expectation. I compare all three interfaces A, B and C. For each, the chart shows if the participants managed to find the answer ("yes") or not ("no). The colors indicate the participants' feedback to the question "Did the system behave in the way you expected it?". From Figure 69, it can be seen that overall interface A received the best feedback, while interface B seemed the most complicated.

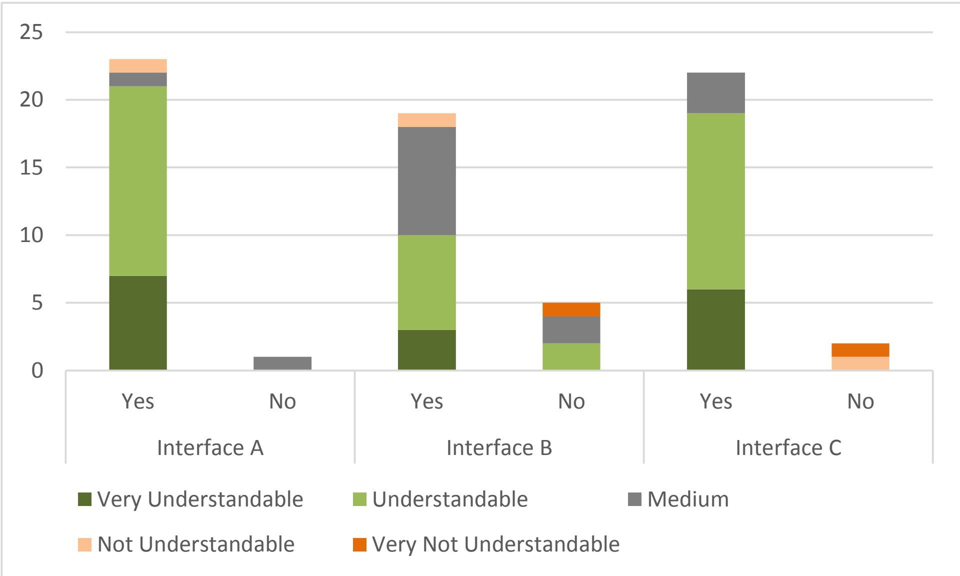


Figure 69: Figure displays the chart for alignment with understanding for group 4.

Figure 70 provides an overview of users' feedback about the ease of use for the system. I compare all three interfaces A, B and C. For each, the chart shows if the participants managed to find the answer ("yes") or not ("no). The colors indicate the participants' feedback to the question "what do you think about the search process?" From Figure 70 it can be seen that overall interface C performs the best, while interface C seemed perform the worst and the most complicated.

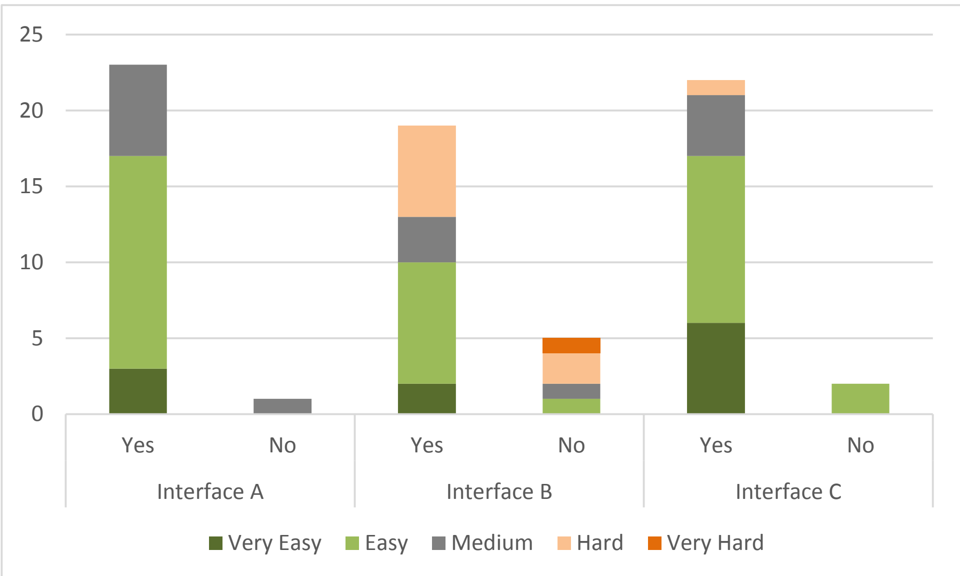


Figure 70: Figure displays the chart for ease of use for group 4.

Figure 71 displays an overview of positive feedback from the participants. I compare all three interfaces A, B and C. For each, the chart shows if the participants managed to find the answer ("yes") or not ("no). The colors indicate the participants' feedback to the question "Are you satisfied with the offered senses?" We can tell from the figure 71 that overall interface C receives the most positive feedback (one more than Interface A), while interface B seems receive the worst feedback (10 positive feedbacks).

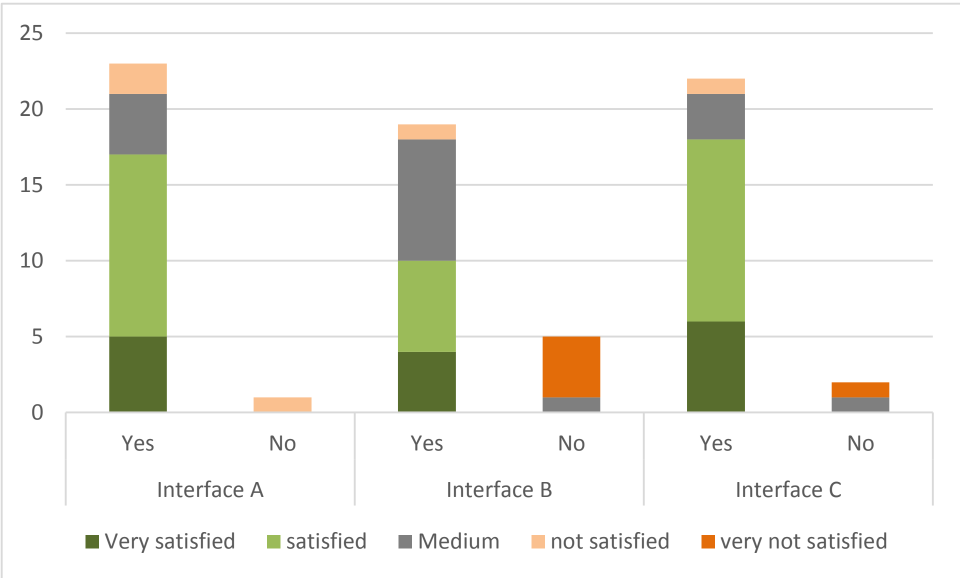


Figure 71: Figure displays the chart for positive experience for group 4.

Analysis of group 4 results

Group 4 contains the last four experiments. The options of the results are more than 42, and the result pages are at least 3 pages. Sometimes, it could be 10 pages or more. In this condition, the ranking of the results are very crucial to the users. If the target result's rank is low, it might very difficult to find it, especially when there are more than 5 pages. Sometimes, the users might give up searching due to the large amount of results.

From the charts and data above, it is apparent that, according to the increasing of difficult level, Group 4 received the most negative feedbacks compared with last three groups. For some of the participants, they even did not find the target senses at the end, when the results comes to more than 5 pages. Sometimes, participants missed the target sense because too many results would interrupt their searching. The participants would be less interested in finding out the target sense when results are more than 5 pages. Apparently, the user experience is more negative when the amount of the results comes to substantial. But, the users still agree that this system is considerably easy to use.

In this group, the feedback differences among three interfaces are complicated. Participants' feedbacks are mostly dependent on the ranking of the target sense. If the target sense in displayed in the first result page or first three result pages, the feedbacks are still positive. Otherwise, they feedbacks are basically quite negative (except 2 failed participants still believe this system is easy to use).

7.3.4.1 User feedback on Group 4 study and suggested improvement

Due to the large amount of the disambiguated concepts that are needed to be identified by the Capisco, the processing time for displaying concepts in this group is much longer than experiments in other groups. There are 7 participants mistakenly believed that the interface was not working or there were no concepts for disambiguation after waiting for a while. For example, P3 said "This is not working. Nothing is displayed". What's more, more than 5 participants gave up browsing the rest of the concepts after exploring the first three concept pages. They said it is too hard to find one target concept among more than one hundred of concepts.

Suggested Improvement

When the system is processing and identifying the disambiguated senses at the back, it would be better to have a processing icon displayed in front page. By doing this, the users can be tell that the search engine is still working. Besides, when the amount of results comes to a certain number, some solutions are supposed to be proposed to refine the concepts rather than display thunders or even thousands concepts to the users directly.

6.3.5 Overall of four groups feedbacks

From the data and analysis above, the data received from first-time participants in Group 1 cannot be counted as being objective with regards to the interface but rather captures the response of novice users. The feedbacks in Group 2 whose degree of difficulty is similar to Group 1 are more trustable. As the increment of difficulty, the feedbacks of the three questions are generally getting worse. As showed in Figure 72, the participants who failed finding the target senses in Group 1 are more than Group 2. However, since Group 2, the numbers of failed participants are increasing. Especially in Group 4, the amount of failed participants are double than other groups.

Specifically, we set up the Likert scale from 1 to 5. The smaller the number is, the more positive feedback it receives. For example, in terms of alignment of understanding, we can tell from Figure 73, from Group 1 to Group 4, it receives 52 positive feedbacks, 67 positive feedbacks, 57 positive feedbacks and 52 positive feedbacks separately. Similarly, with regards to the ease of use and positive experience, the feedbacks are becoming more negative with the increasing level of difficulty (Figure 74and Figure 75).

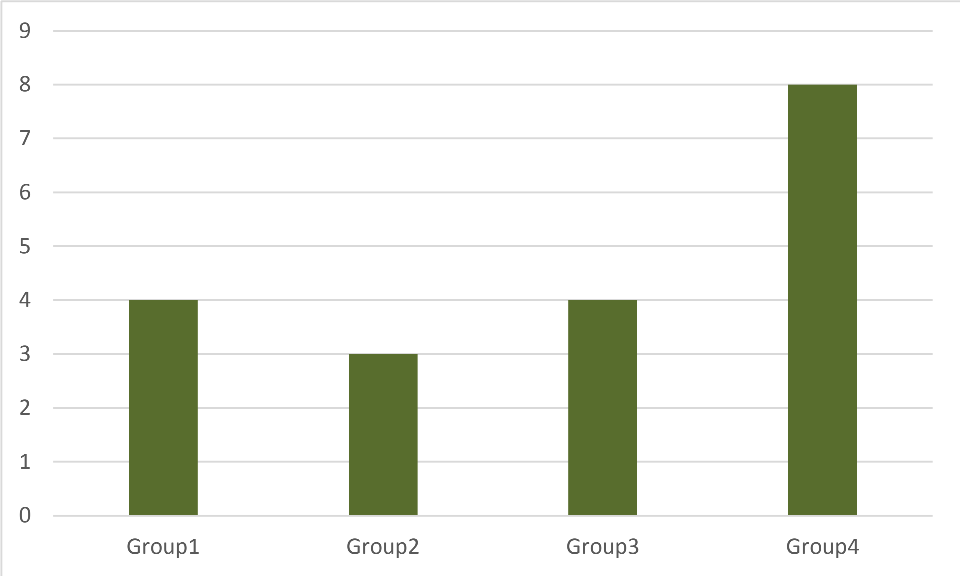


Figure 72: Figure displays the numbers of the failed participants in four groups.

Especially in Group 4, the quality of the feedbacks are mostly based on the rank of the target sense in the result pagers. For example, if the target sense contains many related documents or very popular, the feedback in interface B or Interface C are still positive. If the user's target sense is started by the letter A or B, the interface A receives better feedbacks. In this condition, the differences of the three interfaces do not make too much sense to the participants.

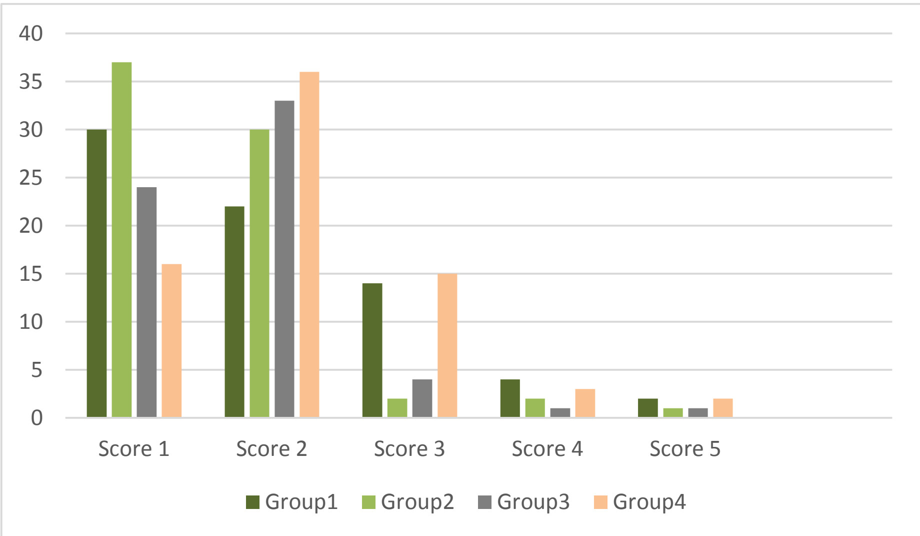


Figure 73: The alignment of understanding for all four groups.

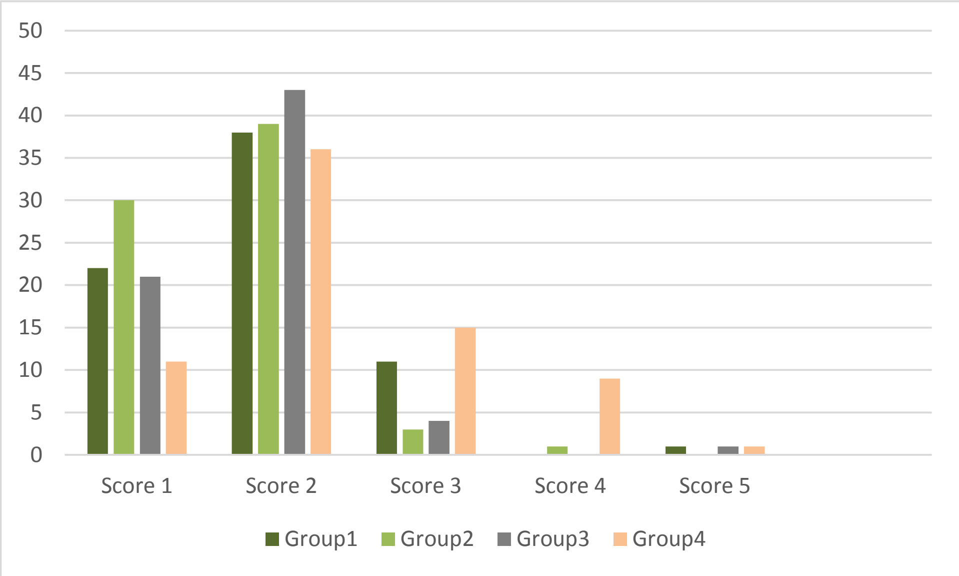


Figure 74: The differences of the ease of use in four groups.

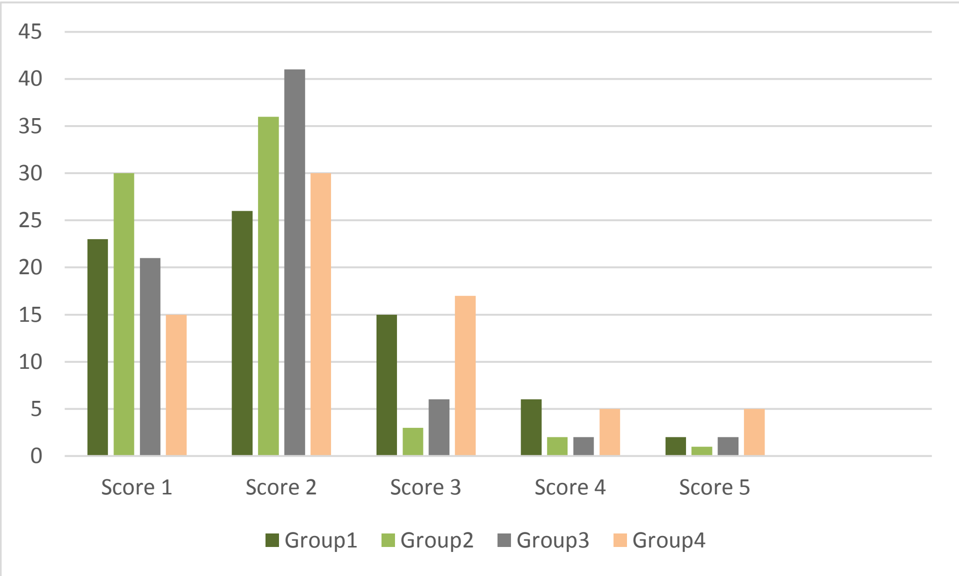


Figure 75: The differences of positive feedback in four groups.

6.4 Conclusion

In this chapter, we evaluate the user interface and the three user-guided disambiguation interfaces. According to the average scores from Group 1 to Group 4 in terms of alignment of understanding, ease of use and positive experience, the feedbacks from the participants are generally satisfied. Even in Group 4 whose feedbacks are the worst in four groups, the average scores in three aspects (2.15, 2.35, and 2.38) are all lower than 3, which means the overall performance is better the medium level. Apparently, there are a lot of details or problems regarding to the user interface need to be refined or solved. They will be discussed and addressed in future work.

All in all, the feedbacks of three questions in Interface A is better than another two interfaces (see Figure 76, Figure 77 and Figure 78), especially in Group 2 and Group 3 (Figure 73, Figure 74 and Figure 75). When there are a lot of senses displayed, the alphabet based results would have a potential influence on the participants even themselves haven't realized. I believe, the participants would feel it is clearer and easier to read the alphabet based concepts.

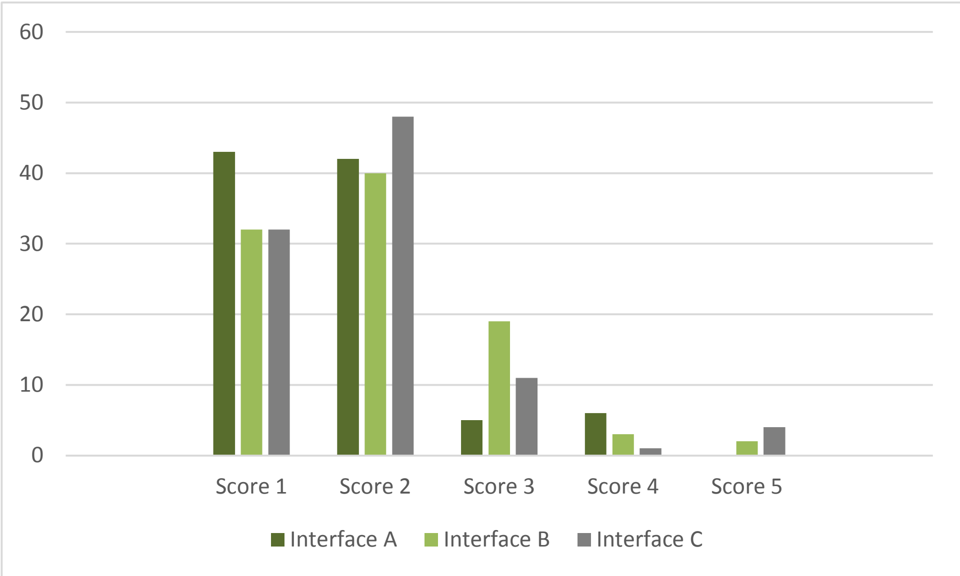


Figure 76: Figure displays the feedback of alignment of understanding in three interfaces.

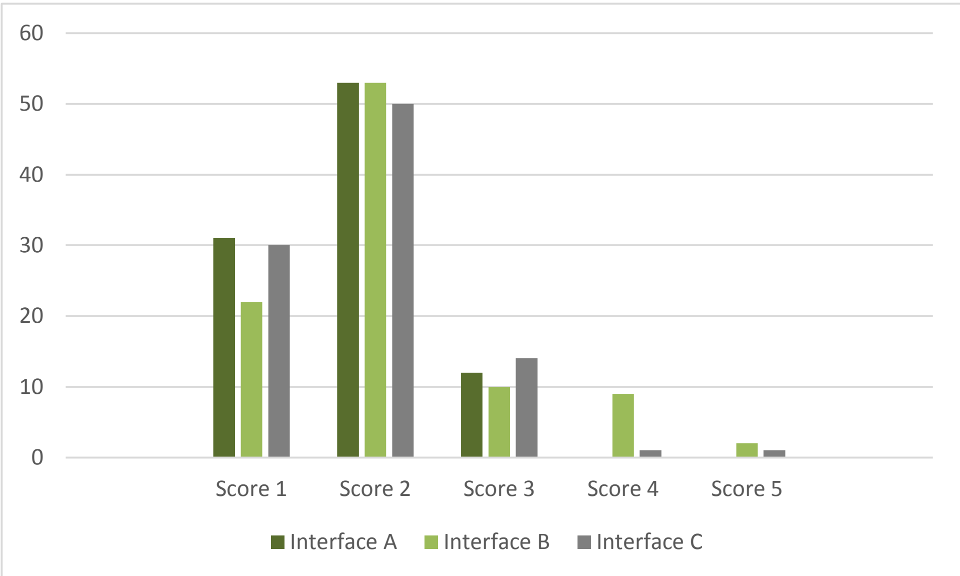


Figure 77: Figure displays the feedback of the ease of use in three interfaces.

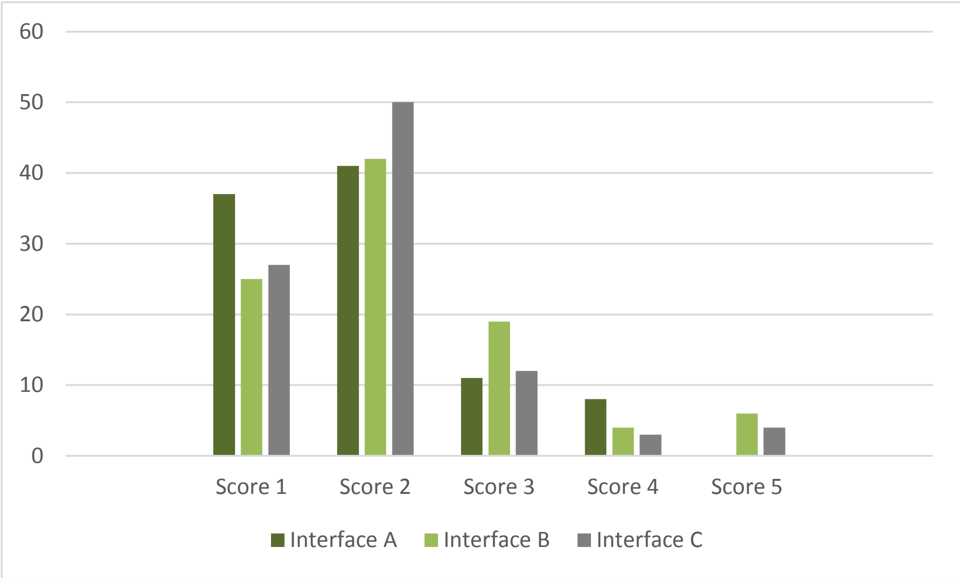


Figure 78: Figure displays the feedback of positive experience in three interfaces.

7 Discussion

This chapter compares the user interface designed for semantic-enhanced search with the interfaces introduced Chapter 3. Also, our approach to disambiguation is compared to other disambiguation methods (Chapter 4) as well. The limitations of our current user interface and related improvement suggestions are proposed. The quality of disambiguation in our system are discussed in this chapter as well.

7.1 Comparison between user interface and related works

As we mentioned in Chapter3, the interfaces presented in related works both have advantages and disadvantages. In this section, some typical interfaces are picked up to compare with the user interface of semantic-enhanced search.

7.1.1 Query/Keyword built

The semantic search query interfaces introduced in related works are mostly very complicated for non-professional end-users. In these query interfaces, it aims to help users to build the semantic language standard queries step by step. However, even if they claimed that the query interfaces introduced in their works are able to reduce the complexity of building the semantic language query (e.g., SPARQL) and the users do not need to have any semantic or ontology knowledge, but most of them are still complex for users to understand how to build the correct queries in given query interfaces (e.g. Broccoli). The query interface described in Structured Interface (see Section 3.2) even needs the users have the background of ontology or semantic web, otherwise they are not able to understand how to manipulate the query interface to formulate a query.

Unlike most query interfaces presented in related works, Mircosearch has the query interface that is as same as the current popular search engines (e.g., Google). Similarly, the query interface designed for Capisco also follows the style of traditional search engine query interface. In both of the two query interfaces, the users are allowed to input the keywords in the search box directly, rather than creating a complicated semantic query. Most users are familiar with the traditional query interface, so the time they spent on understanding and adapting to the new query interface is much less than other complicated query interfaces.

7.1.2 User refinement of the query/keyword

In some user interfaces of related works, the users are allowed to refine the queries that they have built according to their information needs. For example, in Broccoli, the built query is displayed as a tree in the extended breadcrumbs panel, which provides users with the query refinement feature. The nodes in the tree are the user selections from proposal boxes. The users can adjust the query by adding or deleting the nodes in query panel to refine the query. The results are automatically updated according to the change of query. However, in most of the interfaces introduced in related works, the keyword/query cannot be refined or disambiguated by the user (e.g., Microsearch).

In our interface, the user can do the disambiguation themselves as well. After inputting the keyword, the list of identified concepts for user-driven disambiguation is displayed to the users. The users are promoted to choose one or more displayed optional senses according to their information need and search intent. By analyzing the selected senses, the system can "understand" users' intent and provide the related documents.

7.1.3 Search results

Expect the Structured Interface, other interfaces of the related works all introduce the result interface. The results list in theses interfaces are mostly similar to traditional search engine, which contains title and description of the documents. Especially, apart from traditional display of search results, some result interfaces of related works (e.g., Carrot2 and Middlespot) present new solutions for displaying result documents.

Carrot2 provides users the matched documents and the structured category folders for the documents, which classifies the matched documents into different categories. Besides, the result interface offers a circle visualization for the categories. The more portion the category occupies, the more documents this category contains. With the help of this circle, the users are able to have a quick look of all the categories of search results and. By clicking the category in the circle, the relevant documents are displayed to the users. In Middlespot, there are screen shots of candidate web pages are presented and the screenshots can be enlarged when mouse is hovering on it.

Currently, the result interface is not available in our interface. Therefore, we cannot compare the display methods in our interface with others.

7.2 User-guided disambiguation in comparison to related approaches

In our semantic-enhanced search, there are two steps of doing disambiguation: the automatic disambiguation and the user-driven disambiguation. We introduced the disambiguation approaches of related works in Chapter 4 are different from the disambiguation methods that we use here, although some of them have the similarities. In the related works, except Librarian Agent Query Refinement*,* the disambiguation approaches in another two related works both does not involve user query refinement.

7.2.1 Automatic disambiguation

One task of semantic-enhanced search is to identify the input keyword and translate it into relevant concepts. The problem is that the queries that the users input always do not reflect their real information need. Thus, identifying the meanings of the keyword in the given context is the process of automatic disambiguation in our semantic-enhanced search.

In our system, the semantic concepts are captured by the CiC network, which are used to semantically analyze the documents. In the given document, the terms that appear together have the related meaning, so the disambiguation of the term in the document is done by analyzing their related terms. Among the terms used for disambiguation, only those unambiguous terms (refers to only one concept in the given context) are used as anchor pointers for disambiguation. Different from our disambiguation, the disambiguation approach used in Concept based query expansion is query expansion. Concept based query expansion splits a query sentence into separate queries. Each query is processed by using WordNet module. All synonymous concepts of each term in the query are stored in a synset, which is combined into an array. By selecting a synonymous for each of the terms, the new queries are created, which is called query expansion.

7.2.2 User disambiguation

Both *Librarian Agent Query Refinement* and our system includes the user disambiguation. In Librarian Agent Query Refinement, the users are allowed to select a resource from the retrieved results. The resource that are selected by the user reflects the users' information need, which can be used to decrease the irreverent results. Our user-guided disambiguation is similar to that method. In our system, the users are provided with a list of disambiguated candidate concepts. According to their potential intent, they are able to select the concepts which satisfy them. The selection of concepts from users help the system understand their information need and offer users with more relevant documents.

7.3 Limitations and suggested improvements

Although the new interface improves the user experiences and search quality, there are still some shortcoming that should be taken into consideration. In terms of functions, in current interface, the amount of the optional senses determines the quality of the user disambiguation. When a wealth of optional senses are displayed to the users (especially over 100), it is difficult for users to browse all of them, no matter in which disambiguation interface. Therefore, when the amount of results comes to a certain number, the system is better to suggest user to change the search keyword. Also, the meaning of some displayed senses are not easily understood themselves. For example, when “Sad” is typed in as keyword, the users are confused what the optional sense “Sad (Sura)” means. More detailed explanation of each sense is demanded .Besides, instead of forcing users to capitalize the first letter of keyword, the interface should be able to uppercase the first letter automatically.

Regarding to the design of the interface, many details of the interface need to be improved to enhance user experience. For example, according to the feedbacks from the participants in Chapter 6, the behavior of the first-time user should be taken into consideration. Some hints and explanation are needed to help the novice users to be familiar with the interface as soon as possible.

Apart from the help information for new users, some other details needs refinement as well. When there are a large number of optional senses is needed to be displayed to users, the processing time is much longer than normal. It is necessary to add a processing icon or pop up window in the interface, by which it can avoid user misunderstanding of system crashing. What’s more, the color of the pager is too close to the interface background color. Thus, it is possible for users to ignore the pager and then miss other optional senses.

8 Summary and Future work

This chapter concludes the search interface and user-guided disambiguation interfaces that have been done for the semantic-enriched search system Capisco. Also, some possible future works about the interface are discussed. Additionally, we propose some steps for the future research on user-guided disambiguation.

8.1 Summary

With the increasing demand of accuracy and efficiency for search results, the traditional search engines cannot fulfill users' search requirements. On the other hand, the semantic query based search engines are too complicated for users without semantic knowledge to understand and manipulate, even they are able to offer users more satisfied results. Therefore, we introduced the method of semantic-enriched search, which combines some features of traditional text-based search engines and semantic search engines. Our Capisco system implements this approach, which is text-based search engine, instead of on building semantic language queries.

Based on the concepts extracted from Wikipedia, the CiC network is created to capture the semantic concepts and the relationship between them. The documents are semantically analyzed by using the concepts in CiC network and the analyzed results are treated as a concept index, which maintains the links between the concepts contained in CiC networks and documents. The user keyword are identified to relevant concepts firstly, and then these concepts are used to find matching documents by using concept index.

In this thesis, we focus on the user-guided disambiguation which helps system to understand the user information need and search intent. According to the user keyword, the keyword is identified by Capisco and the relevant concepts are listed to the users. The user-driven disambiguation are implemented in this step. We have provided three newly designed disambiguation interfaces for users, by which users are allowed to select the concepts that possibly match their target. The selected concepts are transferred to the search engine to identify matching documents.

The difference of three disambiguation interfaces is the method that used to rank the optional senses. In the first interface, the senses are ranked alphabetically. The second interface orders the optional meanings by decreasing the number of documents. The last interface sorts the possible meanings by their likelihood. In order to observe how three disambiguation interface affect user searching strategies, we designed 16 user experiments and invited 24 participants to finish them. The feedbacks received from the participants showed that the differences of the user performance in three disambiguation interfaces are not very obvious. Generally, the interface which ranks the optional meanings by alphabet performs slightly better than another two interfaces.

8.2 Future work

The user-driven disambiguation interfaces presented in this thesis has improved the accuracy and efficiency of search. Nevertheless, we are aware that some other methods can be combined to user disambiguation. The following steps are recommended for future research on the user-guided disambiguation.

Thematic categories

As we analyzed earlier in this thesis, the display of the optional concepts in user-driven disambiguation stage is a difficult task when the number of the concepts comes to substantial. It is not realistic to expect users to read all the options from the beginning to the end. In this condition, the method used in Carrot2 (see details in Section 3.4) for displaying results brings up a solution. Building the thematic categories and classifying the optional concepts are able to address the problem proposed above. The categories can help users to constrain the target range effectively and quickly, which helps the system to better understand users' target intent. For example, if the users search "Kiwi". There are the categories such as animal and plant are generated. By selecting the category, the possible meanings have been filtered by the users before selecting the senses. Therefore, the users are able to disambiguate the concepts twice to some extent.

Case sensitive

One problem of the current system we proposed before is the case sensitive. The identified concepts of the keyword in lowercase is different from that in uppercase. In current version, the users are asked to uppercase the first letter of the keyword. In the future work, the letter case of the keyword is able to be converted automatically. No matter the keyword is uppercase or lowercase, they are supposed to refer to the same word. However, in some cases, the meaning of the word in uppercase is different from it in lowercase. How to identify these special words for case sensitive is the problem needs to be dealt with.

Spelling correction

In our current system, the keyword is identified completely according to its spelling from users. It is very common that users themselves do not realize that they misspell the keywords. When the users misspell the keyword, the wrong results or even no results are displayed to them. Thus, the correction of misspelling is necessary. The popular search engines like Goggle, are able to provide the users with the suggested word spellings if they recognize the spelling is incorrect. With the help of this functionality, the users are able to find the results when they do not know how to spell the whole keyword.

References

Ali, H., Scholer, F., & Thom, J. A. (2009). "User interaction with novel web search interfaces". Proceedings of the 21st Annual Conference of the Australian Computer-Human Interaction Special Interest Group: Design: Open 24/7 (pp. 301-304). New York, NY, USA: ACM.

Bast, H., Bäurle, F., & Buchhold, B. (2014). "Semantic full-text search with broccoli." *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval* (pp. 1265-1266). ACM, New York, NY, USA

Cambridge Semantics. (2014). *Semantic Search and semantic web*. Retrieved from http://www.cambridgesemantics.com/semantic-university/semantic-search-and-the-semantic-web, last accessed 22/08/2014.

Milne, D., & Witten, I. H. (2013). An Open-source Toolkit for Mining Wikipedia. *Artif. Intell, 194*, 222-239. Doi:10.1016/j.artint.2012.06.007

Faaborg, A., &Lieberman, H. (2006). "A Goal-oriented Web Browser".*Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 751-760). New York, NY, USA: ACM.

Gruber,T. R. (1993). A translation approach to portable ontology specifications. Knowledge Acquisition, 5(2), 199-220.

Guha, R., McCool, R., & Miller E. (2003). Semantic search. *Proceedings of the 12th international conference on World Wide Web*, (pp. 20-24). Budapest, Hungary.

Hinze, A., Taube-Schock, C., Bainbridge,D., Matamua, R., & Downie, J.S (2015). Improving access to large-scale Digital libraries through Semantic-enhanced Search and Disambiguation. *Proceedings of the Joint Conference on Digital Libraries 2015.* Knoxville, TN, UAS.

Horridge, M., Drummond, N., & Goodwin, J., Rector, A., & Wang, H.H (2006). "The Manchester OWL Syntax" (University of Manchester, Manchester, UK). Proceedings of the OWL Experiences and Directions Workshop (OWLED'06) at the ISWC'06.

Hoeber, O., & Xue, D. (2007). "User-Oriented Evaluation Methods for Interactive Web Search Interfaces". *Proceedings of the 2007 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology – Workshops* (pp. 239-243). Washington, DC, USA: IEEE Computer Society.

John, T. (2012). What is Semantic Search and how it works with Google search. Retrieved from http://www.techulator.com/resources/5933-What-Semantic-Search.aspx

Koutsomitropoulos, D. A., Domenech, R. B., &Solomou, G. D. (2011). A Structured Semantic Query Interface for Reasoning-Based Search and Retrieval. *Proceedings of the 8th extended semantic web conference on the semantic web: research and applications* (pp. 17-31). Berlin, Heidelberg: Springer-Verlag.

Lei, Y., Uren, V., & Motta, E. (2006). "SemSearch: A Search Engine for the Semantic Web". *Proceedings of the 15th International Conference on Managing Knowledge in a World of Networks* (pp. 238-245). Berlin, Heidelberg: Springer-Verlag.

Mika, P. (2008).Microsearch: An Interface for Semantic Search. *Proceedings of the Workshop on Semantic Search SemSearch 2008 at the 5th European Semantic Web Conference ESWC 2008*(vol. 334).Tenerife, Spain: CEUR Workshop Proceedings.

Hoeber, X.-D. Yang, and Y. Yao.: Conceptual query expansion. In Proceedings of the Atlantic Web Intelligence Conference(2005)

Horridge, M. (2010). Ontogenesis. Retrieved from http://ontogenesis.knowledgeblog.org/88#wptoc_0_1_3, last accessed 23/08/2014.

Stojanovic, N., Studer, R., & Stojanovic, L. (2004). An Approach for Step-By-Step Query Refinement in the Ontology-Based Information Retrieval. *Proceedings of the 2004 IEEE/WIC/ACM International Conference on Web Intelligence* (pp. 36-43). Washington, DC, USA: IEEE Computer Society.

W3C Working Draft. (2006). *SPARQL Query Language for RDF*. Retrieved from http://www.w3.org/TR/2006/WD-rdf-sparql-query-20061004/, last accessed 09/08/2014.

W3C Recommendation. (2004). *RDF/XML Syntax Specification*. Retrieved from http://www.w3.org/TR/REC-rdf-syntax/#section-Syntax-intro, last accessed 11/08/2014.

W3C Semantic Web. (2014). *Ontology Web Language.* Retrieved from http://www.w3.org/2001/sw/wiki/OWL, last accesses 11/08/2014.

W3C. (2013). *Semantic Web*. Retrieved from http://www.w3.org/standards/semanticweb/, last accessed 19/08/2014.

Witten, I., Gori, M., & Numerico, T. (2007). *Web Dragons: Inside the Myths of Search Engine Technology.* San Francisco, CA, USA: Kaufmann.

Zhang, J., Deng, B., & Li, X. (2009). Concept Based Query Expansion Using WordNet. *Proceedings of the 2009 International e-Conference on Advanced Science and Technology* (pp. 52-55). Washington, DC, USA: IEEE Computer Society.

Appendix A: Material for User Study

A.1 Ethics Approval Letter

Faculty of Computing and Mathematical Sciences
Rorohiko me ngā Pūtaiao Pāngarau
The University of Waikato
Private Bag 3105
Hamilton
New Zealand

Phone +64 7 838 4322
www.fcms.waikato.ac.nz

THE UNIVERSITY OF
WAIKATO
Te Whare Wānanga o Waikato

3 February 2015

Xinning Ren
C/- Department of Computer Science
THE UNIVERSITY OF WAIKATO

Dear Xinning

Application for approval under the Ethical Conduct in Human Research and Related Activities Regulations

I have considered your application for a research project involving human participants entitled "Interface for Semantic-enhanced Search". The project studies how to build user-friendly interfaces for semantic-enhanced search.

The procedure described in your request is acceptable. I note that participants involved in the study will not be identified in publications and/or reports, and that at the conclusion of the project notes will be submitted to the FCMS Data Archive for secure storage for five years.

The Participant Information Sheet, Research Consent Form and Questionaires comply with the requirements of the University's human research ethics policies and procedures.

I therefore approve your application to perform the research project.

Yours sincerely

Mike Mayo
Human Research Ethics Committee
Faculty of Computing and Mathematical Sciences

A.2 Participant Information Sheet

Participant Information Sheet

THE UNIVERSITY OF WAIKATO
Te Whare Wānanga o Waikato

Ethics Committee, Faculty of Computing and Mathematical Sciences

Project Title
Interface for Semantic-enhanced Search

Purpose
This research is conducted as partial requirement for Xinning Ren's Master's research.

What is this research project about?
The project studies how to build user-friendly interfaces for semantic-enhanced search. We explore three interfaces that let users chose between the suggested semantics for a search term (this is called disambiguation).

What will you have to do and how long will it take?
The participants will be given eight search tasks to execute using two or three variations of the Capisco interface. This should take no longer than 30 minutes. After that, the participants will be asked some questions about their experience.

What will happen to the information collected?
The information collected will be used by the researcher to write a research report. It is possible that articles and presentations may be the outcome of the research. Only my supervisor Dr. Annika Hinze and I will be privy to the notes. Afterwards the notes will be kept in the FCMS archive for 5 years. No participants will be named in the publications and every effort will be made to disguise their identity.

Declaration to participants
If you take part in the study, you have the right to:

- Refuse to answer any particular question, and to withdraw from the study before 28/02/2015 / analysis has commenced on the data.
- Ask any further questions about the study that occurs to you during your participation.
- Be given access to a summary of findings from the study when it is concluded.

Who's responsible?
If you have any questions or concerns about the project, either now or in the future, please feel free to contact either:

Researcher:
Name: Xinning Ren
Department: Mathematical and Computer Science, the University of Waikato
Phone: 0220877951
Email: Xinning.ren@gmail.com

Supervisor: *(if applicable)*
Name: Dr Annika Hinze
G 2.26, Department of Computer Science, the University of Waikato
Email: annika.hinze@gmail.com

A.3 Research Consent Form

Research Consent Form

THE UNIVERSITY OF
WAIKATO
Te Whare Wānanga o Waikato

Ethics Committee, Faculty of Computing and Mathematical Sciences

Interface for Semantic-enhanced Search

Consent Form for Participants

I have read the **Participant Information Sheet** for this study and have had the details of the study explained to me. My questions about the study have been answered to my satisfaction, and I understand that I may ask further questions at any time.

I also understand that I am free to withdraw from the study before 28/02/2015 or to decline to answer any particular questions in the study. I understand I can withdraw any information I have provided up until the researcher has commenced analysis on my data. I agree to provide information to the researchers under the conditions of confidentiality set out on the **Participant Information Sheet**.

I agree to participate in this study under the conditions set out in the **Participant Information Sheet**.

Signed: ____

Name: ____

Date: ____

Researcher's Name and contact information:
Name: Xinning Ren
Department: Mathematical and Computer Science, the University of Waikato
Phone: 0220877951
Email: Xinning.ren@gmail.com

Supervisor's Name and contact information: (if applicable)
Name: Dr Annika Hinze
G 2.26, Department of Computer Science, the University of Waikato
Email: annika.hinze@gmail.com

Appendix B: Example of Experiment 1

Experiment 1

Setup: You are interested to find out about Fruit Guava. You enter the search term “Guava”.

1) Did you successfully assign the appropriate concept/sense? [experimenter]

2) Likert scales from 1 to 5:
   a) Did the system behave in the way that you expected it to be?
      1) exactly as I expected
      2) pretty much as I expected
      3) kind of as I expected
      4) quite different as I expected
      5) totally different as I expected

   b) What do you think about the searching processing in this system?
      1) Very easy to use
      2) Easy to use
      3) Medium to use
      4) Hard to use
      5) Very hard to use

   c) Are you satisfied about the searching results?
      1) very satisfied
      2) satisfied
      3) Medium
      4) Not satisfied
      5) Very not satisfied

3) What did you like about this interface?

4) What would you like to change about the interface?