Ontology-Based Semantic Reminiscence Support System

Lei Shi

Cardiff School of Engineering

Cardiff University

A thesis submitted in fulfilment of the requirements for the degree of

Doctor of Philosophy

September, 2012

This thesis is dedicated

To

My parents

My wife

Declaration and Statements

DECLARATION

This work has not previously been accepted in substance for any degree and is not concurrently submitted in candidature for any degree.

Signed ..... (candidate) Date .....

STATEMENT 1

This thesis is being submitted in partial fulfilment of the requirements for the degree of PhD.

Signed ..... (candidate) Date .....

STATEMENT 2

This thesis is the result of my own independent work/investigation, except where otherwise stated. Other sources are acknowledged by explicit references.

Signed ..... (candidate) Date .....

STATEMENT 3

I hereby give consent for my thesis, if accepted, to be available for photocopying and for inter-library loan, and for the title and summary to be made available to outside organisations.

Signed ..... (candidate) Date .....

STATEMENT 4: PREVIOUSLY APPROVED BAR ON ACCESS

I hereby give consent for my thesis, if accepted, to be available for photocopying and for inter-library loans **after expiry of a bar on access previously approved by the Graduate Development Committee**.

Signed ..... (candidate) Date .....

Acknowledgements

The journey towards a PhD degree is long and tough, but it is such a precious experience worth remembering throughout the rest days of my life. Many people accompanied me and gave me help during this journey. For all of them, I owe a very big "thank you":

I sincerely acknowledge my supervisor, Professor Rossitza Setchi, who consistently gave me valuable guidance since the first day I started this research. Without her knowledge and assistance, the thesis would have never been possible. I also would like to thank her for the inspiration and enthusiasm during this period.

I would like to thank the members of Knowledge Engineering Systems Group, Cardiff School of Engineering, for their kind support and feedback on my research.

Finally, I dedicate this thesis to my family, especially my parents. Without their unconditional love and support, my dream cannot come true. I would like to thank my spouse, LIU Xiao (Libby), for her personal support and great patience through this journey. No matter how many ups and downs I had, she always helped me with encouragement. This thesis is as much hers as it is mine.

Summary

This thesis addresses the needs of people who find reminiscence helpful in focusing on the development of a computerised reminiscence support system, which facilitates the access to and retrieval of stored memories used as the basis for positive interactions between elderly and young, and also between people with cognitive impairment and members of their family or caregivers.

To model users' background knowledge, this research defines a light weight user-oriented ontology and its building principles. The ontology is flexible, and has simplified knowledge structure populated with semantically homogeneous ontology concepts. The user-oriented ontology is different from generic ontology models, as it does not rely on knowledge experts. Its structure enables users to browse, edit and create new entries on their own.

To solve the semantic gap problem in personal information retrieval, this thesis proposes a semantic ontology-based feature matching method. It involves natural language processing and semantic feature extraction/selection using the user-oriented ontology. It comprises four stages: (i) user-oriented ontology building, (ii) semantic feature extraction for building vectors representing information objects, (iii) semantic feature selection using the user-oriented ontology, and (iv) measuring the similarity between the information objects.

To facilitate personal information management and dynamic generation of content, the system uses ontologies and advanced algorithms for semantic feature matching. An algorithm named Onto-SVD is also proposed, which uses the user-oriented ontology to automatically detect the semantic relations within the stored memories. It combines semantic feature selection with matrix factorisation and k-means clustering to achieve topic identification based on semantic relations.

The thesis further proposes an ontology-based personalised retrieval mechanism for the system. It aims to assist people to recall, browse and re-discover events from their lives by considering their profiles and background knowledge, and providing them

with customised retrieval results. Furthermore, a user profile space model is defined, and its construction method is also described. The model combines multiple user-oriented ontologies and has a self-organised structure based on relevance feedback. The identification of person's search intentions in this mechanism is on the conceptual level and involves the person's background knowledge. Based on the identified search intentions, knowledge spanning trees are automatically generated from the ontologies or user profile spaces. The knowledge spanning trees are used to expand and reform queries, which enhance the queries' semantic representations by applying domain knowledge.

The crowdsourcing-based system evaluation measures users' satisfaction on the generated content of Sem-LSB. It compares the advantage and disadvantage of three types of content presentations (i.e. unstructured, LSB-based and semantic/knowledge-based). Based on users' feedback, the semantic/knowledge-based presentation is considered to have higher overall satisfaction and stronger reminiscing support effects than the others.

Table of Contents

Table of Figures

List of Tables

List of Publications

Shi, L. and Setchi, R. 2012. Enhancing Semantic Representation using User-Oriented Ontology in Information Retrieval. International Journal of Knowledge-Based and Intelligent Engineering Systems. (Invited paper, in press)

Shi, L. and Setchi, R. 2012. Ontology Based Personalised Retrieval in Support of Reminiscence. Knowledge-Based Systems 45(0), pp. 47-61

Shi, L. and Setchi, R. 2012. User-Oriented Ontology-Based Clustering of Stored Memories. Expert Systems with Applications 39(10), pp. 9730-9742.

Shi, L. and Setchi, R. 2012. Unsupervised Semantic Feature Matching in Information Retrieval using User-Oriented Ontology. 16th International Conference on Knowledge-Based and Intelligent Information and Engineering Systems, 10-12 September 2012, San Sebastian, Spain.

Shi, L. and Setchi, R. 2010. An Ontology Based Approach to Measuring the Semantic Similarity between Information Objects in Personal Information Collections. Lecture Notes in Computer Science 6276, pp. 617-626.

Chapter 1

Introduction

1.1 Motivation

The initial vision for effective recording and storing of personal information is developed by Vannevar Bush (1945). His system prototype Memex, is described as "... *a device in which an individual stores all his books, records, and communications, and which is mechanized so that it may be consulted with exceeding speed and flexibility. It is an enlarged intimate supplement to his memory ...*". The essential purpose of Memex is capturing and reusing personal information. It is treated as a personal memory extender that collects person's memories and experiences. Vannevar highlights the information management mechanism of human's brain, i.e. "... *with one item in its grasp, it* (human's mind) *snaps instantly to the next that is suggested by the association of thoughts, in accordance with some intricate web of trails ...*". The discovery inspired him to develop a different information management mechanism, which uses associations between the information objects, instead of rigid hierarchical structures.

Due to the technical limitation in the early period, Vannevar was not able to provide a detailed specification of the system matching technical standards expected nowadays. However, the well-established fundamental features, e.g. data storage, interaction mechanisms and information object links, have influenced the development of personal computers, hypertext and World Wide Web. More importantly, his vision significantly influences research on modern databases and personal information management (Gemmell et al., 2002, Jones and Teevan, 2007).

Nowadays, a large amount of personal information, including work related, social and private, is generated throughout a person's life. Advanced database mechanisms

turn the construction of personal data repositories into reality. These personal data repositories could be local based, e.g. on personal computers, tablets, smart phones, or cloud based, e.g. using applications such as Flickr, Facebook, Dropbox. Additionally, the wide use of electronic devices enables automatic personal information capturing, which further facilitates customisation and sharing.

In practice, the construction of a reminiscence support system like Memex, is complicated. As a memory extender, the system should have a robust mechanism with the ability and capability of recording, storing large-scale data and handling heterogeneous data formats. In addition, the reminiscence support system should have facilities to analyse and organise data including fragmented data from distributed resources. From an information management point of view, one main challenge is the management of cumulative personal information, which needs to include the inner data structure of the information, and external factors, such as person's profile, background, preference. The content generation, which should include ad-hoc retrieval and personalisation, is also a major challenge. These challenges are further discussed below.

Automatic processing large-scale data. The data generated by a person's life activities is increasing constantly during his/her life span and could reach large amounts which are difficult to manage. A Microsoft research project, MyLifeBits, provides an estimation of the amount of lifetime personal information (Gemmell et al., 2006). The basic data includes text, audio and some video, and the cumulative data amounts cost more than 100 gigabytes, which could be extremely time-consuming for manual management (shown in Table 1.1).

To record all important life experiences, the recording method needs to capture as much data as possible. As shown in Table 1.2, the size of the captured data exceeds 200,000 gigabytes, and the manual management of such data amount is apparently not feasible. In addition, the increase of personal data amount is not linear, considering the technology developments and people's changing information needs in the future, which means the actual amount could be even larger than the estimations.

Table 1.1 Modest estimation of a person's personal information amount during his/her life span, which records the essential information based on what is seen and heard

Item type	Number	Size (GB)
Email/ Messages	91,266	0.3
Web pages	67,491	5.4
Pictures	41,908	9.0
Doc/ Rtf	13,445	1.2
Other	5,873	0.9
Audio	5,067	12.5
PDF	3,215	5.0
Tiff	2,821	8.1
PowerPoint	1,772	4.6
Video	1,301	62.7
Total	234,159	109.6

Table 1.2 Modest estimation of a person's personal information amount during his/her life span, which has no format limitation, and attempts to capture as much information as possible (Gemmell et al., 2006)

Item type	Daily number	Monthly total (MB)	83 years Life total (GB)
1 MB Books/ reports	0.1	3	3
5Kbyte Emails	100	12	12
100 KB Image scans	5	12	12
75 KB Web pages/docs	100	225	225
100 MB Music (1, compressed CD)	0.1	250	250
1 KB/s Listened audio (low quality)	40,000	1,000	1,000
1 MB Photos (Medium quality)	10	250	250
SenseCam photos (50KB)	1,000	1,250	1,250
2 GB/hr. TV (S-VHS quality)	4	200,000	200,000

Although the information amount nowadays is large, human efforts, such as reading, annotating and classifying, are still significantly required in personal information management. Manual work causes a lot of data being intentionally discarded, even the valuable. Therefore, a prerequisite for building a reliable

reminiscence support system is to have an automatic data management mechanism that can treat and reuse large-scale personal information adequately.

Data analysis and organisation on conceptual level. The management of fragmented and heterogeneous personal information is difficult due to the fact that the contained information objects have many inner connections and ambiguities on semantic level. To organise autobiographic memory and life events, the current methods used are mainly chronological, theme-based (chapters) or use both timeline and chapters (Rubin et al., 1998, Bayer and Reban, 2004, Berntsen and Rubin, 2004, Thomsen et al., 2011). The information objects of a theme may include various data formats, for example, data about "Prince William's wedding ceremony" may contain documents, photos, video clips, etc. Typically, these information objects are stored according to their date, e.g. "29 April 2011", or theme, e.g. "wedding ceremony". The occurrence dates and theme words are the means to recall those events later on. This simplified direct method, however, does not refer to details contained within the information objects. Besides the occurrence dates and theme words, the information objects may contain other concepts, e.g. mention of participants, family members and relations, friends, or other locations. Clearly, these concepts also have certain inner connections with other information objects or themes; for example, information objects containing "*Prince William, Westminster Abbey*" are also related to objects containing references to his grandmother "*Queen Elizabeth II*" or her residence "*Buckingham Palace*". These inner connections are semantic relations which are important in reminiscence support and should also be considered, in addition to the time line and themes.

Involving user's background knowledge. According to one of the definitions "*...knowledge is information combined with experience, context, interpretation and reflection. It is a high-value form of information that is ready to apply to decisions and actions...*" (Davenport et al., 1998). Comparing with knowledge-based data analysis, relatively lower level data analysis mechanisms are usually based on words, syntax, morphology, and other lexical elements. Lack of background knowledge could cause semantic gap, for example, "*Westminster Abbey*" and "*Queen Elizabeth II*" are

not identified as related based on word level data analysis, although there are certain relations between them. For the reminiscence support system, to achieve personal information analysis and organisation on conceptual level, it is necessary to take into consideration personal background knowledge. Such knowledge should be defined as user-oriented, which needs to include essential information for the system's understanding of the individual's significant life activities. Additionally, it should be modelled in a computationally feasible way using a human and machine understandable format.

Personalisation and customisation with content generation. Information retrieval mechanisms aim to provide retrieval results matching users' queries, however, most of them cannot recognise the search intention behind the query words (Teevan et al., 2005). Additionally, these mechanisms hardly consider the differences between users, so that the same query by different users always has an identical retrieval result.

Using the memory identification provided by the user, a reminiscence support system should understand the requirements on conceptual level and provide concepts related results, rather than keyword matched results only. The captured data of all users within a group should be shared, e.g. the memories of family members or members of a specific social hub. Based on their profiles and backgrounds, the system should be able to provide personalised content, rather than the same output to everyone. Furthermore, instead of the ranking based result, i.e. a flat list of information objects, the generated content needs to have a dynamic structure reflecting the semantic relations between the retrieved information objects, and it should be adjustable in order to match different requirements of the reminiscing tasks, e.g. recalling events based on time, themes, participants, locations or any other specific needs.

1.2 Thesis Statement

1.2.1 Hypotheses

In this thesis, the following hypotheses are proposed and examined. The first hypothesis is that users' profile and background, considered important in reminiscence support, can be modelled in a computationally feasible way using ontologies.

Next, statistical approaches to information retrieval and data processing have to be employed in reminiscence support as they are useful for personal information analysis, management and content generation. To achieve the analysis and management on semantic basis, and further bridge the knowledge gap which cannot be addressed by statistical approaches, it is necessary to involve external knowledge.

Finally, user's background knowledge is essential to enable personalised retrieval and dynamic generation of content. It could facilitate the identification of information needs on conceptual level, and provide personalised content to different users from various perspectives.

1.2.2 Aim and Objectives

The overall aim of this research is to propose a semantic reminiscence support system which facilitates the access to and retrieval of stored memories used as the basis for positive interactions between elderly and young, and also between people with cognitive impairment and members of their family or caregivers. The goal is to develop relevant technologies to identify, index and organise personal information, and apply these technologies to retrieve dynamic content according to user's information needs and background.

The individual objectives of this research are as follows:

- To create a conceptual model and architecture of a semantic reminiscence support system that integrates the essential features of systems for personal information management and reminiscence therapy;

- To define an interactive knowledge model with simple, flexible and extendable structure, which handles person's backgrounds;
- To develop a computationally feasible ontology-based approach to measuring semantic similarity;
- To develop a method to enhancing the semantic representation of personal information objects according to user's background knowledge;
- To develop a knowledge-based topic identification and clustering approach which facilitates personal information management on conceptual level;
- To create a personalised retrieval mechanism that provides customised content to users based on their background.

1.3 Thesis Outline

The thesis is organised in the following structure:

- Chapter 2 reviews recent research on mechanisms for reminiscence support including personal information capture, personal information management and reminiscence therapy. This chapter also studies the relations between human memory systems and personal information;
- Chapter 3 reviews advanced enabling technologies such as natural language processing, named entity recognition, information retrieval, personalised retrieval and knowledge modelling. The evaluation methods applied in this thesis are also introduced in this chapter;
- Chapter 4 proposes a conceptual model of the Semantic Life Story Book (Sem-LSB) that includes its definition, characteristics, architectures and definition of an information object;
- Chapter 5 describes an interactive knowledge model, named user-oriented ontology and introduces its definition, building process and an approach to measuring semantic similarity;

- Chapter 6 proposes a semantic feature matching algorithm aiming to enhance the semantic representation of information objects based on user's personal background. Relevant experiments and evaluation are also described;
- Chapter 7 introduces the ontology-based topic identification and clustering approach developed. It uses the user-oriented ontology to automatically detect semantic relations within the stored memories. Relevant experiments and evaluation are also included in this chapter;
- Chapter 8 describes the ontology-based personalised retrieval mechanism and knowledge spanning tree generation approach developed. The mechanism utilises the user's model of and the proposed approach for generating customised content. Relevant experiments and evaluation are also included in this chapter;
- Chapter 9 highlights the contributions and conclusion of this thesis, and outlines further work.

Chapter 2

Personal Information Management and Reminiscence Support

This chapter reviews related work in the area of personal information management and reminiscence support. Section 2.1 introduces personal information capture and management. To understand the relation between personal information and human memory, Section 2.2 then studies semantic memory, episodic memory and autobiographical memory. Section 2.3 examines reminiscence therapy. Section 2.4 summarises this chapter.

2.1 Personal Information Management

Recent research on using personal information in support of person's daily lives studies various topics, including personal information capture and management (Gemmell et al., 2006, Hodges et al., 2006, Blunschi et al., 2007), the development of reminiscence support systems with user-friendly interfaces involving the use of human computer interaction (HCI) technologies (Sellen et al., 2007, Peesapati et al., 2010, Hoven et al., 2012), and also semantic technologies in order to facilitate data analysis and organisation on conceptual level (Cai et al., 2005, Sauermann et al., 2006, Sauermann et al., 2008, Shi and Setchi, 2012).

Personal information capture is an essential task for building personal data repository. Nowadays, the use of personal computers, smart phones, and other mobile devices are considered as the most convenient ways to collect personal information. The information may be generated from various applications on different devices, and stored on either local or remote databases. Recent research projects, e.g. Semantic desktop, iMemex, NEPOMUK (Blunschi et al., 2007, Groza et al., 2007, Sauermann

et al., 2008), focus on information capture from distributed sources. However, information capture through these devices has to be executed by users, as most of the captured data is based on users' input, and the activities, e.g. outdoor and all other types which are not computer-based, cannot be effectively captured. Recent research attempts to address this problem using special designed devices, such as wearable cameras, wearable computers. A well-known effort in this direction is a Microsoft research project called SenseCam (Hodges et al., 2006, Hodges et al., 2011). SenseCam is a portable camera used to automatically record daily activities. It takes a photo every 30 seconds, and records most of the daily activities. Figure 2.1 shows the device and the images captured by it.

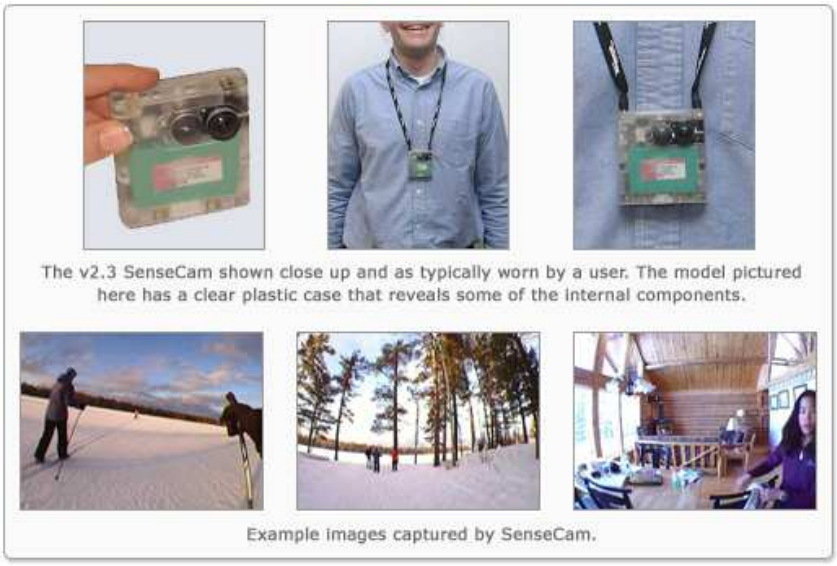


Figure 2.1 SenseCam device and its captured images[1]

However, there are considerable challenges after the information capture stage, such as the analysis and management of the stored data. As pointed out by Jones and Teevan (2007), users tend to lack the necessary attention and management effort in respect of their personal data management. Personal information management (PIM) is an area of research which addresses the problem; it emphasises the management of

[1] research.microsoft.com/sensecam

existing personal data collections, and provides an effective data reuse mechanism. According to its formal definition given by Jones and Teevan, PIM refers to "*... both the practice and the study of the activities a person performs in order to acquire or create, organise, maintain, retrieve, use and distribute the information needed to complete tasks (every day and long-term, work-related and not) and to fulfil various roles and responsibilities (as parent, spouse, friend, employee, member of community, etc.)...*".

Until recently, PIM research primarily focused on the organisation of information in databases. Personal information is generated from people's daily activities using documents, emails, voicemails, entries in address books, calendar events etc., in different forms, e.g. text, image, audio, video etc. Current database mechanisms are not efficient enough to process personal information, and as a result a new generation of databases is required to handle heterogeneous and distributed personal information (Kersten et al., 2003, Dittrich et al., 2008). As stated by Abiteboul (2005), the required advanced features include integration of multi-modal data, information fusion, multimedia query, reasoning under uncertainty and personalisation. An example in this direction is iMeMex, a personal data management system applied to manage complex personal data spaces without much low-level data management effort (Dittrich et al., 2005, Blunschi et al., 2007).

More recent research focuses on the use of semantic technologies in data management and retrieval by considering semantic associations between personal information objects. A prominent development in this direction is the semantic desktop that aims to organise and reuse personal information from different applications on semantic basis. These information objects are connected by following the knowledge structure of selected ontologies, and then they are treated, similar to the semantic web resources, i.e. they can be identified by URIs (Uniform Resource Identifiers), and queried as RDF (Resource Description Framework) graphs (Sauermann et al., 2005, Sauermann et al., 2008). The NEPOMUK project extends these ideas by developing a collaborative environment, which supports personal information sharing and exchange across social and organisational barriers (Groza et

al., 2007). The SeMex system organises personal information objects in a semantically meaningful way by providing a domain model which consists of classes and associations between them (Cai et al., 2005). Based on the logical view of the information, SeMex allows users to browse information by associations and access information composed by integrating personal and public data.

2.2 Semantic Memory and Episodic Memory

Personal information about an individual's daily activities reflects his/her thinking, experience and emotions. Therefore the captured data can be applied to stimulate and rehabilitate the individual's autobiographical memory (Sellen et al., 2007, Hodges et al., 2011). From a psychological point of view, human memory includes five major memory systems that are episodic memory, semantic memory, procedural memory, perceptual representation memory and short-term memory (Tulving, 2002). These memory systems are briefly described below:

- **Episodic memory** (also termed **autobiographical memory**) is a type of long-term memory. It represents the memory of individual's experience in specific events. Information contained in an individual's episodic memory concerns his/her life experience in subjective space and time, e.g. memories from one's wedding day or graduation ceremony;
- **Semantic memory** is a type of long-term memory. It stores knowledge as facts that contain meanings, understandings, concepts and general/specific knowledge, e.g. "*London eye is located on the banks of the River Thames*".
- **Procedural memory** is a type of long-term memory, which refers to the way of carrying out specific activities, e.g. swimming, driving. The creation of procedural memory does not involve conscious control from humans, but relies on external factors, such as procedural learning (consistently repeating a learning process).

- **Perceptual representation memory** is a type of implicit memory, which has connections between semantic memories (Schacter, 1990). The current understanding of perceptual representation memory is still limited. However, it is considered important for category learning, e.g. grouping events or objects into functional categories (Casale and Ashby, 2008).
- **Short-term memory** is the memory of recent/current activities. Related research indicates that the information stored in short-term memory is relatively limited; it is approximately 7 items plus or minus 2 items, e.g. a mobile phone number or email address. Meanwhile, its duration is usually between 5 and 20 seconds. (Miller, 1956, Miller, 2003).

The semantic memory and episodic memory have a dependency relationship. In general, the semantic memory derives from the episodic memory, and the operation of episodic memory always depends on the semantic memory. Tulving (1993) states that there are hierarchical relations between these two memory systems. People gain episodic memory from their life experiences, and this process involves a lot of learning and reasoning. In the course of this process, the episodic memory supports and reinforces the semantic memory. Meanwhile, the existing semantic memory facilitates the learning and reasoning behaviours. Conway and Pleydell-Pearce (2000) have proposed a hierarchical model of an autobiographical memory knowledge base (see Figure 2.2). It maps the episodic memory and semantic memory, and also represents the relationship between general events and event-specific knowledge. An individual's episodic memory contains various themes, e.g. social or work related, which are organised sequentially in mind. Each general event indicates a specific life experience of the individual. As shown in the figure, a general event can be linked to the themes or other events by the associations. In this model, event-specific knowledge is treated as a set of features which can be used to distinguish the general events, i.e. to identify if they are generated by the actual life experience or individual's imagination.

In a reminiscence support system, the captured personal information always relates to the individual's background knowledge, e.g. understanding, thinking and feelings.

The relationship between information objects and background knowledge is similar to the dependency relationship between episodic and semantic memory. The individual's background knowledge contains information on conceptual level that can be used to understand and categorise the information objects. On the other hand, the information object is linked to its related knowledge, which means if there are plenty of information objects, the knowledge structure could be determined by them too.

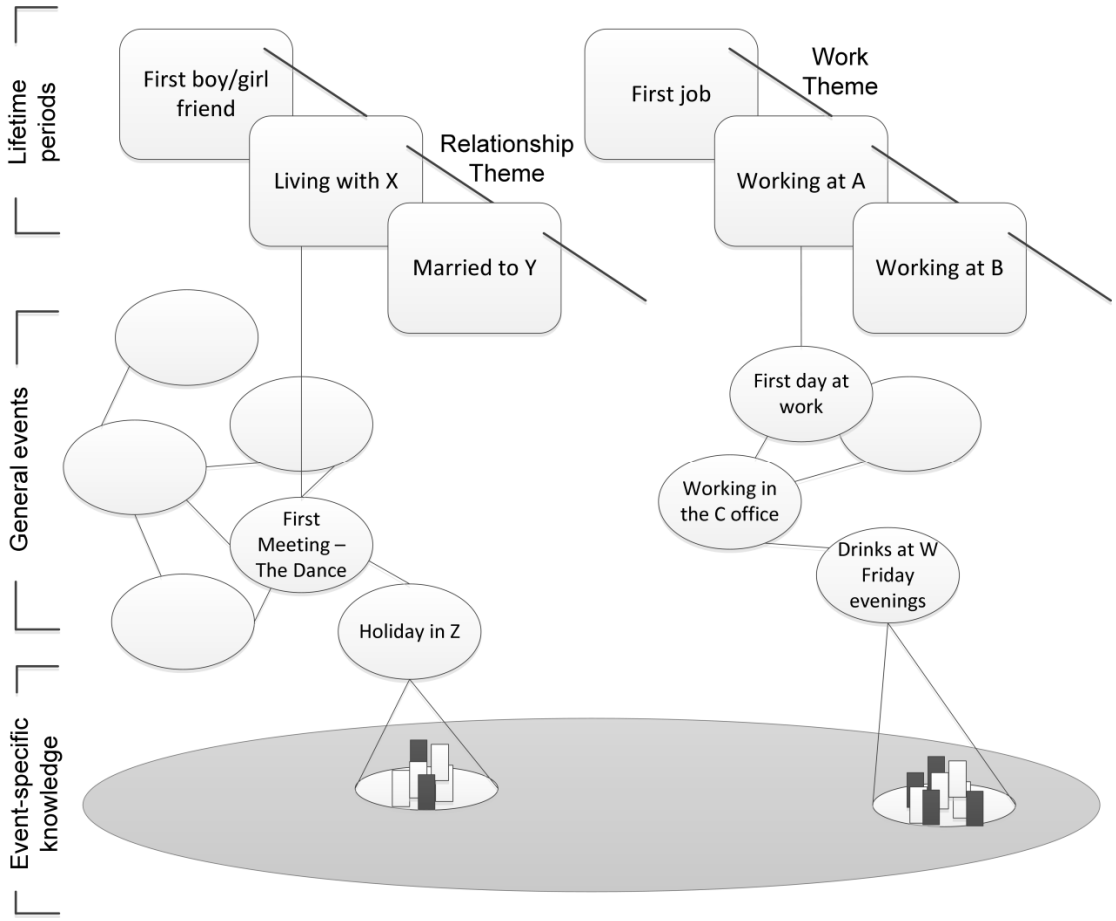


Figure 2.2 Conway and Pleydell-Pearce's model: autobiographical memory knowledge base (2000)

2.3 Reminiscence Therapy

The use of personal information for therapeutic purpose began in the 1960s, and the treatment, called *life review,* utilises reminiscence to benefit elderly people (Butler, 1963). The life review is one-to-one therapy that relies on the face-to-face interactions between an individual and a therapist. In a typical therapy session, the communication is focused on the past experiences of the individual which could be both positive and negative, and the participant is encouraged to review his/her past life events. Artefacts belonging to the individual such as photos, postcards, pieces of music, and video clips can be used as cues during the session. The life review is known to increase the sense of personal identity and motivate the participant (Butler, 1980, Haber, 2006).

Reminiscence is defined as the recollection of significant life experiences, e.g. people, events, thoughts and feelings (Havighurst and Glasser, 1972). As indicated by research, reminiscence enhances positive and diminishes negative experiences (Haight and Burnside, 1993), and can have a positive impact on the memory recall of elderly people including those with memory loss and dementia (Mather and Carstensen, 2005, Woods B et al., 2005). Reminiscence therapy includes various treatments of mental health which are dependent on autobiographical memory (Bluck and Levine, 1998, Serrano et al., 2004). Besides dementia patients, related research indicates that reminiscence therapy also has positive effects on, children and adults, with benefits including improved personal and social adjustment, mental satisfaction and better memory (Ryan and Walker, 1985, Bayer and Reban, 2004, Woods B et al., 2005, Bohn and Berntsen, 2011, Thomsen et al., 2011). Moreover, it was shown that mutual reminiscence could be equally beneficial for both elderly and young people, as it could trigger positive emotions (Pasupathi and Carstensen, 2003, Bryant et al., 2005). The recent trend in the reminiscence therapy is to involve in the session not only the therapist but also caregivers and family members (Thorgrimsen et al., 2002, Haight et al., 2003, Haight et al., 2006). The benefits are increased sense of personal identity,

enjoyable interactions with others, and improved mood (Woods B et al., 2005). In addition, this therapy enables people to pass on their memories to their families and make their mark on the succeeding generations.

Autobiographical memory is a collection of reminders of important events in a person's life, and in reminiscence therapy it is organised as a life story. In practice, the organisation is normally chronological, and memories with negative impacts are not included. The collection is divided into chapters, and each of them corresponds to a particular time period or predefined theme category (Bayer and Reban, 2004). The time periods are later used as hints to help the individuals recall certain events (Conway and Pleydell-Pearce, 2000, Conway, 2005, Thomsen and Berntsen, 2008).

As a useful tool for reminiscing, Life Story Book (LSB) was created in the 1980s to help children in foster care develop a sense of identity and provide them with some personal history (Ryan and Walker, 1985). From information management point of view, a LSB is an information container which facilitates reminiscing by providing individuals with memory problems and/or their family members with means for reviewing and recalling life events. It increases the sense of personal identity and motivates the participant (Butler, 1980). In the context of a LSB, a chapter is a type of high-level memory representation, which includes various memories of a particular time period or theme. A hierarchy model is normally used to organise chapters into categories (Conway and Pleydell-Pearce, 2000, Conway, 2005). From top to bottom, the levels are life story, lifetime periods, mini-narratives and categorised memories. One aim of this model is to reduce the complexity of long and nested chapters. When an individual is asked to recall a specific memory, he/she can start from an important category instead of a brief time period (Thomsen et al., 2011). Figure 2.3 and 2.4 show life story books examples. In general, similar information objects are usually stored together. Each page title represents the theme of the page content, e.g. special days or places for the person. Theme words and dates are important identifications for the information objects.

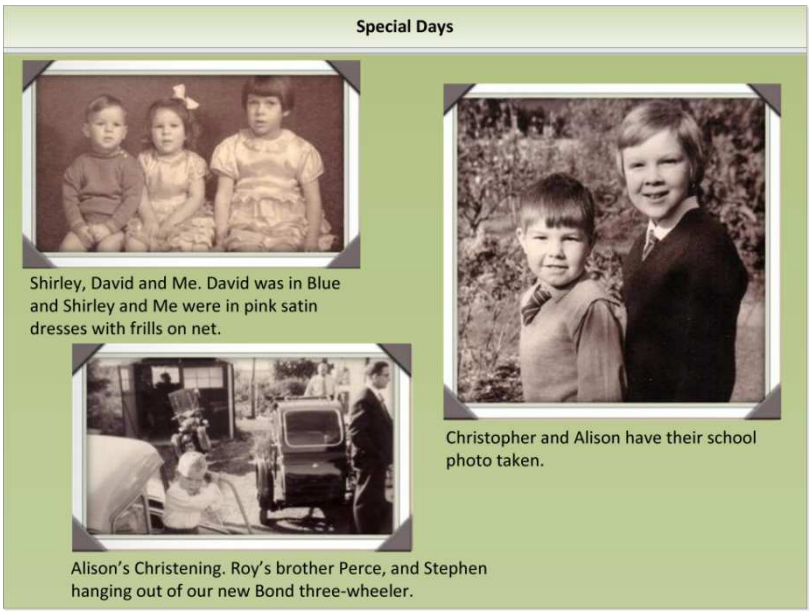


Figure 2.3 Sample of a life story book, information objects within the theme "Special Days" (www.alzscot.org)

Figure 2.4 Sample of a life story book, information objects within the theme "Special Places" (www.alzscot.org)

As indicated by Bayer and Reban (2004), the LSB creation consists of three main steps: (i) Data and materials collection; (ii) Devising captions for materials and data; (iii) Maintaining and updating. These steps are briefly discussed below.

Data and materials collection. The scope of the collection is the person, his/her family members and friends. Data and materials may be virtual or physical, for example, an artefact related to a person's hobby or a postcard from a friend. These data and materials are organised chronologically and related to important persons, places or events in his/her past and present life.

Devising captions for materials and data. A caption is represented by key words or several sentences, for example, the caption of a photo could be "*My Mum Florence was born in Canada 1910*". The caption reflects the content of the photo and includes the name of the participant, her year of birth, birth place and family relation. It can be used as an effective clue for a person to recall memory; it also provides useful information to caregivers and close family members. In real life, a person may need to review some life events related to specific people, e.g. certain family members, but may forget the memory identifications of those life events. The strategy on this issue is to use a chronological series to organise the photographs and relevant information regarding the family members in a LSB. Similarly, if a person needs to review some specific life events, such as the past living places or an important date, a series of photographs and information can be also added in it (see Figure 2.5).

Maintaining and updating. A LSB records a person's important life experience, thus its content needs to be edited and updated along with the occurrence of new significant life events.

Figure 2.5 Chronological photographs and captions about a family member, i.e. (a, b and c); Photographs and captions about an event (d and e) (Bayer and Reban, 2004)

2.4 Summary

This chapter provides the background of reminiscence support. Two main approaches are examined: personal information management and reminiscence therapy. The former approach focuses on information recording and management, and involves automatic personal information capture, heterogeneous/distributed data processing and semantic data management. The latter concentrates on using personal information in support of people's mental health, especially of the elderly and people with dementia. Life story book as an instance of reminiscence therapy records significant life experiences, and then organises the information objects using the timeline and themes, thus providing an interactive mechanism for people to recall and review their life events. The review of the five major memory systems indicates that those related to reminiscence therapy are semantic memory and episodic memory. Semantic memory and episodic memory represent people's background knowledge and personal information objects, respectively. The existing relation between them in human memory indicates that managing personal information objects based on people's knowledge is feasible.

Chapter 3

Information Retrieval and Knowledge Modelling

This chapter reviews related work in the areas of information retrieval and knowledge modelling. The chapter is organised as follows. Section 3.1 reviews related natural language processing technologies, e.g. tokenisation, stop words removal, stemming and POS tagging. Section 3.2 introduces named entity and named entity recognition. Information retrieval models and topic identification are studied in Section 3.3. Section 3.4 focuses on personalised retrieval, and introduces the language model, relevance feedback and query expansion. Section 3.5 describes knowledge modelling approaches including semantic networks, semantic annotations, ontologies and ontology-based similarity measures. Section 3.6 highlights the evaluation methods employed in this thesis. Section 3.7 summarises the chapter.

3.1 Natural Language Processing

Tokenisation

In general, a token refers to a word or phrase. For textual data, tokenisation uses the punctuations and spaces between words to determine the tokens. For example, applying tokenisation to a sentence "*London eye is located on the banks of the River Thames*", it is converted to two phrases, i.e. *London eye*, *River Thames*, and six words, i.e. *is*, *located*, *on*, *the*, *banks*, *of*. Tokenisation is a basic but important task for information retrieval (IR), as the tokens are usually considered as atomic elements of the textual data. For example, the most popular model, *bag-of-words*, treats all documents as a collection of unordered tokens.

Stop Words Removal

Stop words are certain function words and lexical words, e.g. *such*, *so*, *as*, *the*, which lack actual semantic meanings. In IR systems, the stop words increase computing costs, and decrease retrieval precision. The reason is that they frequently appear in most documents, but cannot be used to identify and distinguish these documents. As a result, the stop words are required to be removed. Referring to the previous example, the remaining tokens after the stop words removal are:

{*London eye*, *located*, *banks*, *River Thames*}.

Stemming

Certain words with the same meaning may have different morphologies in natural language, e.g. <*book*, *books*>, <*come*, *came*>. To avoid mismatching, these morphologies should be treated as identical in IR systems. For example, if two documents contain "*book*" and "*books*" respectively, a query including "*book*" is supposed to match both documents.

Stemming is an approach that converts words to their root forms. The related methods of stemming include Porter stemming (Porter, 1980) and snowball stemming. On the previous example, the remaining tokens after Porter stemming are:

{*London eye*, *locate*, *bank*, *River Thames*}.

POS tagging

The term Part-of-Speech (POS) represents different lexical categories of a language such as *verb*, *noun*, *pronoun*, *adjective*, *adverb*, *preposition*, *conjunction* and *interjection*. POS tagging is an automatic mechanism to classify the words according to these categories. It is widely used in lexical analysis, and it also can be an important pre-process in information retrieval and named entity recognition. Back to the previous example, the words with their POS labels are:

(NNP) *London* (NN) *eye* (VBZ) *is* (VBN) *located* (IN) *on* (DT) *the* (NNS) *banks* (IN) *of* (DT) *the* (NNP) *River* (NN) *Thames.*
• NNP - Proper singular noun; • NN - Singular noun; • VBZ - Verb, 3rd ps. sing. Present; • VBN - Verb, past participle; • IN – Preposition; • DT – Determiner; • NNS - Plural noun;

3.2 Named Entity Recognition

Named Entity Recognition (NER), formally introduced by MUC-6 in 1990s (Grishman and Sundheim, 1996), has been widely used in information extraction, information retrieval and question answering. It aims to recognise named entities and classify them based on predefined categories.

The formal definition of *named entity* was given at MUC-7 (Chinchor, 1998), which introduced a category structure with seven categories including *organisations*, *persons*, *locations*, *dates*, *times*, *monetary values* and *percentages*, and words or phrases belonging to any of these categories are considered as named entities. Later on, CoNLL proposed another category structure that includes *persons*, *locations*, *organisations* and *names of miscellaneous entities* (Sang and Meulder, 2003)where the last category, *miscellaneous entities*, is used to cover all the other recognised named entities which cannot be classified into the first three categories. Currently most of the NER tools, e.g. GATE[1], Stanford Named Entity Recogniser[2], LingPipe[3], are all based on these two category structure settings. This research selects GATE as the NER toolkit, and Figure 3.1 shows an example of the result using GATE.

[1] http://gate.ac.uk

[2] http://nlp.stanford.edu/software/CRF-NER.shtml

[3] http://alias-i.com/lingpipe/

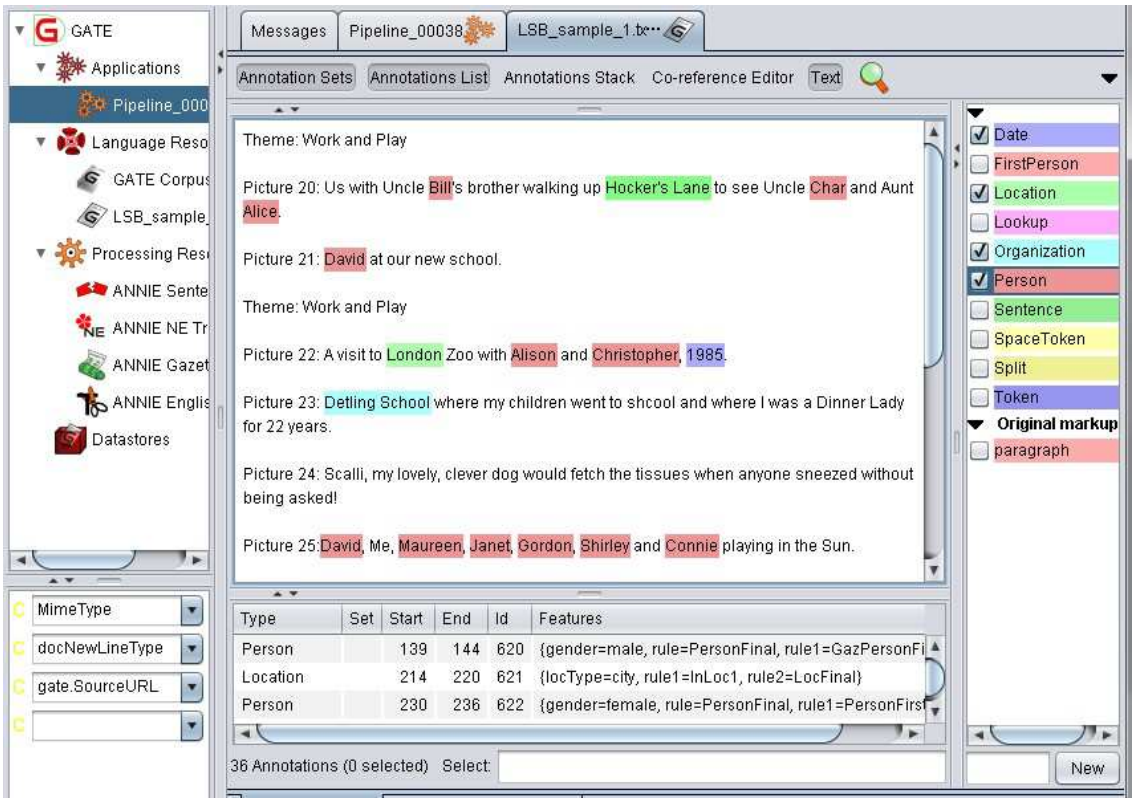


Figure 3.1 GATE, the NER toolkit employed

In practice, the category structure setting is usually scenario dependent, which means those limited categories may not suit some sophisticated NER tasks, e.g. engineering, scientific or medical related. To address this limitation, recent research considers the application of an extended category structure to replace the previous category structure settings. The extended category structure provides a detailed classification for the named entities, which benefits the further processing of information, e.g. content analysis, similarity measure, topic identification, etc. The structure can be produced from various resources. ENE (extended named entity) has a complicated category hierarchy which contains around 200 categories generated from the Japanese encyclopaedia (Sekine, 2008). In addition, the category structure can also be generated from web resources, such as Wikipedia (Cucerzan, 2007, Kazama and Torisawa, 2007), or the Open Directory Project (Whitelaw et al., 2008).

Furthermore, the generic or domain-specific ontology provides additional information about the named entities and their categories, therefore, facilitates NER

and improves recognition accuracy (Sekine, 2008). For example, to recognise the named entity from a personal document collection, an ontology related to the person's family relations and social hub can provide essential cues to reduce the ambiguities, such as aliases, name duplications, etc. For this reason, the proposed system applies user defined domain-specific ontologies to facilitate NER tasks. The extracted named entities are treated as semantic features, and then labelled according to the related ontology topics.

3.3 Information Retrieval

3.3.1 Classical Models

Three classical IR models are widely used today: the Boolean, vector space and probabilistic retrieval model.

The **Boolean model** is based on set theory and Boolean algebra, which allows users to specify their information needs using Boolean operators, i.e. *and* ($\wedge$), *or* ($\vee$), and *not* ($\neg$) (Yates and Neto, 1999, Singhal, 2001). A document d_j is represented as a set of terms, i.e. $d_j = \{t_{1,j}, t_{2,j}, ..., t_{i,j}\}$, then the value of term is set to binary, which indicates if term $t_{i,j}$ is contained or not within document d_j,

$$t_{i,j} = \begin{cases} 1 \; if \; t_i \in d_j \\ 0 \; otherwise \end{cases}. \tag{3.1}$$

The similarity measure between document set is based on the logical conjunction operation,

$$sim(d_i, d_j) = d_i \wedge d_j. \tag{3.2}$$

The Boolean model has several limitations: (i) all terms are equally treated means their weights in the documents are not considered; (ii) for a given query, the retrieved documents have to match the query exactly; (iii) no ranking mechanism is included in this model. At present, the Boolean model is not as popular as the other models, as most users prefer to have the ranked retrieval result with a higher recall. However, it

is still used by some experienced users as they could have more control in the retrieval process.

The **vector space model** (VSM) was introduced by Salton et al. in 1975 to address the limitations and improve on the Boolean model. Using VSM, documents are converted to vectors, and the entries of these vectors indicate the contained terms of the documents. If a term belongs to a document, then its value in the related vector should be non-negative. Most vectors are multidimensional, and their dimensionalities are determined by the number of their contained terms. A vector space is constructed by multiple document vectors, and it can be written in a matrix format,

$$\begin{array}{c} \\ \vec{t}_i^{\,T} \rightarrow \end{array} \begin{array}{c} \vec{d}_j \\ \downarrow \\ \begin{bmatrix} t_{1,1} & \cdots & t_{1,n} \\ \vdots & \ddots & \vdots \\ t_{m,1} & \cdots & t_{m,n} \end{bmatrix} \end{array}, \tag{3.3}$$

where $\vec{d}_j = \begin{bmatrix} t_{1,j} \\ \vdots \\ t_{m,j} \end{bmatrix}$ denotes the vector of document d_j, and $\vec{t}_i^{\,T} = \left[t_{i,1}, \cdots, t_{i,n} \right]$ is the vector of term t_i occurrence in the documents.

Different from the Boolean model, the contained terms of documents are not treated equally in VSM. The importance of a term is measured by its term weight, which considers the impacts of term frequency and document length. For example, if a term appears in millions of documents, its term weight should be relatively low; in contrast, a term appearing in very few documents usually has a high term weight. The reason is that a document could be identified and retrieved easily using the terms it contained with the low occurrences.

A widely used term weight normalisation method in VSM is TF-IDF (Jones, 1972),

$$tfidf_{i,j} = tf_{i,j} \cdot idf_i = \frac{|\, t_i \in d_j \,|}{\max |\, t_m \in d_j \,|} \cdot \log \frac{|\, D \,|}{|\, t_i \in d_n \,|_{d_n \in D} + 1}, \tag{3.4}$$

where $tf_{i,j}$ is the term frequency of t_i in d_j; idf_i is inverted document frequency of t_i in dataset D; $|\, t_i \in d_j \,|$ is the term count of t_i in d_j; $\max |\, t_m \in d_j \,|$ is the highest term count of t_m in d_j; $|\, D \,|$ is the number of documents contained in D; $|\, t_i \in d_n \,| + 1$ is the number of document containing t_i in D with smoothing (to avoid division by zero).

The retrieval process of VSM applies a threshold of similarity between a query and document vectors, to determine which documents can be returned as part of the retrieval result. By comparing the similarity values, VSM can provide a ranking mechanism. Table 3.1 lists the main methods of vector similarity measure:

Table 3.1 Vector similarity measures

Inner product	$sim(\vec{d}_j, \vec{q}) = \sum_{i=1}^{m} t_{i,j} t_{i,q}$
Jaccard coefficient	$sim(\vec{d}_i, \vec{q}) = \frac{\sum_{i=1}^{m} t_{i,j} t_{i,q}}{\sum_{i=1}^{m} t_{i,j}^2 + \sum_{i=1}^{m} t_{i,q}^2 - \sum_{i=1}^{m} t_{i,j} t_{i,q}}$
Cosine similarity[1]	$sim(\vec{d}_i, \vec{q}) = \hat{d}_i \cdot \hat{q} = \frac{\sum_{i=1}^{m} t_{i,j} t_{i,q}}{\sqrt{\sum_{i=1}^{m} (t_{i,j})^2} \cdot \sqrt{\sum_{i=1}^{m} (t_{i,q})^2}}$ $\hat{d}_j = \frac{\sum_{i=1}^{m} (t_{i,j})}{\sqrt{\sum_{i=1}^{m} (t_{i,j})^2}}$

The **probabilistic retrieval model** was introduced by Maron and Kuhns in 1960. It is based on the assumption that each document has a term distribution, and the divergences of theses term distributions are measurable, which reflect the similarities

[1] $\hat{d}_j$ is the normalised vector of d_j.

between the documents. The *probabilistic ranking principle* used in the probabilistic retrieval model (Robertson, 1977, Gudivada et al., 1997) is represented as,

$$P(R \mid d_j, q) = \frac{P(d_j \mid R, q)P(R \mid q)}{P(d_j \mid q)}, \tag{3.5}$$

where $R \in \{0,1\}$; $P(R \mid q)$ is prior probability of the query retrieving a relevant or irrelevant document, and $P(R \mid q) + P(\overline{R} \mid q) = 1$.

The ranking mechanism can be simplified using *odd of relevance*. Let $d_j = \{t_1, t_2, ..., t_m\}$ represent document d_j and its contained terms, q denotes a query, then the odd of relevance between d_j and q is,

$$O(R \mid d_j, q) = O(R \mid q) \cdot \prod_{i=1}^{m} \frac{P(t_i \mid R, q)}{P(t_i \mid \overline{R}, q)}, \tag{3.6}$$

where the result of $O(R \mid d_j, q)$ is monotonic with the probability of relevance between d_j and q.

A well-known approach of the probabilistic retrieval model is Okapi BM25 (Jones et al., 2000). With different datasets, it provides an optimization mechanism by involving adjustable parameters,

$$R(q_i, d_j) = \sum_{i=1}^{n} idf(q_i) \cdot \frac{tf_{i,j} \cdot (k_1 + 1)}{tf_{i,j} + k_1 \cdot [1 - b + b \cdot \frac{|\, d_j \,|}{avlen(d_n)}]}, \tag{3.7}$$

where q_i is the query term; $tf_{i,j}$ is the term frequency of t_i in document d_j; $|\, d_j \,|$ is the length of d_j; k_1 is a positive adjustable parameter, when $k_1 = 0$, this approach becomes to binary; b is another adjustable parameter that can be adjusted based on the length of documents; $avlen(d)$ is the average document length of dataset D. The inversed document frequency in Okapi BM25 is,

$$idf(q_i) = \log \frac{|\, D \,| - |\, q_i \in d_n \,| + 0.75}{|\, q_i \in d_n \,| + 0.75}, \tag{3.8}$$

where $d_n \in D$; $|D|$ is the number of document in D; $|q_i \in d_n|$ is the number of document containing q_i.

3.3.2 Other Models

Multimedia Information Retrieval (MIR) concentrates on the retrieval of multimedia data (Maybury, 1997). Recent MIR systems use keyword annotations to describe the multimedia data, and store and retrieve the data with standard term based indexing and retrieval mechanisms (Bashir et al., 2005). Most of the annotations are still generated manually, making the use of MIR systems be limited in large-scale data repositories. To address this problem, the content-based MIR model is introduced which aims to process and retrieve data based on its content and features such as color, shape and motion for video data, phonemes, pitch and rhythm for audio data (Lew et al., 2006, Casey et al., 2008). Despite the effort made, the content-based approaches still have limitations in terms of the semantic gap between the low level features used by the retireval system and the high level semantic concepts understood by human (Sjöberg et al., 2008).

Semantic Information Retrieval aims to improve the accuracy and address the limitations of term-based mechanisms. The main limitations of term-based mechanisms are described as follows, (i) their similarity measure approaches mainly examine common terms between queries and documents. Due to the complexity of natural language, these approaches could be disordered by the ambiguous meaning of words, then cause mismatching problems, e.g. the ambiguity caused by polysemy and synonymy; (ii) the query is constructed according to user's search intention and backgrounds (Nguyen and Phan, 2008). Although with the same search intention, different users may use different words to describe, and hence the search intention may have various query forms. The search intention behind these query forms is difficult to be identified by term-based approaches, and the retrieval results therefore cannot meet users' requirements on semantic basis.

These limitations can be addressed by integrating semantic technologies, e.g. semantic network, semantic annotation and ontology (Jiang and Tan, 2009). Chen et al. (2010) summarise the objectives of a semantic IR system, (i) it needs to analyse and determine semantic features of data, and then generate semantic annotations (metadata); (ii) to analyse the queries and use semantic extension to produce more semantic features for data retrieval process; (iii) to generate retrieval result according to the features and their related semantic resources. Recent research indicates that the integration of ontology with IR systems can improve the retrieval performance, e.g. precision, recall and ranking (Nagypál, 2005, Vallet et al., 2005). Ontology-based IR also provides additional functions, such as automatic document annotation and query expansion. Moreover, employing external knowledge to generate semantic retrieval results is practically useful in large-scale data retrieval, e.g. geographic information behind server logs improves local search results (Berberich et al., 2011), and RDF enhances web search results (Haas et al., 2011).

However, semantic IR system also has limitations, (i) designing the suitable knowledge base can be time consuming and expensive; (iii) the semantic annotation involved is confined to the knowledge base of the system; (iii) generating the metadata for large-scale data may require high volume of work and it can be difficult to control the quality (Vallet et al., 2005).

3.3.3 Topic Identification

From a mathematical point of view, the similarity of vectors is measured through the angle (cosine similarity) or distance (Euclidean, Hamming and Manhattan) between them. However the similarity measure is not limited to this mathematical interpretation and, depending on the system capability, application scope or user demand, one can employ geographic distances (Liu et al., 2004, Lu et al., 2010, Berberich et al., 2011), strength of family relationships (Shi and Setchi, 2010), latent semantic relations (Deerwester et al., 1990, Hofmann, 1999, Blei et al., 2003) or topic correlation (Blei and Lafferty, 2007). This allows constructing advanced IR systems which can provide different perspectives to the retrieval result. Data clustering, a

useful method in information retrieval, also relies on using similarity measures. The documents in a cluster have consistent conceptual meaning and the assigned label (topic) of the cluster indicates this meaning, which facilitates processing the documents and topic correlations on conceptual level.

To detect the topic of documents on semantic basis, unsupervised topic models rely on analysing variables, such as terms occurrence and co-occurrence, of the documents. Three main algorithms are applied in information retrieval to facilitate topic identification of textual information. These are LSI (Latent Semantic Indexing), pLSI (probabilistic Latent Semantic Indexing) and LDA (Latent Dirichlet Allocation).

LSI (Deerwester et al., 1990) is based on matrix factorisation using singular value decomposition (SVD) to reduce the dimensionality of document vectors. The approach factorises the original term-document matrix into three matrices, two orthogonal and one diagonal, which contain left singular vectors, singular values and right singular vectors, respectively. The singular value indicates the significance of its related singular vectors (left and right), e.g. a greater singular value indicates greater significance. The left and right singular vectors matrices represent the semantic space of terms and documents. In terms of its geometric interpretation, LSI projects documents from a high dimensional space into a lower dimensional semantic space, where terms (documents) with similar meaning have higher probability to be placed on the same dimension. Instead of using the original vectors, the approach detects the latent semantic relation of documents and terms-based on the reduced dimensional projections.

pLSI (Hofmann, 1999) and LDA (Blei et al., 2003) are both generative models. The idea behind them is that the terms contained in a document may represent several topics, and the document is treated as a mixture of topics; the models employed measure the probability of different topics assigned to each document. pLSI uses multinomial distribution to represent the topic-terms relation but it does not provide a satisfying mechanism to model document-topics relations (Hofmann, 1999). It was claimed that pLSI outperforms LSI, however, this was disputed by later research which indicated that the evaluation dataset used was small, and the complexity of

evaluation data was far lower than any real world IR system would have been (Wei and Croft, 2006). Moreover, in some situations pLSI produces overfitting when, for example, the number of variables (topics) of the pLSI model grows linearly with the number of documents in the dataset. It means that the computational cost of using pLSI with large datasets is prohibitive; adding new documents to an existing pLSI model has been also found difficult (Blei et al., 2003).

LDA aims to address the limitations of the pLSI which plays an essential role in statistical topic modelling. It applies Dirichlet distribution to represent document-topics relations, and addresses several over-fitting problems of pLSI, such as growing topic numbers and new document generation. A limitation of LDA is the weak identification of correlated topics (Blei and Lafferty, 2007). In addition, one disadvantage of LDA is the complexity of the model itself as parameter selection directly impacts its complexity. Besides, there is no formal way to set the parameters, which necessitates their empirical selection. Another drawback is that it requires to evaluate the LDA performance with a large data collection, such as web data (Wei and Croft, 2006).

Comparing with term-based IR approaches, these models can partly solve the semantic gap, but they do not, or rarely, use external knowledge, which may lead to limitations in practice.

3.4 Personalised Retrieval

3.4.1 Language Model

Information retrieval (IR) technologies facilitate effective acquisition of content. IR systems such as Google, Yahoo and Bing collect data from public resources, e.g. blogs, newspapers, websites content, and then provide it to their users using similar search interfaces. A widely acknowledged problem is that the user's background is not or only occasionally considered during the search process (Rhodes and Maes, 2000, Bhogal et al., 2007). In practice, the retrieved results should not necessarily be the

same for all users, e.g. the results of the query "dessert" for a child and a diabetes patient probably need to be different, depending on the user's current search intention. Unlike IR systems with general purposes, personalised retrieval aims to provide customised retrieval results to individual users by considering user feedback, browsing behaviour, profile, knowledge, preferences, etc. The implementation of personalised retrieval involves language models (Song and Croft, 1999, Lafferty and Zhai, 2001, Huang et al., 2010), relevance feedback (Salton and Buckley, 1997, Cao et al., 2008, Lv and Zhai, 2009) and query expansion (Xu and Croft, 2000, Collins-Thompson, 2009, Carpineto and Romano, 2012).

The language model is a statistical approach of natural language modelling. It is widely used in machine translation, speech recognition and information retrieval. As a generative model, the language model can be applied to present document generation process. In that process, a document is generated by certain particular terms with a specific sequence, and the generation process is similar to selecting appropriate adjacent term(s) (unobserved) of the observed term(s). The term is selected from a vocabulary, and each selection needs to follow a term probability distribution assigned by the language model. One typical language model is n-gram which is $n-1$ order Markov model. In the n-gram, the selection of an unobserved term depends on its previous $n-1$ observed terms and their sequences. Let $d_j = \{t_{1,j}, t_{2,j}, ..., t_{m,j}\}$ denotes the term set of a document, then the generation process can be represented as,

$$P(t_{1,j}, t_{2,j}, ..., t_{m,j}) = \prod_{i=2}^{m} (t_{i,j} \mid t_{1,j}, ..., t_{i-1,j}) \tag{3.9}$$

The increase of n leads to the exponential increase of the computational complexity, thus the n-gram model is usually simplified to unigram, bigram or trigram. In information retrieval, one common application of language model is automatic spelling correction. It improves the accuracy of user input which prevents IR system from returning irrelevant results. For example, the wrong typed word "thps" is more likely to be corrected to "this" rather than "tops", as the latter one may have a lower occurrence probability in a corpus. Language models can be various. Different language models are used in different application domains or data sources. Query

sense prediction is an application of language model in personalised retrieval. It predicts the sense of queries using resources such as server logs, query logs, web data (Kumaran and Carvalho, 2009, Huang et al., 2010). The basic idea is to treat each query as a short document with i observed term(s) that could be extended by adding j unobserved terms. Based on the i observations, there are few combinations of $i + j$. A language model evaluates the probability of each combination, and then selects the most likely ones. The additional j terms enhance the semantic representation of the original query. For example, a short query "Amazon" has potential adjacent terms "rainforest" and "online store", with each having different senses. Using the query logs of a geography researcher, the language model could assign a higher probability to "rainforest" as the adjacent term of "Amazon", rather than "online store". Therefore, the retrieval result could contain more geographical documents rather than business news about the online retailor.

3.4.2 Relevance Feedback

Relevance feedback is an interactive mechanism that improves the retrieval performance using user's explicit feedback (Salton and Buckley, 1997, Manning et al., 2008). The user marks the result from the initial query as relevant or non-relevant, or adjusts the ranking order of the result according to his/ her expectation. The revised result is treated as a feedback from the query. The system collects the feedback and uses it to improve the next search queries. Relevance feedback facilitates systems optimisation, as the valid feedbacks are accurate and relevant to the specific problems. However, explicit relevance feedback heavily relies on the interactions with users, who have to assess the results of many queries, and any updates of the data could trigger the process again. Practically, most users prefer simpler search interactions, leading to potential refusals to provide feedback (Spink et al., 2000, Manning et al., 2008). The feedback collection problem is addressed in pseudo-relevance feedback. It replaces the manual adjustments by an automatic feedback generation process, which does not involve the users. For a given query, the pseudo-relevance feedback considers the top-n

retrieval results as the feedback for the query. Certain terms are extracted from the feedback and used to reformulate the query. As an enclosed local analysing mechanism, its performance strongly relies on the initial retrieval result, as excess noises contained in the initial feedbacks could generate more irrelevant results for similar queries in further searches. Contrary to this approach, implicit feedback does not rely on explicit feedbacks from the users or retrieval results. It focuses on user's behaviour or the use of external resources, e.g. clicking sequence, dwell time, browsing history, server logs (Joachims et al., 2005, Cheng et al., 2006, White et al., 2007, Liu et al., 2010). The collection cost of implicit feedback is relatively lower as large amount of feedback can be more easily obtained.

3.4.3 Query Expansion

Query expansion is an automatic mechanism for query reconstruction, extension and refinement. In most search sessions, the average length of queries is less than five words (Lau and Horvitz, 1999, Shen et al., 2006). It is difficult to present sophisticated search intention with short queries due to their ambiguity, e.g. a query "blackberry" could indicate a type of fruit or the mobile company. Additionally, the limited number of query terms may cause mismatching, e.g. a query "laptop" may match documents containing "notebook". Query expansion addresses the short query problem. It uses relevance feedback technologies to generate expansion terms, and then the original query is converted to one or several reconstructed queries by adding the new terms (Salton and Buckley, 1997, Cao et al., 2008, Lee et al., 2008). For some IR systems, query logs are used as feedback resources for query expansion by employing information from previous search queries with similar search intention (Song and He, 2010, Bhatia et al., 2011). The performance of this approach depends on the system's usage, as a large quantity of recorded queries could have a high coverage of search intentions. Some query expansion approaches employ lexical knowledge bases, e.g. thesauruses, taxonomies, corpuses, etc. (Voorhees, 1994, Mandala et al., 2000). In the thesaurus-based query expansion, the generation of expansion term is based on the semantic relations between the query term and the

thesaurus content, e.g. a query "laptop" can be expanded with the synonym "notebook", hypernyms "personal computer, PC", or hyponyms "keyboard, CUP, RAM, hard disk". Users are able to refine the result by clicking on the additional words based on their needs. Some thesaurus-based query expansion approaches use modified thesaurus, as manually composed or automatic generated, to achieve the specific search requirements (Jing and Croft, 1994, Xu and Croft, 1996, Beaulieu, 1997, Shiri and Revie, 2006). Compared to the general thesaurus, these thesauruses have smaller scales and narrower application domains, which are specialised on domain specified retrieval. For example, in a commercial IR application, the query "laptop" can be expanded with the hyponyms "DELL, Lenovo, Apple", which can facilitate users locating a range of products.

3.5 Knowledge Modelling

3.5.1 Semantic Networks

Collins and Quillian (1969) suggested that words can be represented as nodes in a tree-structured network. Its related model was widely used to represent the words of taxonomy and associations between the taxonomies. Semantic network is a further developed model based on that idea, and has been widely employed to support learning, data analysis, decision making and knowledge inference (Keil, 1979, Steyvers and Tenenbaum, 2005). It applies graphic notations, e.g. interconnected nodes, arcs, to represent the knowledge (categorised or not) and relations (Sowa, 1991, Sowa, 2006). The knowledge concepts in a semantic network correspond to the nodes, and these nodes are linked together by their inner associations. The structure of a semantic network is determined by its represented knowledge, and it could be simple or complicated. Figure 3.2 shows the structure of WordNet. As a complicated semantic network, it consist of words, taxonomies, word senses, synsets, various relations and other lexcial elements.

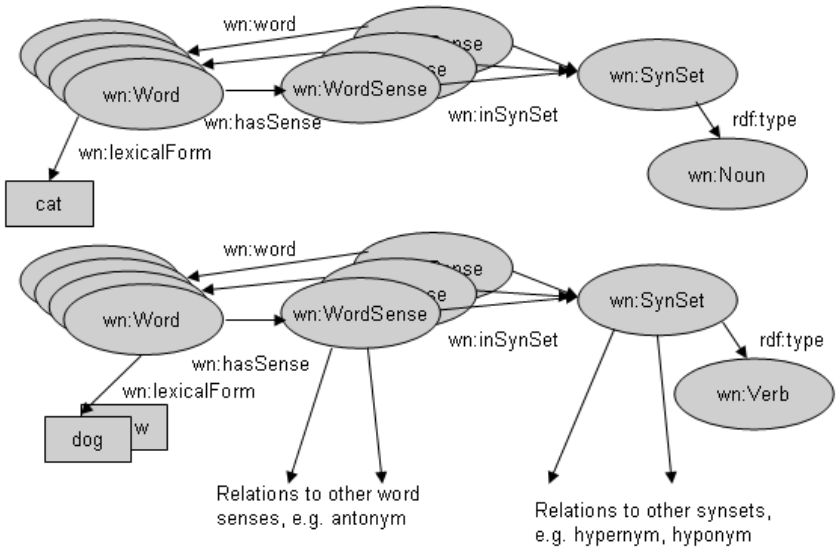


Figure 3.2 WordNet, a semantic network instance[1]

3.5.2 Topic Map

Topic Map is an international industry standard (ISO/IEC 13250)[2] that is proposed for information management and knowledge integration. It contains the following specifications, (i) Data model, (ii) XML syntax, (iii) Canonicalization, (iv) Reference model, (v) Compact syntax, and (vi) Graphical notation. As the core of Topic Map, Data model formally defines topics, associations and occurrences. A single topic can represent one concept or a group of concepts, and sometimes it may be related to a collection of resources by its occurrences. A topic map usually contains a collection of topics, and these topics link to each other by associations. There is no restriction on topics, thus Topic Map is not a domain-specific knowledge model, which also means the contained concepts of Topic Map are heterogeneous. Furthermore, there are no pre-defined categories involved in the topic creation process, thus the hierarchy structure of topics cannot be clearly represented by Topic Map.

[1] www.w3.org/TR/wordnet-rdf

[2] www.iso.org

3.5.3 Semantic Web

Semantic Web as a data model aims to solve the information overload problem within the internet and other large-scale datasets by providing a formal methodology for resource management and utilisation (Berners-Lee et al., 2001). It organises the Web data using semantic theory, which facilitates data analysis, content meaning interpretation and also provides a foundation to establish logical associations between the data (Shadbolt et al., 2006). As a result, data from different resources, e.g. servers, systems, is linked by the logical associations in Semantic Web, and most data has descriptions according to its content. Moreover, the specification of Semantic Web, named RDF (Resource Description Framework), further helps the creation of statements describing different types of resources using URIs (Universal Resources Identifiers) (Downes, 2005). In general, information objects are accompanied by descriptions in Semantic Web, e.g. title, date, creator or other essential information. High level general description is usually attached to a group of information objects, while more specific description works with smaller numbers of information objects, or even individual items.

3.5.4 Semantic Annotation

Semantic annotation aims to generate metadata of information objects. The metadata facilitates data analysis, management and knowledge acquisition, and it also enables extension and creation of new information access methods (Popov et al., 2004, Uren et al., 2006). As an integrated task, semantic annotation involves named entity recognition, relation identification and annotation generation. Meanwhile, its key components include ontologies (with entities and relations), entity identifiers and knowledge bases with entity descriptions (Kiryakov et al., 2004). The semantic annotation generates metadata for named entities including the information of categories, properties, descriptions, notes, etc. Furthermore, the generated metadata enable machines to understand the conceptual meaning of information objects. In

practice, the generated annotation or metadata is various, which depends on the content of information objects and the knowledge domain of applied ontologies.

3.5.5 Ontology

Gruber (1993) defines that "*an ontology as an explicit specification of a conceptualization. ... Ontologies are often equated with taxonomic hierarchies of classes, class definition, and the subsumption relation, but ontologies need not be limited to these forms*" meaning that an ontology is a conceptual knowledge representation, and it can describe a set of concepts in a specific domain and the relations between them. In reality, the ontology concepts are not limited to words, but also could be entities, concept attributes, rules, restrictions or other types of high level information.

From the knowledge coverage point of view, ontologies are classified as generic or domain specific. The former, generic ontology contains domain independent concepts and covers multiple topics, e.g. YAGO2, OpenCyc, so that it has a wider application range. The latter represents the knowledge in a specific domain, e.g. GENIA ontology, Gene ontology, Open biological ontology, and due to the nature of being independent, a domain-specific ontology is hard to be used in other knowledge fields (Suchanek et al., 2008). Considering the functionality, a formal ontology classification was introduced by Guarino (1998) who distinguishes between four types of ontology: top-level ontology, domain ontology, task ontology and application ontology. A top-level ontology attempts to describe domain independent knowledge, where application ontology describes a particular domain or task.

Compared to some other knowledge resources, e.g. thesaurus, taxonomy, an ontology has a higher information capability that provides more comprehensive knowledge coverage. For example, the YAGO2 ontology contains more than 10 million entities and 120 million facts; OpenCyc contains knowledge from various resources, such as WordNet, DBpedia and Wikipedia. In addition, ontologies have well-defined structure that provides sophisticated knowledge representations. For example, the

lexical ontology WordNet has featured structures that are not contained in most thesauruses and taxonomies, e.g. hypernym hierarchies, antonymy of adjectives and verb synsets. Moreover, ontologies consider the conceptual meaning behind words, e.g. "London Eye" and "Tower Bridge" are not identified as synonyms in typical thesauruses, but they could be found as related concepts in a geography ontology.

One of the challenges for many information systems dealing with personalisation in application domains, such as e-learning, decision support, information retrieval, is to choose a formal method to effectively model user's behaviour/knowledge. Recent research indicates that it is beneficial to employ knowledge bases/ontology to organise and manage user's behaviour and knowledge (Gaeta et al., 2011, Razmerita, 2011). The use of a top-level ontology could be computationally expensive as most of the top-level ontologies have complicated structure and contain compound domain knowledge. In contrast, an application ontology concentrates on one domain or a specific task, meaning that it is more suitable for certain information systems with more specific requirements (Shi and Setchi, 2010).

Specific applications often require to build up a suitable ontology. Overall, the ontology creation is difficult and expensive work, as it is time consuming and costs lots of human efforts. For example, the building of an industry design ontology needs to be based on design knowledge (identified by engineering experts) and cognitive studies, and the ontology should include concepts that are related to product functions, performance, material, manufacturing processes, environment, etc. (Li and Ramani, 2007). In general, the ontology building process needs to identify the system's application scope, and then follows on with the system requirements and domain experts' recommendations (Fernandez et al., 1997). Another building method is by ontology integration, which integrates a number of predefined ontologies together to enhance the knowledge capability (Buitelaar et al., 2008).

3.5.6 Ontology-based Semantic Similarity Measures

A prerequisite for using ontology in information systems is to have capability to measure the semantic similarity between ontology concepts. Overall, semantic

similarity measures are classified based on edge counting, information content or features. In the edge counting approaches (Rada et al., 1989), the relations between any two concepts in an ontology are treated as being of type $<is\text{-}a>$ or nothing, and their similarity is determined by the shortest distance between them. A common problem in edge counting is the uniform relation (distance) setting of the concepts. Information content-based (Resnik, 1995, Bollegala et al., 2009) and feature-based (Petrakis et al., 2006) approaches have been proposed to address this problem. Information content-based approaches measure similarity through the amount of shared information between the ontology concepts. This principle however introduces additional computation complexity when the ontology is updated (Sánchez et al., 2012). As the approach ignores the weight of the relations between the ontology concepts, it may potentially lose certain significant semantic information carried by the relations. In the feature-based approaches, the ontology concepts have their related feature sets, which include attributes, properties or glossaries. The feature-based approach uses feature sets to measure the similarity and dissimilarity of the ontology concepts. Sánchez et.al (2012) indicates that the definition of feature set is important in the feature-based approach as it significantly impacts the result of the approach. Table 3.2 shows the main semantic similarity measures based on ontologies,

Table 3.2 Approaches for determining semantic similarity

Rada's distance	$dist(c_i, c_j) = \min len_{edge}(c_i, c_j)$
Wu and Palmer's similarity	$sim_{wp} = \max \frac{2 \cdot dep[LCS(c_i, c_j)]}{len_{edge}(c_i, c_j) + 2 \cdot dep[LCS(c_i, c_j)]}$
Leacock and Chodorow's similarity	$sim_{lc} = -\log \frac{\min len_{synset}(c_i, c_j)}{2 \cdot D}$
Resnik's similarity	$sim_{res}(c_i, c_j) = -\log p[LCS(c_i, c_j)]$
Lin's similarity	$sim_{lin}(c_i, c_j) = \frac{2 \cdot \log p[LCS(c_i, c_j)]}{\log p(c_i) + \log p(c_j)}$

The notations used are as follows: c_i and c_j denote ontology concepts; $len_{edge}(c_i, c_j)$ is the edge length between c_i and c_j ; $len_{synset}(c_i, c_j)$ is the length between the taxonomies/synsets containing c_i and c_j ; $LCS(c_i, c_j)$ represents the lowest common subsume of c_i and c_j , i.e. the most specific ancestor node of c_i and c_j ; $dep[LCS(c_i, c_j)]$ is the length between the ontology root and $LCS(c_i, c_j)$; D is the maximum depth of the taxonomies/synsets (from the lowest node to the top) in which c_i and c_j locate.

3.6 Evaluation Methods

The evaluation tasks in this research examine the retrieval and data clustering performance of the proposed algorithms, in terms of precision, recall, f-score, purity, normalised mutual information and entropy.

Precision and recall are the most common evaluation methods in information retrieval. Let $|retrieved|$ denotes the number of retrieved documents for a query, and $|relevant|$ is the number of relevant documents for the query. Precision and recall are then defined as,

$$precision = \frac{|retrieved| \cap |relevant|}{|retrieved|}, \tag{3.10}$$

$$recall = \frac{|retrieved| \cap |relevant|}{|relevant|}. \tag{3.11}$$

Precision is computed as the number of relevant documents returned by a search divided by the total number of documents retrieved by that search. Recall is the number of relevant documents that are returned by a search divided by the total number of existing relevant documents.

As shown in Table 3.3, states of documents after a search session might be: (i) *true positive (TP),* i.e. the retrieved document is related to the query; (ii) *false positive (FP)*, i.e. the retrieved document is not related to the query; (iii) *false negative (FN),*

i.e. the document is related to the query, but has not been retrieved; (iv) *true negative (TN)*, i.e. the document is not related to the query, and has not been retrieved.

Table 3.3 States of the documents in retrieval output

	relevant	**no-relevant**
retrieved	*true positive (TP)*	*false positive (FP)*
not retrieved	*false negative (FN)*	*true negative (TN)*

According to these states, precision and recall can be alternatively represented as follows:

$$P = \frac{TP}{TP + FP}, \tag{3.12}$$

$$R = \frac{TP}{TP + FN}, \tag{3.13}$$

where $P, R \in [0,1]$. Note that the increase of precision usually causes decrease of recall.

F-score combines precision and recall by computing their weighted harmonic mean,

$$f\text{-}score = \frac{(\beta^2 + 1) \cdot precision \cdot recall}{\beta^2 \cdot precision + recall} \tag{3.14}$$

where β is an adjustable parameter. If $\beta > 1$, F-score prioritises recall; when $\beta < 1$, it emphasises the importance of precision. In most IR evaluations, precision and recall are treated equally, therefore β is set to 1. Balanced F-score[1] is represented as,

$$f\text{-}score_{\beta=1} = \frac{2 \cdot precision \cdot recall}{precision + recall} = \frac{2 \cdot TP}{TP + FP + TP + FN}. \tag{3.15}$$

Purity is an evaluation that measures clustering performance. For a single cluster, it is the maximum number of correctly classified documents within the cluster divided

[1] Experiments in this thesis apply the balanced F-score, i.e. $\beta = 1$.

by the cluster size. Let R denotes the cluster set (result), where $R = \{r_1, r_2, \cdots, r_k\}$, C is the class set, where $C = \{c_1, c_2, \cdots, c_j\}$, and then purity is represented as,

$$purity(R, C) = \frac{1}{N} \sum_k \max_j \mid r_k \cap c_j \mid, \tag{3.16}$$

where $purity(R, C) \in [0,1]$, a higher value of purity indicates a better clustering performance.

Normalised mutual information (NMI) also measures clustering performance. It is based on information theory, and uses mutual information and entropy,

$$NMI(R, C) = \frac{I(R;C)}{[H(R) + H(C)] / 2}. \tag{3.17}$$

$I(R;C)$ is mutual information of R and C,

$$I(R;C) = \sum_k \sum_j \frac{\mid r_k \cap c_j \mid}{N} \log \frac{N \cdot \mid r_k \cap c_j \mid}{\mid r_k \mid \cdot \mid c_j \mid}. \tag{3.18}$$

where $\mid r_k \cap c_j \mid$ is the number of documents in cluster r_k belonging to class c_j; $\mid r_k \mid$ is the number of documents in cluster r_k; $\mid c_j \mid$ is the size of class c_j.

$H(R)$ is the cumulative entropy of clusters by calculating the summation of entropy of each cluster,

$$H(R) = -p(r_k) \log p(r_k) = -\sum_k \frac{\mid r_k \mid}{N} \log \frac{r_k}{N}. \tag{3.19}$$

A smaller entropy value of a cluster indicates the lower uncertainty of it, resulting in better clustering performance. NMI measures the reduction in the uncertainty of a cluster, and the higher NMI value indicates the better performance of the clustering approach.

3.7 Summary

As an important task of IR, natural language processing is applied to remove insignificant words and convert raw data into meaningful tokens. Named entity is a special type of words/phrases with rich semantic information, which could facilitate data analysis and management on semantic level. Its related task, named entity recognition, detects and extracts named entities automatically from documents. The meaningful tokens and extracted named entities can be used to represent the documents using vectors or sets.

Three classical models are widely used in IR: Boolean, VSM and probabilistic retrieval. VSM has several advantages: (i) non-binary term weight, which can be computed using term frequency or other term weight normalisation methods, e.g. TF-IDF; (ii) its mathematical representation is intuitive and clear, i.e. each document is represented as a multidimensional vector in the Euclidean space; (iii) the similarity measure is efficient, e.g. it measures the angle or distance between vectors; (iv) as it does not employ exact matching, it provides partial matching and ranking; (v) the vector representation of documents is not limited to IR, but can also be used for other applications, e.g. data clustering, topic identification; (vi) prior knowledge can be integrated to VSM easily.

Approaches such as the language model, relevance feedback and query expansion are widely used in IR to achieve personalisation. However, these approaches rarely consider the use of external knowledge, which is clearly an aspect to consider.

To bridge the semantic knowledge gap, it is necessary to involve external knowledge. Topic Maps as a portable approach for information management and knowledge integration is based on topics and resource space. As its contained concepts are heterogeneous, Topic Map is not suitable for domain-specific knowledge modelling tasks. Semantic Web is proposed to facilitate large-scale data organisation on semantic basis. It organises information objects by considering their metadata and annotations as descriptions of their content. These metadata and annotations can be generated

automatically or manually. Moreover, as indicated by recent research, ontology as a formal knowledge model is able to model personal knowledge, and address the semantic knowledge gap in personalised retrieval. It has well-defined structure and large information capability that can provide comprehensive or in-depth domain knowledge to IR systems. The similarity between any ontology concepts is measureable which makes ontologies computationally feasible.

This research adopts standard IR and data clustering evaluation methods including precision, recall, balanced F-score, purity, normalised mutual information and entropy.

Chapter 4

Conceptual Model of a Semantic System for Reminiscence Support

This chapter addresses the first objective of this research by proposing a conceptual model of the system for reminiscence support. It is organised as follows. Section 4.1 outlines the traditional process of creating Life Story Books and highlights its limitations. Section 4.2 introduces definitions and describes the system conceptual model and its main characteristics. Section 4.3 presents the system architecture. Section 4.4 summarises the chapter.

4.1 Analysis of Traditional Life Story Books

The Life Story Books (LSB) model consists of three parts: collecting, annotating and maintaining information and materials (Bayer and Reban, 2004). Personal information is collected from the archives of a person and his/her family and friends. The data could be electronic or physical; it is normally organised chronologically and relates to participants, places or events which have played a significant role in a person's life. The second step includes devising annotations, which can be used as cues in memory identification and recall. The final step, maintenance, includes editing and updating of the content as significant life events occur or new material is added to the collection. Due to its predominantly paper-based format, traditional LSBs have three obvious limitations related to their information processing capability, content management and content retrieval.

Information processing capability. The data collection process of LSB involves a lot of manual work including communication, analysis, reading, recording and other interactions, which are time consuming, and prone to human errors. Besides, LSB

stores limited amount of information because of its simple structure and physical format, e.g. being paper-based. It is also difficult to handle large amount of data and materials, e.g. the data can be collected from the whole lifespan of an individual. That fragmented and heterogeneous personal information is usually in different formats and of varied data quality, which is also difficult to be stored/processed in a LSB. These limitations, are addressed in this work by developing a computerised system that integrates databases, knowledge bases and automatic data collection/gathering mechanisms.

Content management. The content management in LSB uses a weak structure and does not benefit from existing semantic associations. The typical authoring strategy of organising personal information is either chronological or based on themes (see Figure 4.1). People mainly use predefined theme categories or a time line to classify and link life events, therefore, the semantic associations between the events are occasionally considered. Due to the weak structure, conjoint relations between events or themes are not utilised. Moreover, if an event does not have a clear time stamp, it could not be organised correctly.

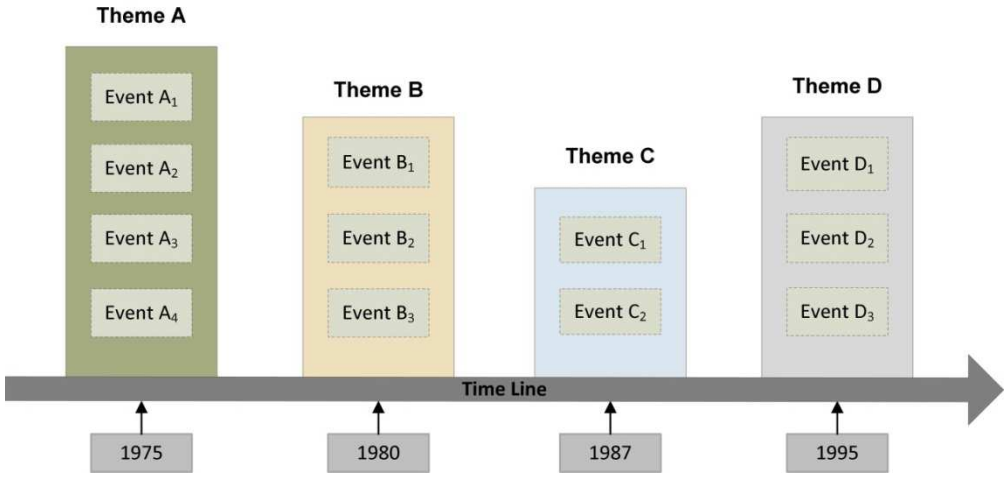


Figure 4.1 Structure of a traditional LSB

In practice, most life events have some semantic associations with each other, and these associations can be easily detected and established by analysing the content. For instance, life events belonging to one theme (e.g. "*children*") may have semantic

associations with others (e.g. "*special days*" or "*special locations*"). Such associations should be considered in the LSB authoring stage. To achieve dynamic content management, some techniques, such as natural language processing, topic identification, automatic indexing can be used to analyse life events, and then automatically detect and establish semantic associations between them.

Content retrieval. The usage of LSB is similar to the use of books, where the user reviews the information sequentially or browses it page by page. As the LSB model does not include any retrieval mechanism, it is difficult to retrieve efficiently and precisely specific content related to users information needs. In other words, this limitation prevents the reuse of the information, thus in reality LSBs always have limited content. These issues can be addressed by employing advanced information retrieval techniques, which provide automatic retrieval with high efficiency and accuracy. More importantly, the automatic retrieval is a foundation of building the large-scale personal information repository, which enables user to record and reuse life events on large scale during his/her life span. Figure 4.2 shows an example of content retrieval using automatic retrieval mechanism and considering semantic associations between the data (a solid line indicates an exact match, and a dash line indicates an established semantic association).

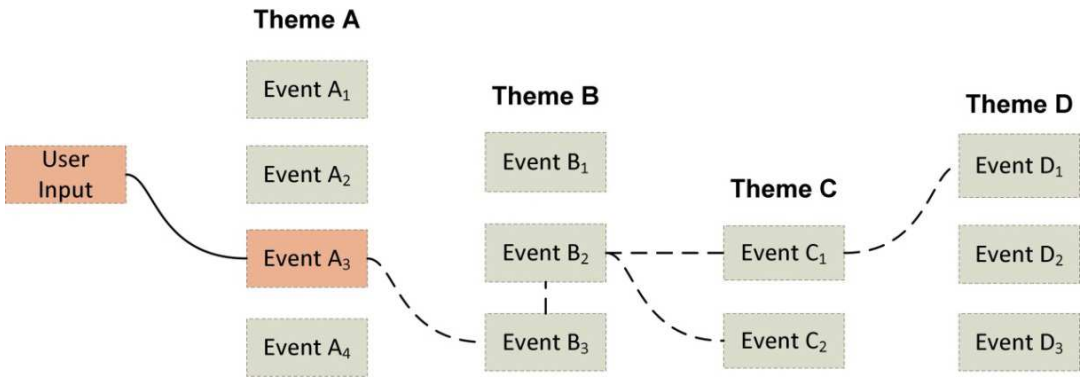


Figure 4.2 Content retrieval based on semantic associations

Based on the discussion above, a summary of the limitations of LSB and the potential improvements is given in Table 4.1.

Table 4.1 Limitations and potential improvements of LSB

	Limitations of tradition LSBs	**Potential Improvements**
Storage Capability	• Very limited.	• Theoretically unlimited.
Structure	• Simple and fixed structure. • Based on time stamp and themes.	• Extendable and flexible structure. • Based on data content, semantic associations and users' knowledge structures.
Information Collecting	• A lot of manual work, e.g. face to face communication, recording, reading, analysing, creating. • Time consuming and expensive.	• Automatic or semi-automatic. • Using natural language processing, topic identification, semantic technologies and knowledge bases.
Content retrieval	• Retrieving from paper-based content. • Poor efficiency and accuracy. • Sequential access to information.	• Using information retrieval technologies and knowledge bases. • High efficiency and accuracy, even data is large-scale. • Customised retrieval result based on user's background.
Maintaining and Updating	• Difficult to reuse or modify the existing content due to the weak structure. • A lot of manual work. • Time consuming and expensive.	• Using data analysis, automatic indexing, topic identification and data clustering. • Manual work is minimised.
Collaboration and Sharing	• Lack of collaborations, e.g. one LSB is only for one individual, the common life experiences cannot be edited together with the participants. • Lack of associations with others' LSBs, e.g. other family members, close friends. • Difficult to share and exchange content.	• Collaborative building is possible. • Access control enables different people to use a common data repository • Advanced information management techniques further facilitate content sharing.
User Interface	• N/A	• User-friendly interface based on psychological research and HCI.

4.2 Conceptual Model

4.2.1 Definitions of Sem-LSB and Information Object

The definition of Sem-LSB (Semantic Life Story Book) is given below,

> **Sem-LSB (Semantic Life Story Book)** is a computerised tool for personal information management and interactive reminiscence support. It uses advanced information techniques, e.g. natural language processing, knowledge-based data analysis/management, personalised retrieval, to provide support in capturing knowledge related to a person's life. It also generates dynamic content with a story-like format which reflects person's needs and interests.

For the purpose of this research, an **information object** is defined as follows:

> An information object is the atomic element of a reminiscence support system. It has no format limitation, but must include a textual annotation with at least one named entity. Each information object is distinguished from or related to other information objects by the named entity/entities included in it.

4.2.2 Main Characteristics

Sem-LSB aims to facilitate personal information reuse and management, address the limitations of traditional LSB and enhance its functionality. The main characteristics of Sem-LSB are: (i) automatic; (ii) knowledge-based; (iii) user's background knowledge-driven; (iv) semantic content management; (v) personalised content generation. These characteristics are discussed below.

Automatic. As mentioned before, the authoring process of traditional LSBs requires a lot of manual work. In contrast, Sem-LSB provides automatic and intelligent assistance in authoring and retrieval.

In the initial stage of building a Sem-LSB, the personal information collected is stored in central databases with unorganised structures, and most of the personal

information objects are discrete on semantic level. Sem-LSB employs advanced information technologies, e.g. natural language processing, information retrieval together with knowledge bases, to achieve automatic indexing, detection of semantic associations, topic identification and data clustering, which enables automatic authoring and retrieval. Therefore, users only need to provide data during the initial process of collecting personal information, and then the system would automatically analyse, index, group and manage the information objects. Furthermore, during the retrieval process, relevant information objects can be automatically retrieved based on queries provided by the user.

Knowledge-based. In reminiscence support, information objects should be organised according to their semantic relations and conceptual meaning.

Knowledge bases related to a person's life experience can be used in Sem-LSB, to help the system understand the content of his/her personal information, predict the conceptual meaning, and detect/establish semantic relations between the information objects. In general, knowledge concepts in a knowledge base usually represent entities and relations, e.g. an ontology "family" includes names of family members, such as "*Elizabeth Alexandra*", "*Charles Philip*", and "*William Philip*". Considering their relations, these entities are organised hierarchically, e.g. "*Charles Philip* $<$*child-of*$>$ *Elizabeth Alexandra*", "*William Philip* $<$*child-of*$>$ *Charles Philip*".

Furthermore, the use of lexical knowledge bases could help with low level data analysis, e.g. natural language processing, cross-language text processing.

In addition, Sem-LSB should be able to integrate various knowledge resources, e.g. generic or domain-specific ontologies, taxonomies, terminologies. For example, if a person's hobby is gardening, a gardening terminology may help analyse his/her pieces of information or life events.

User's background knowledge-driven. Besides the existing knowledge bases, it is necessary to create user-oriented knowledge bases which contain user's own knowledge and personal experiences.

Knowledge is subjective and depends on a person's background, e.g. culture, beliefs, values, insights, intuitions and emotions (Sunassee and Sewry, 2002). However,

most existing knowledge bases are designed for general purposes and do not include personal content. In Sem-LSB, the data analysis and management should consider the differences between the users due to the requirement for personalisation. Moreover, conventional term-based retrieval is not suitable as the dynamic generation process should provide not only terms relevant but also semantically related content.

Semantic content management. It is important to identify the semantic relations between personal information objects, not only to understand and classify them but also to provide the prerequisites for building the system.

To enable semantic content management, named entities (and their categories) should be identified and extracted. The extraction process uses automatic named entity recognition and user-oriented knowledge bases. The knowledge bases provide essential information to facilitate the recognition, classification and labelling of the entities. The relations between the entities need to be considered too. In content management, the semantic relations between the information objects can be established by their contained named entity, which means that the hierarchical structure between the entities in the ontology determines the semantic relations between those information objects. Using the semantic relations, the organisation of discrete information objects on semantic basis can be achieved.

An information object cluster is recognised as a group of semantically related information objects. They are grouped in accordance with their terms, named entities and the semantic relations between them. The cluster implicitly reflects the abstracted meaning (topic) of the information objects contained in it. Based on the topics and contained information objects, relations between multiple clusters can be established and then the degree of similarity between them can be calculated.

By considering the essential factors mentioned above, e.g. knowledge bases, terms, named entities, similarities between information objects/clusters, semantic relations, cluster topics, Sem-LSB can be provided with the capability to manage personal information on semantic level.

Personalised content generation. Sem-LSB provides the efficient access to the stored personal information, and generates dynamic content on user's demand. In

reality, the conceptual meaning behind user's information needs is related to his/her knowledge. Also, users' knowledge structure is constantly changing, which means that the same query by the same person may represent different information needs in different periods. Therefore, the content generation process needs to consider user's background, and involve feedback to handle the user's changing knowledge structure. Moreover, for different users, Sem-LSB should provide personalised results based on their background and relationships between them, e.g. a group of users can share the same collected data, but obtain content generated in a different way.

4.3 Architecture

Figure 4.3 shows the conceptual model of Sem-LSB. It includes modules for the following tasks: pre-processing textual data, extracting terms, recognising entities, detecting relations, modelling knowledge, capturing concepts, indexing, collecting feedback and generating content. The model is split in two parts: (i) authoring; (ii) content generation. These parts are described below.

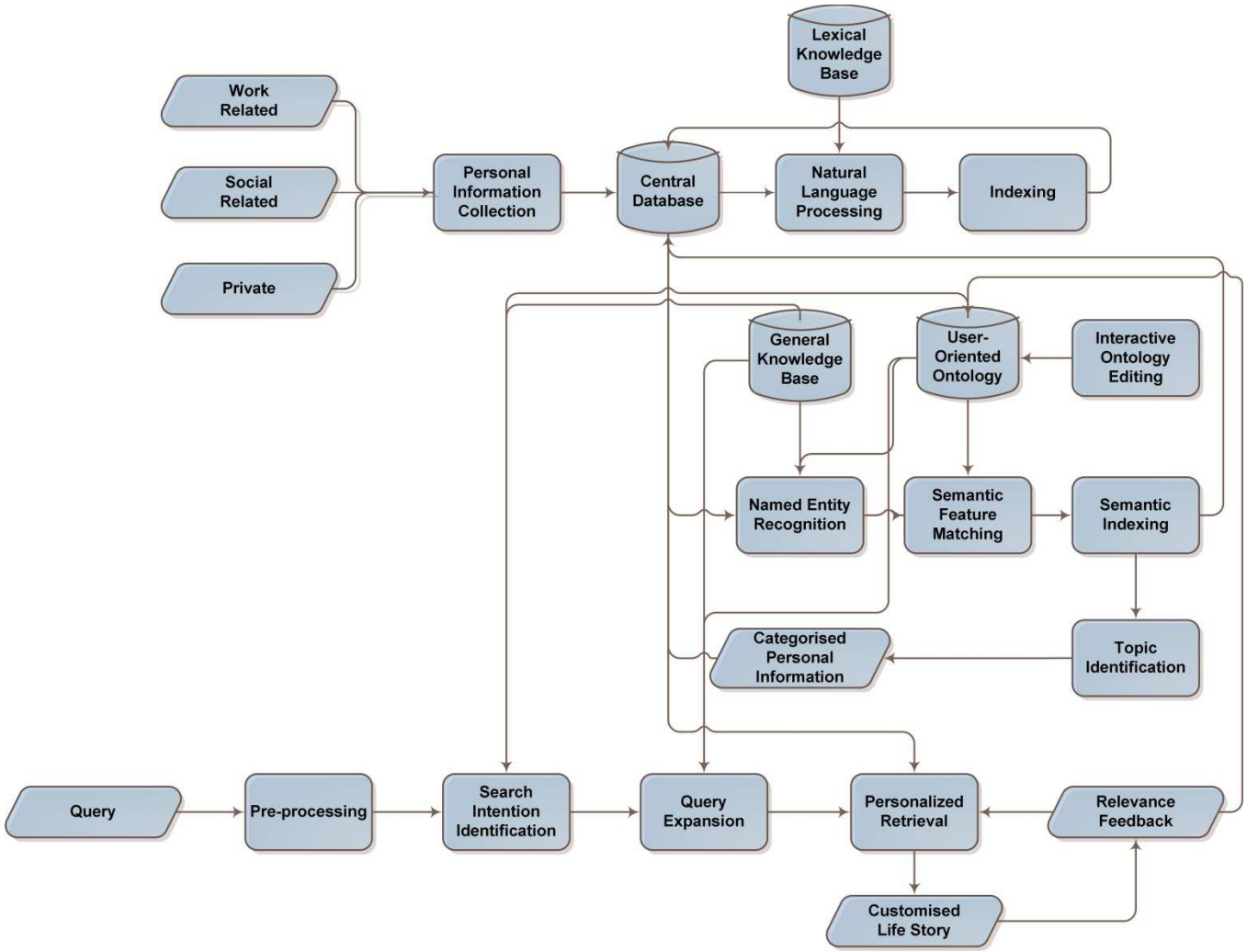


Figure 4.3 Sem-LSB conceptual model

4.3.1 Authoring

The authoring of Sem-LSB includes four stages (Figure 4.4): NLP processing, semantic feature matching, topic identification, indexing and interactive ontology editing.

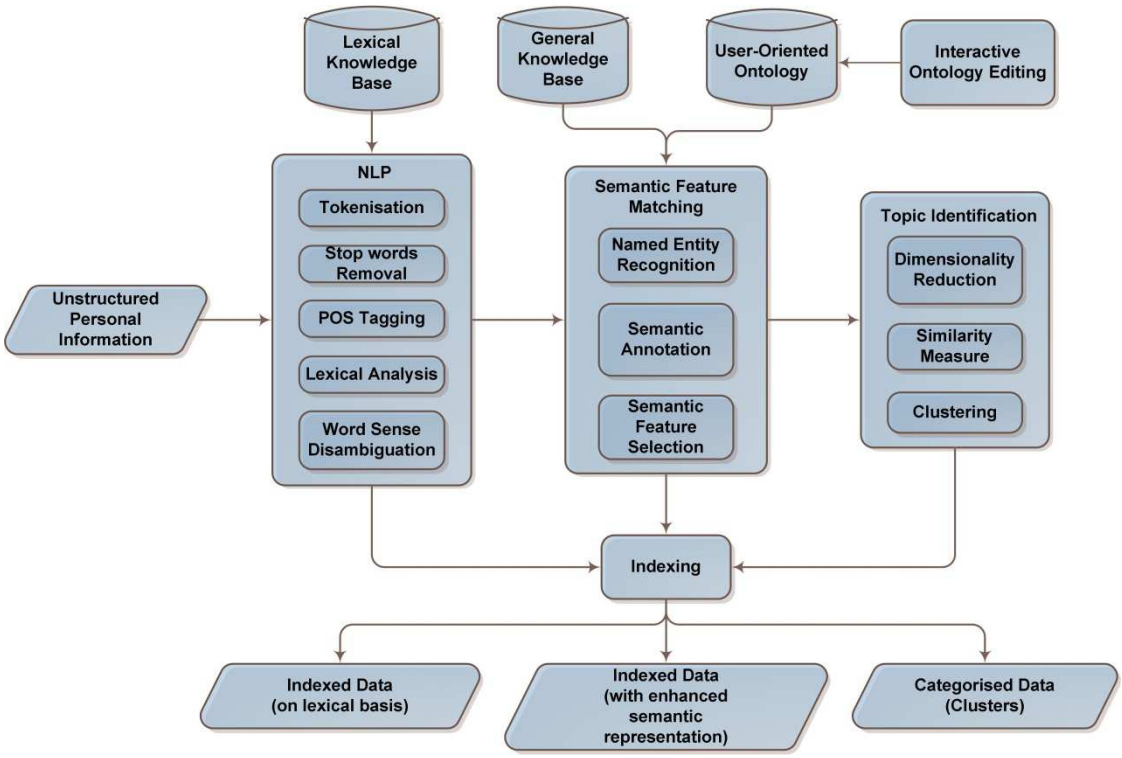


Figure 4.4 Sem-LSB authoring process

The personal information captured is treated as a set of information objects. Each of them is analysed on lexical bases using NLP technologies, e.g. tokenisation, POS tagging, stemming, stop words removal, word sense disambiguation.

The semantic feature matching stage includes two phases. In first phase, named entity recognition (NER) uses knowledge bases to detect the named entities contained within the information objects. Next, to improve the recognition accuracy, semantic annotation is utilised to generate metadata of the uncategorised entities, and it could involve manual processing and human decisions. In practice, certain named entities

may introduce confusion, e.g. if a person called James also use the name "Steve", and each of these names is used in two separate information objects, then the system will fail to recognise and distinguish them correctly using NER only. The semantic annotation is a solution to address this problem by generating annotations of different entities referring to the same object or person, so that NER is able to identify them correctly based on the annotations. In the second phase, the recognised named entities are labelled according to their categories. The semantic feature extraction identifies the significant semantic features (entities) of each information object, and then the semantic feature selection selects more features correlated to the identified features to enhance the semantic representation of the information object (more detail is given in Chapter 6). The knowledge bases provide essential background knowledge to the semantic feature matching, such knowledge could be related to various knowledge domains and be in different formats, e.g. gazetteers or family trees. In addition, the interactive ontology editing module provides personalisation by allowing uses to add, update or delete the knowledge concepts based on their own.

Considering the fact that named entities in one information object often have weak or strong semantic relations with the entities in other information objects, these can be used to identify the topic and degree of similarity between them. The topic identification considers each information object as a combination of terms and named entities. It groups the relevant information objects into different clusters using the named entities (more detail is given in Chapter 7).

In the authoring process, three types of index are generated: based on terms, semantics and categorises. The term-based index does not involve knowledge, so it is only based on normalised term weights. The semantic index however involves knowledge bases and semantic feature matching, thus it is based on the enhanced semantic representation of information objects. The third type uses the semantic index and topic identification, thus it is based on the clustered information objects and the semantic relations between the clusters.

4.3.2 Content Generation

The content generation of Sem-LSB aims to achieve personalised retrieval on user's demand. Figure 4.5 shows its stages: pre-processing, search intention identification, query expansion, personalised retrieval, interactive ontology editing and feedback collection.

Search content is provided by the user using keywords or natural language. At the pre-processing stage, the search content is processed using NLP technologies. Meanwhile, the language model and its related applications, e.g. query suggestion, auto spell correction, are used to facilitate the use of the system.

Search intention identification considers both the terms and named entities contained in the search content, and then identifies the search intention based on the knowledge bases. Query expansion is an important task of the content generation process, which involves semantic feature selection and generation of a knowledge spanning tree. It can extend and reform the original search content according to the identified search intention and relevant knowledge bases.

Personalised retrieval uses similarity between the expanded search content and the indexed information objects. To enable cross-domain search, a user profile space is constructed by integrating multiple user-oriented ontologies. The structure of the user profile space is organised by analysing user's feedback from previous search results. The feedback collection stage helps the user to enhance the content of knowledge bases and optimise the personalised retrieval performance. Explicit feedback reflects user's satisfaction with the retrieved results; thus the user can directly modify the structure, concepts, and relations within the knowledge bases. In contrast, implicit feedback, collected automatically by the system, detects user's behaviour, e.g. dwell time, click sequence, frequently used search content.

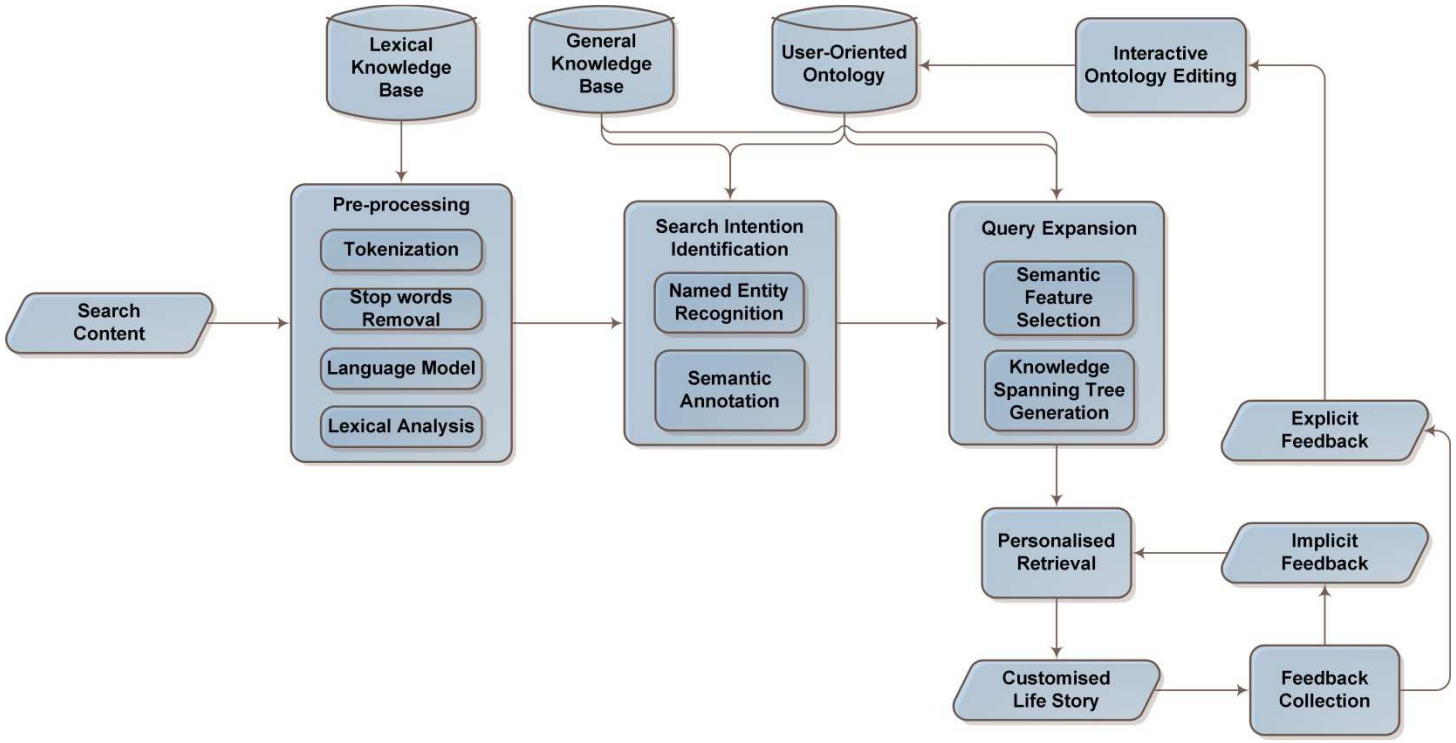


Figure 4.5 Sem-LSB generation process

4.4 Summary

This chapter examines the traditional LSB model, and discusses its limitations and possible improvements. The proposed conceptual model of Sem-LSB is then introduced. As a computerised reminiscence support system, Sem-LSB provides automatic content authoring, content generation and a means to overcome the limitations of traditional LSB. The chapter suggests that advanced information technologies, such as natural language processing, information retrieval, and knowledge bases, can facilitate personal data analysis and retrieval, and provide intelligent assistance to people in managing their personal memories.

Chapter 5

Knowledge Modelling for Reminiscence Support

This chapter introduces a knowledge modelling approach to developing the system proposed in Chapter 4. The chapter is organised as follows. Section 5.1 first discusses potential problems of integrating ontologies with Sem-LSB. It then defines the requirements of a user-oriented ontology and describes its building process. Section 5.2 proposes an approach to measuring semantic similarity which is based on the proposed user-oriented ontology. Section 5.3 summarises the chapter.

5.1 User-Oriented Ontology

5.1.1 Problems of Integrating Ontologies

The integration of Sem-LSB with an existing ontology is problematic due to the following reasons. Firstly, each ontology reflects the structure of its modelled knowledge. Due to the complexity of knowledge in reality, the ontology may have a complex structure and could be of large scale. Moreover, most ontologies are not designed for IR use, therefore the computational cost of using them have to be carefully taken into account. Secondly, a generic ontology is not an appropriate knowledge resource for personalised retrieval, as its knowledge coverage is pre-determined. Each individual user has background knowledge which is different from that of other users. For a large number of users, one ontology cannot provide such high knowledge coverage of all users' backgrounds. Besides, the structure of the generic ontology is fixed and users are not allowed to edit it, as non-solicited modifications could destroy the sophisticated structure designed by the knowledge experts. Therefore, integrating user's background knowledge with the generic ontology

is a difficult task. Finally, the generic ontology lacks interaction with users. As personalised retrieval needs to provide customised results to different users, it relies on well-modelled users' background knowledge. An interactive mechanism can help them in building and editing their own ontologies. These considerations are the reason for proposing an approach, which requires the development of an interactive ontology model.

5.1.2 Requirements

The requirements to a user-oriented ontology are as follows.

- **Interactive building.** A user-oriented ontology involves users in interactive creation and modification. In most cases ontologies are created and maintained by knowledge experts, while a user-oriented ontology expects users to undertake these processes themselves. The personal information belongs to the users, thus they should understand the facts contained better than the knowledge experts. This mechanism lets users apply their knowledge intuitively, and allows them to decide what essential knowledge needs to be included.
- **Simple structure.** The knowledge (features) selection for a user-oriented ontology employs Occam's Razor principle (Koller and Sahami, 1996, Gheyas and Smith, 2010), which means that users need to select a minimum number of significant features to build the ontology, in order to reduce the redundancy and irrelevancy as much as possible. The use of a large number of features in an ontology may cause ambiguity and conflicts. Therefore the benefit of employing this principle is twofold: decreased complexity and improved computational performance.
- **Homogeneous semantic topics.** Named entities and their relations are fundamental elements of a user-oriented ontology. In a single ontology, all entities belong to the same semantic category (topic).

- **Flexible structure**. The purpose of a user-oriented ontology depends on its application scope. It can be light-weight and include a relatively small number of entities and relations.

5.1.3 Building the User-Oriented Ontology

The process includes populating the ontology with information, which is normally part of person's semantic memory, e.g. family tree, important people or events in their lives. Users themselves determine the scope of the knowledge, i.e. who among their family and friends and what topics they want to be included in it. The selection process is completely intuitive. It also enables semantic feature selection, which identifies named entities within the scope of the ontology as semantic features. As mentioned before, typical categories of named entities include organisations, persons, locations, dates, times, monetary values and percentages. In the case of the user-oriented ontology, the categories of named entities could vary according to the user's preferences. For example, all named entities within an ontology representing a person's social hub would belong to category "*people: friends and colleagues*" and they all would be connected via suitable relations. Weight is used to distinguish different types of relations. It represents, using a relative value, the strength of the relations between any two connected named entities.

The semantic representation of a user-oriented ontology is determined by its category. A multi-ontology model can be used to represent a heterogeneous dataset with several topics. A predefined relation set is applied to connect entities from different ontologies. For example, if ontology *people* and ontology *location* contain "*Elizabeth*" and "*Cardiff*" respectively, these named entities could be connected by a relation <*visited*>, if an information object depicting this event exists. In that case, each entity can be mapped to one or more entities from different ontologies. Each information object (often containing more than one named entity) could be mapped through its named entities to the related ontologies: one-to-one, one-to-many, many-to-one and many-to-many. In the single ontology model, the entities are connected

horizontally by relations. In addition to having horizontal relations, the multi-ontology model has vertical connections between entities from different ontologies. This provides a mechanism for representing more complex knowledge structures.

5.1.4 Illustrative Example

Different ontology concepts (entities) are linked different distances (relation weights) that reflect the relevance of the concepts in reality, e.g. <*child-of*> and <*grandchild-of*> have different weights since they indicate different types of relations. Meanwhile, one relation may have various weights, depending on its connected entities, e.g. <*friend-of*> may have different weights reflecting degrees of the friendship. Most relations usually have different weights, however, there are two exceptions: (i) relations with inverse properties are considered to have the same relation weight, e.g. <*parent-of*> and <*child-of*>, and (ii) relations with symmetrical properties are regarded as having unique relation weight, e.g. <*spouse-of*> or <*sibling-of*>.

Figure 5.1 shows a family circle represented as a user-oriented ontology. In the figure, the symbol '•' represents a named entity, and it is treated as a concept in the ontology. The symbol '○' denotes a relation that is applied to connect the entities together. In the example, "*Joy*" and "*Shirley*" are sisters, "*Shirley*" and "*David*" are friends. As seen in the figure, the path distances between each of these pairs are respectively 1 and 2, as "family" (indicated by a solid line) has a stronger link than "friends" (shown as a dashed line). The user-oriented ontology can be treated as weighted undirected graph. In the graph, the concept and relation represent vertex and edge respectively, and the relation weight indicates the edge length. In this thesis, the ontology graph is the foundation of semantic similarity measure.

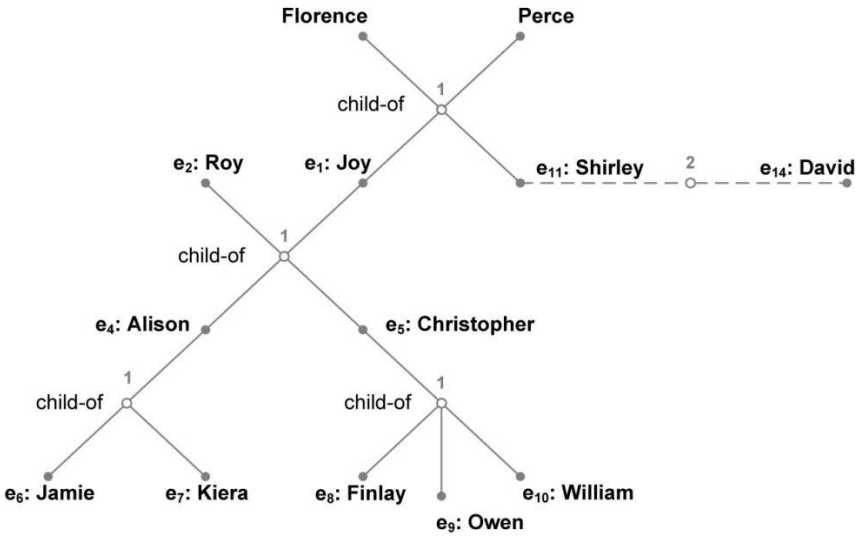


Figure 5.1 An ontology representing a family circle (friends and family)

For the case of multiple ontologies, Figure 5.2 shows two information objects linked to three user-oriented ontologies. The two information objects are: "*20 November 1947: Princess Elizabeth and Prince Philip leave Westminster Abbey after their wedding*", and "*Prince Charles and Princess Diana on the balcony of Buckingham Palace in 1981*". Eight named entities are extracted from the annotations and used as semantic features: "*Elizabeth*", "*Philip*", "*Charles*", "*Diana*", "*Westminster Abbey*", "*Buckingham Palace*", "*1947*" and "*1981*". The relations <*celebrate*> (wedding), <*attend*> (activity) and <*happened-on*> (date) establish the vertical connections and link the three ontologies: *people*, *location* and *time*. The representation of the two information objects as shown in Figure 5.2 enables these information objects to be linked by semantic relations which are preserved by the ontologies.

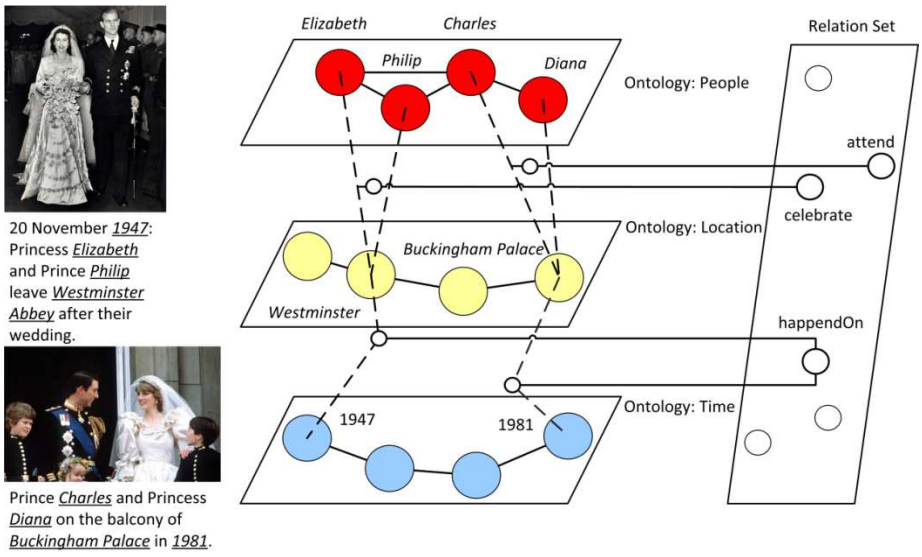


Figure 5.2 Multiple user-oriented ontologies

5.2 Similarity Measure with User-Oriented Ontology

5.2.1 Algorithm

Let D denote a dataset containing information objects, where $D = \{d_1, d_2, ..., d_n\}$. An information object d_j contains one or more named entities, thus its feature set is $d_j = \{e_{1,j}, e_{2,j}, ..., e_{i,j}\}$. Use $O = \{o_1, o_2, ..., o_N\}$ to denote a set of ontologies, where $\{e_{1,j}, e_{2,j}, ..., e_{i,j}\} \cap o_N \neq \varnothing$, and d_j can be written as,

$$d_j = \bigcup_{k=1}^{N} \{e_{1,j}^k, e_{2,j}^k, ..., e_{i,j}^k\}, \tag{5.1}$$

where k indicates a certain ontology.

User-oriented ontologies are used to reduce the semantic ambiguity, structure the named entity categories and detect the semantic relations between the information objects. Moreover, the ontologies are used to measure the semantic similarity between

named entities using the distance between them. The entity similarity between entities e_i and e_j in an ontology is defined as follows (Shi and Setchi, 2010):

$$sim_{onto}(e_i, e_j) = [\log_c(dist(e_i, e_j) + c)]^{-1}, \tag{5.2}$$

where c is the logarithm base, and its empirical setting is e; $dist(e_i, e_j)$ is the distance between e_i, e_j. The similarity of the entities is monotonically increasing as the distance decreases. If there is more than one relation between two entities, the one with the shortest distance should be selected. Based on the example shown in Figure 5.1, *e4: Alison* is a child of *e1: Joy*, and the distance (<*child-of*>) is 1, thus the entity similarity between those two entities is calculated as 0.76; *e6: Jamie* is a child of *e4: Alison*, and the cumulative distance between *e1: Joy* to *e6: Jamie* (<*child-of*> and <*child-of*>) equals to 2, then the entity similarity between "*Joy*" and "*Jamie*" is 0.64. This result shows that "*Joy*" is closer to "*Alison*" (her daughter) than "*Jamie*" (her grandson).

As mentioned above, the entities set $\{e_{i,j}^k\}$ is the feature set of d_j. Let $w(e_{i,j}^k)$ denotes the *entity weight* of $e_{i,j}^k$ in d_j; $|e_{i,j}^k|$ represents the occurrence rate of $e_{i,j}^k$ in d_j; $|e_{i,j}^k \in d_j|$ is the total number of named entities contained in d_j. A high value of the entity weight indicates a greater importance of the entity to the corresponding information object. Formally, the entity weight is:

$$w(e_{i,j}^k) = \frac{|e_{i,j}^k|}{|e_{i,j}^k \in d_j|}. \tag{5.3}$$

For example, if an information object contains two named entities, "*Alison*" and "*Christopher*", both from the same ontology, the weight of these named entities in that information object is $1 \times 2^{-1} = 0.50$.

The cosine similarity is applied to measure the similarity between two information objects d_i and d_j, based on their common terms,

$$sim_{\cos}(d_i, d_j) = \frac{\sum_{k=1}^{n} w(t_{1,i}) w(t_{1,j})}{\sqrt{\sum_{k=1}^{n} w(t_{1,i})^2} \sqrt{\sum_{k=1}^{n} w(t_{1,j})^2}}, \tag{5.4}$$

where n is the number of terms contained in the information objects d_i and d_j.

Named entities with the same topic (from the same ontology) are called *related entities*. Suppose two information objects d_i and d_j contain named entities $e_{m,i}^k$ and $e_{n,j}^k$ respectively, where $e_{m,i}^k$ and $e_{n,j}^k$ is a pair of *related entities*. Let $sim_{related}(d_i, d_j)$ denotes *related entities-based similarity* of two information objects containing related entities,

$$sim_{related}(d_i, d_j) = \sum_k sim_{onto}(e_{m,i}^k, e_{n,j}^k) \cdot w(e_{m,i}^k) \cdot w(e_{n,j}^k). \tag{5.5}$$

If two information objects contain more than one pair of related entities, only the closest pair is considered in order to avoid bias towards objects containing many related entities. Next, the *ontology-based similarity measure* for the general case of having named entities belonging to various ontologies is defined as:

$$sim_{semantic}(d_i, d_j) = sim_{\cos}(d_i, d_j) + sim_{related}(d_i, d_j). \tag{5.6}$$

Considering the related entities between any two information objects can facilitate their similarity measure. The next section illustrates the approach through an experimental study.

5.2.2 Experiment and Evaluation

This experiment uses a person's life story collection from which eight information objects are selected (see Table 5.1). A family tree has been manually created as an ontology to assist with the similarity measure.

Table 5.1 Information objects used in the experiment

d_1	Jackie and me in the playground at the College where Roy first saw me.
d_2	A visit to London Zoo with Alison and Christopher.
d_3	Christopher's Christening. Photo taken on the front lawn. Notting Hill
d_4	The Grandchildren together at the Anniversary: Jamie, Kiera, Finlay, Owen and William.
d_5	Roy took this photo of me when he was on leave. He was not home when I wore this dress for Shirley's wedding.
d_6	I made Alison's wedding dress without a pattern.
d_7	In Mrs Trim's garden when the snow stayed for weeks and weeks. Jim fell in the snow and got frost bite. Cardiff
d_8	David and me with Jim by our back door. Penarth

Two ontologies are used in this experiment, o_1: *family member* and o_2: *location*. The result of the named entity recognition is shown in Table 5.2.

Table 5.2 Named entities with ontology topics

	o_1: person	**o_2: location**
d_1	e_1: Joy; e_2: Roy; e_3: Jackie	n/a
d_2	e_4: Alison; e_5: Christopher;	e_1: London;
d_3	e_5: Christopher;	e_2: Notting Hill;
d_4	e_6: Jamie; e_7: Kiera; e_8: Finlay; e_9: Owen; e_{10}: William;	n/a
d_5	e_2: Roy; e_1: Joy; e_{11}: Shirley;	n/a
d_6	e_1: Joy; e_4: Alison;	n/a
d_7	e_{12}: Trim; e_{13}: Jim;	e_3: Cardiff;
d_8	e_{14}: David; e_1: Joy; e_{13}: Jim;	e_4: Penarth;

The named entity weight matrix of the information objects is shown in Table 5.3. For example, the entity weight of e_6: *Jamie* contained in d_4 is 0.20 because this information object contains four more named entities, thus $w(e_6^1) = 1 \times 5^{-1} = 0.20$.

Table 5.4 shows the similarity computed using formula 5.5 based on the weight of the common entities (i.e. without the ontology). For example, both d_1 and d_5 contain entities e_1: *Joy* and e_2: *Roy*, and they both have entity weight 0.33 (see Table 5.3). Applying formula 5.5, $sim_{\cos}(d_1, d_5)$ is computed as 0.67, thus d_1 and d_5 are considered

relevant in terms of e_1 and e_2. On the other hand, some of the information objects such as d_1 and d_4 have no common entities, thus the similarity between them is 0.

Table 5.3 Matrix representing the named entity weights

	o_1														o_2			
	e_1	e_2	e_3	e_4	e_5	e_6	e_7	e_8	e_9	e_{10}	e_{11}	e_{12}	e_{13}	e_{14}	e_1	e_2	e_3	e_4
d_1	0.33	0.33	0.33	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
d_2	0	0	0	0.33	0.33	0	0	0	0	0	0	0	0	0	0.33	0	0	0
d_3	0	0	0	0	0.50	0	0	0	0	0	0	0	0	0	0	0.50	0	0
d_4	0	0	0	0	0	0.20	0.20	0.20	0.20	0.20	0	0	0	0	0	0	0	0
d_5	0.33	0.33	0	0	0	0	0	0	0	0	0.33	0	0	0	0	0	0	0
d_6	0.50	0	0	0.50	0	0	0	0	0	0	0	0	0	0	0	0	0	0
d_7	0	0	0	0	0	0	0	0	0	0	0	0.33	0.33	0	0	0	0.33	0
d_8	0.25	0	0	0	0	0	0	0	0	0	0	0	0.25	0.25	0	0	0	0.25

Table 5.4 Similarity measure of information objects without ontology

	o_1								o_2							
	d_1	d_2	d_3	d_4	d_5	d_6	d_7	d_8	d_1	d_2	d_3	d_4	d_5	d_6	d_7	d_8
d_1		0	0	0	0.67	0.71	0	0.33		0	0	0	0	0	0	0
d_2	0		0.71	0	0	0.50	0	0	0		0	0	0	0	0	0
d_3	0	0.71		0	0	0	0	0	0	0		0	0	0	0	0
d_4	0	0	0		0	0	0	0	0	0	0		0	0	0	0
d_5	0.67	0	0	0		0.41	0	0.33	0	0	0	0		0	0	0
d_6	0.71	0.50	0	0	0.41		0	0.68	0	0	0	0	0		0	0
d_7	0	0	0	0	0	0		0.41	0	0	0	0	0	0		0
d_8	0.33	0	0	0	0.33	0.68	0.41		0	0	0	0	0	0	0	

However, based on the ontology, d_1 and d_4 contain two related entities: e_1*: Joy* and e_6*: Jamie*. Table 5.5 shows the semantic similarity computed using the weight of both the common and related entities, and formulae 5.5, 5.6 and 5.7. As mentioned above, the similarity of d_1 and d_4 is 0, if the calculation does not involve the ontology. Based on formula 5.6, $sim_{related}(d_1, d_4)$ is 0.07, which is the semantic similarity of d_1 and d_4 when the related entities are considered.

Table 5.5 Semantic similarity measure of information objects with ontology

	o_1								o_2							
	d_1	d_2	d_3	d_4	d_5	d_6	d_7	d_8	d_1	d_2	d_3	d_4	d_5	d_6	d_7	d_8
d_1		0.08	0.13	0.07	0.74	0.84	0	0.04		0	0	0	0	0	0	0
d_2	0.08		0.71	0.05	0.08	0.63	0	0.03	0		0.13	0	0	0	0	0
d_3	0.13	0.71		0.08	0.13	0.19	0	0.05	0	0.13		0	0	0	0	0
d_4	0.07	0.05	0.08		0.04	0.08	0	0.02	0	0	0		0	0	0	0
d_5	0.74	0.08	0.13	0.04		0.59	0	0.37	0	0	0	0		0	0	0
d_6	0.84	0.63	0.19	0.08	0.59		0	0.74	0	0	0	0	0		0	0
d_7	0	0	0	0	0	0		0.41	0	0	0	0	0	0		0.13
d_8	0.04	0.03	0.05	0.02	0.37	0.74	0.41		0	0	0	0	0	0	0.13	

Furthermore, in another ontology regarding to location, the relation weight between e_1*: London* and e_2*: Notting Hill* is set as 1. Within the ontology, the similarity of d_2 and d_3 is 0.13, and it is calculated based on formula 5.7 as follows:

$$sim_{semantic}(d_2, d_3) = sim_{\cos}(d_2, d_3) + \sum sim_{related}(d_2, d_3)$$
$$= sim_{\cos}(d_2, d_3) + \overset{o_1:person}{sim_{related}}(d_2, d_3) + \overset{o_2:location}{sim_{related}}(d_2, d_3)$$
$$= 0.71 + 0 + 0.13 = 0.84$$

The comparison of the data in Table 5.4 and Table 5.5 shows that semantic connections between information objects are detected based on the relation of the *related entities* in the ontologies. By calculating the semantic similarity of any two information objects, the information objects in the data collection can be clustered or distinguished based on their similarity or dissimilarity. Furthermore, the semantic similarity measure facilitates the understanding of the semantic characteristics of the information objects and their clusters. It provides a way to improve the analysis of their meaning, relations and organisation in personal information collections.

5.3 Summary

This chapter proposes an approach to modelling user's background knowledge using a user-oriented ontology. The ontology has a small scale and flexible structure that supports users in browsing, editing and creating new entries. Moreover, its simplified

knowledge structure is populated with semantically homogeneous ontology concepts. The ontology topics are not limited to those mentioned in the chapter, but are extendable to other categories as well.

The chapter also introduces an innovative approach to measuring the semantic similarity between information objects, which applies user-oriented ontologies to detect latent semantic relations between the terms and named entities within the information objects. In this approach, the user-oriented ontology is straightforward, extendable and comprises users' background knowledge, which provides a reliable way to address the semantic gap. As shown by the experimental results, this approach is able to discover similarity between information objects which do not share common content.

Chapter 6

Semantic Feature Matching in Personalised Retrieval for Reminiscence Support

---

The semantic gap remains an unsolved challenge in information retrieval (IR) systems as it reduces their precision and recall. Previously reported term-based approaches lack the capability to process information semantically. As indicated in Chapter 3, some of the unsupervised models developed use statistical analysis to partly solve the gap. However, their capability to analyse and retrieve information at conceptual level is limited. Furthermore, these approaches are not effective enough as users often need to further filter results based on their search requirements or background knowledge.

This chapter introduces an unsupervised semantic feature extraction and selection approach, which uses ontologies to address the semantic gap. This approach is used in both the semantic data analysis and retrieval of Sem-LSB. The chapter is organised as follows. Section 6.1 introduces the semantic feature matching algorithm. Section 6.2 shows the experiment and evaluation of the proposed approach. Section 6.3 summarises this chapter.

6.1 Semantic Features

In the *bag-of-words* model, an information object d_j can be represented as a set of terms, where $d_j = \{t_{1,j}, t_{2,j}, ..., t_{i,j}\}$, $t_{i,j}$ indicates the word appearing in d_j. D is the set of information objects, where $d_j \in D$ and $t_{i,j} \in V_D$; V_D is the vocabulary of D; a word inside of V_D is called a term. Most of the typical IR approaches use normalised term weights to fill information object vectors, and the similarity measure is based on comparing such vectors. For instance, in the vector space model, if information objects

contain common terms, their similarity must be greater than 0. The more common terms they share, the greater their similarity is.

However, certain approaches, such as LSI (latent semantic indexing), pLSI and LDA, are not so straightforward. Besides the common words, they also consider the co-occurrence rate of uncommon words. This enables these methods to detect hidden (latent semantic) relevance. However, if the information objects have no common words, or very few co-occurred words, these approaches may not provide a satisfactory result. For example, assume there are two sentences, "*Lumia 800 is a new model from Nokia in 2011*" and "*Microsoft CEO, Steve Ballmer, announced the new Windows Phone 7 Series at Mobile World Congress 2010*", their relevance (similarity) is hard to be determined by the models without using external knowledge because the sentences do not share words and context. However, most people know that "*Nokia*" is a mobile phone company, the two sentences are both about mobile phones, and therefore they are relevant to some degree to each other. Furthermore, if some of these people are more knowledgeable about "*Windows Phone 7*" and "*Lumia 800*", they may think that the relevance (similarity) is even stronger, because "*Lumia 800*" is the first smart phone of "*Nokia*" which uses the "*Windows Phone 7*" operating system. This example indicates two points: firstly, user's knowledge influences their perspective on the retrieval task, and secondly, external knowledge can help to deal with the semantic gap.

As mentioned earlier, and information object d_j can be represented as a set of terms, where $d_j = \{t_{1,j}, t_{2,j}, ..., t_{i,j}\}$. Suppose an existing user-oriented ontology is related to a user's search intention. Using terms and entities, the information object d_j can be written as $d_j = \{t_{1,j}, ..., t_{i,j}, e_{1,j}, ..., e_{i,j}\}$, where $e_{i,j}$ is the entity contained in d_j; $t_{i,j} \neq e_{i,j}$ and $e_{i,j} \in o_N$. o_N denotes the user-oriented ontology. O is an ontology set including multiple ontologies, $o_N \in O$. A term weight normalisation function is denoted by $g(x)$. If $\hat{t}_{i,j} = g(t_{i,j})$ and $\hat{e}_{i,j} = g(e_{i,j})$ denote the normalised term and

entity weight respectively, then the vector of d_j can be presented as $\vec{d}_j = [\hat{t}_{1,j}, ..., \hat{t}_{i,j}, \hat{e}_{1,j}, ..., \hat{e}_{i,j}]^T$.

Let M denote a term-information object matrix, where $M = \left[\vec{d}_1 \quad \cdots \quad \vec{d}_n\right]$. Each column of M is the vector of an information object, and the matrix does not employ any external knowledge. The idea is to reconstruct a feature matrix $\hat{M}$ of the information objects, using semantic features from the user-oriented ontology. $\hat{M}$ can then be used in information retrieval or clustering in an attempt to fill the semantic gap. Based on the previous example, if the ontology can facilitate detecting latent relevance between "*Windows Phone 7*", "*Nokia*" and "*Lumia 800*", the similarity measure can make the work with those models possible.

6.2 Semantic Feature Extraction

Semantic feature matching involves semantic feature extraction and semantic feature selection that aim to convert M into $\hat{M}$. As mentioned before, the user-oriented ontology o_N contains named entities (refers to its concepts) and relations. The entities are connected by the relations, and different weights are assigned to the relations. In o_N an entity may have more than one other entity (neighbour) connected to it. To measure the similarity of the entities, the employed method is based on formula 5.2. It provides the foundation of the semantic feature matching in this research,

$$edge\_dist(v_a, v_b) = \frac{1}{\ln(w_{r(a,b)} + e)}, \tag{6.1}$$

where $w_{r(a,b)}$ is the weight of the relation between entities v_a and v_b in an ontology.

The named entities contained in an information object are treated as semantic features. Assuming the observed semantic features of d_j belong to the ontology o_N, then $e_{i,j} \in o_N$. The aim of *semantic feature extraction* is to extract the significant

entities from the observed semantic features of d_j by computing the entity frequency in d_j and entropy in o_N. The entity frequency of $e_{i,j}$ in d_j is defined as:

$$\hat{e}_{i,j} = g(e_{i,j}) = \frac{|\, e_{i,j} \in d_j \,|}{len(d_j)} \log \frac{|\, D \,|}{|\, e_{i,j} \in D \,| + 1}, \tag{6.2}$$

where $|\, e_{i,j} \in d_j \,|$ is the occurrence rate of $e_{i,j}$ in d_j; $len(d_j)$ is length of d_j; $|\, D \,|$ is the number of information objects in D; $|\, e_{i,j} \in D \,| + 1$ is the number of information objects containing $e_{i,j}$ in D with smoothing.

The ontology o_N can be converted to a weighted undirected graph (Figure 6.1). The vertices and edges represent named entities and relations respectively. To simplify, all weights of relations are assigned the same value in this example. Based on information theory (Cover and Thomas, 2006), the Shannon entropy of $e_{i,j}$ is defined as:

$$H(e_{i,j}) = -\sum_{i=1}^{n} p(e_{i,j}) \log p(e_{i,j}), \tag{6.3}$$

where n is the number of edges adjacent to $e_{i,j}$ in the ontology graph. $p(e_{i,j})$ denotes the probability of selecting $e_{i,j}$ from any of its adjacent vertices.

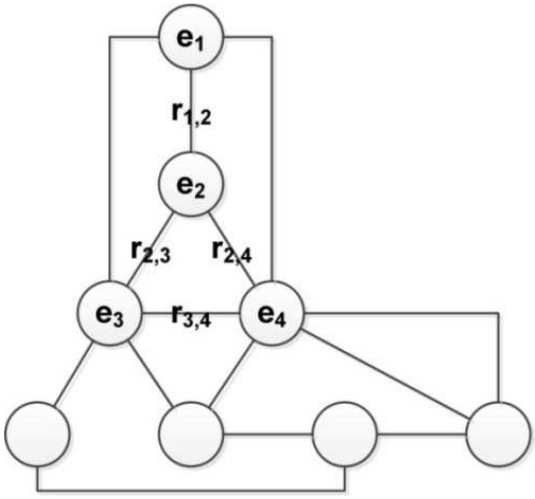


Figure 6.1 Sample of a user-oriented ontology

The entity entropy indicates the semantic information amount carried by an entity. In o_N, an entity with a greater entropy means it carries more semantic information than other entities. In semantic feature extraction, an observed semantic feature with high entropy indicates a high probability of finding its semantically related entities (latent semantic features) from the same ontology. Meanwhile, a high entity frequency means the entity is important for the information object. Combining the impacts of entity frequency and entity entropy, the estimator of semantic feature extraction of d_j is defined as:

$$\hat{\vartheta}(e_{i,j}) = a\hat{e}_{i,j} + \beta H(e_{i,j}), \tag{6.4}$$

where a and β are adjustable parameters representing the weights of $\hat{e}_{i,j}$ and $H(e_{i,j})$ respectively in the feature extraction process, where $a + \beta = 1$, and $a, \beta \geq 0$. The weights are applied to indicate the importance of entity frequency and entropy. a and β are adjustable based on the semantic strength needed in the semantic feature extraction process, e.g. when $\beta = 0$, the process only considers term weight. The parameters can be tuned based on the ontology and the dataset that the system is using. A empirical setting is $a = \beta$, which means the entity frequency and entropy have equal importance.

The set of semantic features contained in d_j is denoted by $F_{obs} = \{e_{1,j}, e_{2,j}, \cdots, e_{i,j}\}$, where $F_{obs} \subseteq o_N$. Appling the *maximum entropy principle* (Berger et al., 1996), semantic feature extraction needs to determine the semantic feature e_h from F_{obs} with the maximum $\hat{\vartheta}(e_{i,j})$, i.e. the one with the highest summation of entity frequency $a\hat{e}_{i,j}$ and entropy $\beta H(e_{i,j})$. Therefore, the determined semantic feature e_h satisfies,

$$\arg\max \hat{\vartheta}(e_{i,j}) \rightarrow e_h, \tag{6.5}$$

which ensures e_h is the semantic feature of d_j that carries a maximal amount of semantic information in o_N, and also has the high importance for d_j.

6.3 Semantic Feature Selection

For e_h, semantic feature selection is applied to select its latent semantic features from o_N. Let $F_{lat} = \{e_{h,1}, e_{h,2}, \cdots, e_{h,n}\}$ denote the entity set adjacent to e_h in o_N, and it represents latent semantic features, where $F_{lat} \cap d_j = \varnothing$. The cumulative entropy indicates the carried semantic information amount the semantic features contained by F_{lat}, and it is defined as:

$$\varphi(F_{lat}) = -\sum_{j=1}^{n}\sum_{i=1}^{m} p(e_{h,j}) \log p(e_{h,j}) \tag{6.6}$$

where m is the edge number of $e_{h,j}$ in the graph of o_N; n is the number of entities in F_{lat}.

Let $F_{lat\max}$ denote set of k latent semantic features (selected) of d_j generated by e_h with maximal value of $\varphi(F_{lat})$, where $F_{lat\max} \subseteq F_{lat}$. The neighbours may be connected to e_h by different relations, according to formula 6.1, the neighbour with the shorter distance has a higher priority to be selected in the semantic feature selection process. The selection process follows the maximum entropy principle. This implies that the sum of the entropies of the k selected features, $F_{lat\max}$ must be greater than the sum of the entropies of any combination of k features out of the ones remaining in F_{lat}. Then $F_{lat\max} = \{e_{l,1}, e_{l,2}, ..., e_{l,k}\}$ represents the selected latent features, k can be interpreted as a threshold, which controls the impact of the ontology in the semantic feature matching process. The entity frequency value of the features is set as c^k, and c is an constant satisfying $0 < c \le \dfrac{\max \mid t_m \in d_j \mid}{len(d_j)}$, where $\max \mid t_m \in d_j \mid$ is the highest term count of t_m in d_j, and $len(d_j)$ is the length of d_j (

c is set as 0.5 in this work). Then latent semantic features with their frequency values are filled in the vector. Let $\vec{d}_j^{'}$ denote the semantic feature vector, then

$$\vec{d}_j^{'} = [\hat{t}_{1,j}, ..., \hat{t}_{i,j}, \hat{e}_{1,j}, ..., \hat{e}_{i,j}, e_{l,1}, e_{l,2}, ..., e_{l,k}]^T \tag{6.7}$$

Based on the feature matrix and its contained vectors, IR systems are able to process data on semantic basis. Figure 6.2 shows how the observed terms, entities and latent semantic features establish the connection between the information objects. The solid line means that the connection is from a pairs of observed variables, while the dash line indicates a connection from the latent semantic features. Based on the figure, d_1 and d_2 are relevant because of the common entity e_1. Using the ontology, e_3 and e_5 are found correlated due to their neighbour e_4. In other words, e_4 is treated as the latent semantic feature that establishes a semantic connection between d_1 and d_2. The established semantic connection makes the similarity between these information objects measurable. Clearly, d_2 and d_3 do not have common terms or share any contexts. Based on the ontology, e_3, e_4 and e_2, e_3 are treated as correlated semantic features, then the semantic connections between d_2 and d_3 are established.

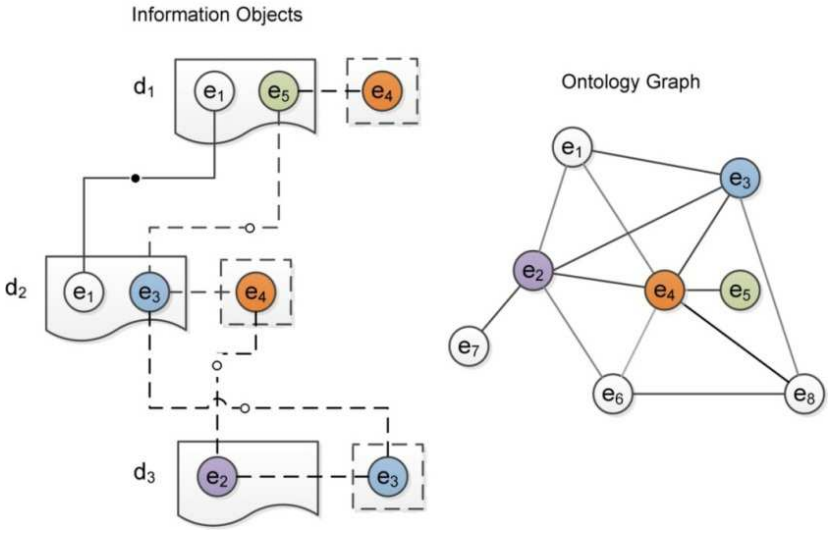


Figure 6.2 Information objects linked using a user-oriented ontology

6.4 Multi-ontology Semantic Model

In practice, information objects may involve cross-domain knowledge. As mentioned before, a single user-oriented ontology contains knowledge from a specific domain, meaning that it can only process the data related to the topics in that domain. To analyse and select the latent semantic features from cross-domain knowledge, a multi-ontology semantic model is proposed.

The multi-ontology semantic model integrates several user-oriented ontologies and the relations between them. In general, the multi-ontology semantic model represents the hierarchy structure of knowledge. It is constructed by various named entities (belonging to different topics) with horizontal and vertical connections, explained as follows.

A sample of the model is visualised in Figure 6.3. It includes *ontology*<***Noika***>, *ontology*<***HTC***> and *ontology*<***Samsung***>, each of which contains the related mobile models. *ontology*<***Mobile OS***> contains the mobile operation systems. *Horizontal connection* represents the relations within a single ontology. Referring to Figure 6.3, a horizontal connection can simply be represented by the relation between "*Nokia: N9*" and "*Nokia: Lumia 800*"within *ontology*<***Nokia***>.

The dash lines in Figure 6.3represent the connections between different ontologies and connect the correlated named entities from these ontologies. This type of connection is defined as *vertical connection*. Back to Figure 6.3, e.g. *ontology*<***Nokia***> includes "*Nokia: N9*" and "*Nokia: Lumia 800*", and *ontology*<***Mobile OS***> includes "*MeeGo*" and "*Windows Phone 7*", hence the connection between these two ontologies in this case can be established as a "has-a" relation. This relation can be deemed as a vertical connection.

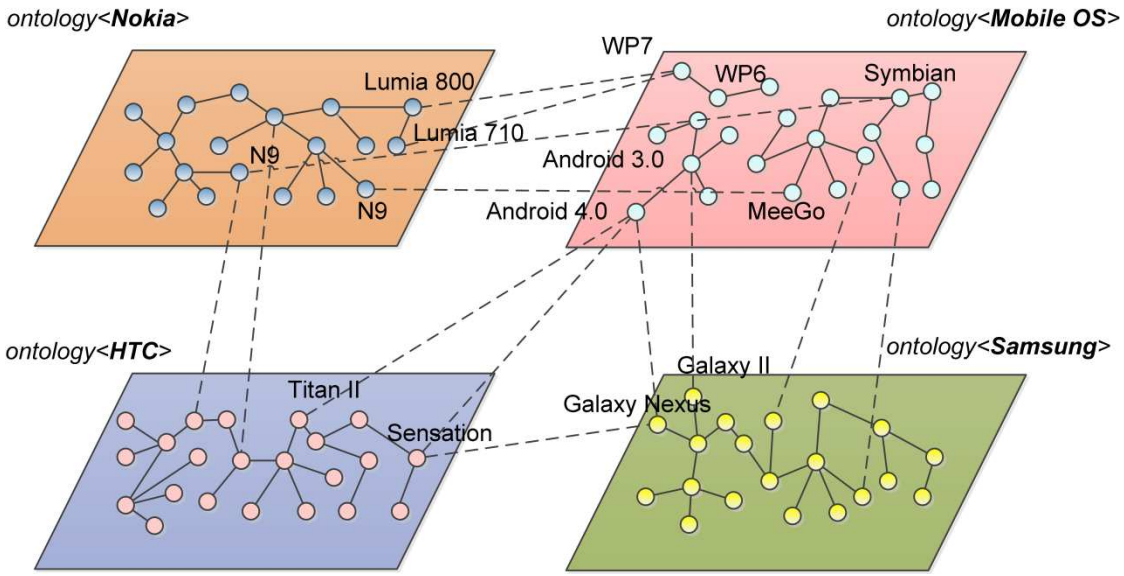


Figure 6.3 A multi-ontology semantic model to handle cross-domain knowledge

Based on the model, the feature vector of d_j is represented as:

$$\vec{d}_j' = [\hat{t}_{1,j},\cdots,\hat{t}_{i,j},\hat{e}_{1,j},\cdots,\hat{e}_{i,j},a_1\cdot\overbrace{e_{l,1},e_{l,2},\cdots,e_{l,k1}}^{o_1},\cdots,$$
$$a_N\cdot\overbrace{e_{l,1},e_{l,2},\cdots,e_{l,kN}}^{o_N}]^T \tag{6.8}$$

where $o_1,\cdots,o_N$ are different user-oriented ontologies; $a_1,\cdots,a_N$ are the weights of the ontologies satisfying $\sum a_N = 1$; $k_1,\cdots,k_N$ are the thresholds corresponding to $o_1,\cdots,o_N$.

Using the semantic feature matching with the multi-ontology semantic model, cross-domain retrieval can be achieved, e.g. if an information object only contains "*HTC: Titan II*", it still can establish semantic connections with the information objects describing "*Android*", "*Samsung: Galaxy II*".

6.5 Experiment and Evaluation

The evaluation uses a collection of short news documents, a dataset harvested from three English news websites[1]. The testing data is about technology news, e.g. mobile phones and mobile operation systems. Each document is manually tagged with one of the several labels: *Nokia*, *HTC*, *Samsung*, *Apple*, *Android*, *IOS* and *MeeGo*. For example, d_1: "*The model **Nokia** 8110, one of the original slider phones, was released in 1996 to great acclaim. For the time it provided a powerful and fast mobile phone with innovative design. It also featured in the original Matrix movie.*" and d_2: "*Lumia 710 is a budget-friendly version of the Lumia 800 flagship. With basically the same internals, it skimps only on storage and camera quality relative to its older sibling.*" describe different phone models of Nokia, thus they have the same label "*Nokia*". The knowledge base is created by identifying topics, such as *"**HTC**"*, *"**Samsung**"*. Each topic is represented as an ontology of named entities. In the evaluation these are phone models, e.g. *ontology*<***HTC***> includes "*Titan II*" and "*Sensation*" (see Figure 6.3), or mobile operating system names, e.g. *ontology*<***Mobile OS***> includes "*Android 3.0*" and "*MeeGo*".

There are two types of testing queries: with and without topic words. Topic words can provide a high-level semantic indication of queries. "***Samsung***, *Galaxy*" and "***HTC**, Titan II*" are queries with topic words, while "*Galaxy*" and "*Titan II*" are queries without topic words. The documents under each topic can be classified as two types: containing topic words (e.g. d_1 with ***Nokia***) and not containing topic words (e.g. d_2). Following a cross validation principle, a query set for a topic is generated using 20% of the documents which contain the topic words, and 20% of the documents in which the topic words are not mentioned. Each topic generates 5 different query sets, i.e. one set for each 20% of the documents within the topic.

[1] www.bbc.co.uk, www.telegraph.co.uk and www.dailymail.co.uk

The evaluation compares precision and recall between VSM and sVSM (semantic VSM, with ontologies) by testing the query sets. The term weight normalisation of document vectors is TF-IDF, and the similarity measure between them is cosine similarity $sim_{\cos} = \frac{v_a \cdot v_b^T}{\|v_a\|_2 \cdot \|v_b\|_2}$. VSM T+ and VSM T- represent the evaluation of VSM with the testing queries including and excluding topic words. sVSM applies ontology and semantic feature matching to reconstruct the document vectors, and k is the threshold in the semantic feature selection process, e.g. k1 means that the reconstructed vector of information object contains 1 selected semantic feature from the ontology.

Table 6.1 lists the detailed semantic feature matching results of several queries. For example, in a query "other models launched with *Hero*", "*Hero*" is a named entity from the *ontology*<***HTC***>, and its suitable neighbour (semantic feature) is "*Nexus One*", when k is 1. Then the original query is converted to "other models launched with *Hero*" + "e_1: *Nexus One*", and then the semantic retrieval process should consider both of "*Hero*" and "*Nexus One*", rather than "*Hero*" only. Practically, most of the short queries have high semantic ambiguity which is difficult to eliminate. The approach reported in this work can eliminate the ambiguity and improve recall without losing too much precision, as the selected semantic features always belong to the same topic or knowledge domain of the original queries. Furthermore, the gathered semantic features enable the IR system to identify the topics without using topic words, that improving precision on semantic basis. For example, the word "*Hero*" may retrieve the song or the movie at the same time, which is irrelevant. According to the user's background knowledge of mobile products, by adding the semantic feature "*Nexus One*", the retrieval result should be more related to the user's intention.

Table 6.1 Results of the semantic feature selection using ontologies for different tests. N90, Hero and Moment are the identified features. Nokia, HTC, Samsung and Mobile OS are the topics.

VSM T-	VSM T+	sVSM-k1 T-	sVSM-k2 T-	sVSM-k2 T+
N90	N90, **Nokia**	N90, e_1: N95	N90, e_1: N95 e_2: Symbian	N90, e_1: N95, e_2: Lumia 710, **Nokia, Mobile OS**
Hero	Hero, **HTC**	Hero, e_1: Nexus One	Hero, e_1: Nexus One, e_2: Android 2.0	Hero, e_1: Nexus One, e_2: Desire, **HTC, Mobile OS**
Moment	Moment, **Samsung**	Moment, e_1: Galaxy	Moment, e_1: Galaxy, e_2: Android 2.0	Moment, e_1: Galaxy, e_2: Note, **Samsung, Mobile OS**

Figure 6.4 shows that the performance of VSM is better if the queries contain topic words as queries with topic words and corresponding model names, such as "***Samsung**, Galaxy*" or "***HTC**, Titan II*", always get better retrieval results than queries which only have the model names, such as "*Galaxy*" or "*Titan II*". The retrieval performance improvement of VSM T+ over VSM T- is +27.75%. Using the same testing queries without topic words, sVSM-k1 T- outperforms VSM T- and VSM T+, and the improvements are +38.34% and +8.29%.

The strength of the semantic representation of an information object can be changed by adding (or reducing) semantic features. In Figure 6.5, when k is 2, the retrieval performance of sVSM-k2 T- is better than sVSM-k1 T-, and the improvement is +9.73%. In addition, Figure 5 shows that topic words have positive effect in sVSM retrieval, i.e. sVSM-k2 T+ outperforms both sVSM-k1 T- and sVSM-k2 T-, the improvements are +13.33% and +4.32%.

The F-score results of all the tests are shown in Figure 6.6 and 6.7. VSM T- has the lowest F-score in the experiment, and the average improvements of others against VSM T- are, +13.41% VSM T+, +17.68% sVSM-k1 T-, +22.90% sVSM-k2 T- and +23.07% sVSM-k2 T+, respectively. The effect of topic words on retrieval

performance steadily decreases with the increasing of threshold k. However, the retrieval performance is not always linearly increasing with k, e.g. if the threshold k is inappropriately high, it may cause semantic ambiguity and over-fitting. Overall, the results of the experiment indicate that the proposed algorithm with the user-oriented ontology has ability to improve the retrieval performance on semantic basis.

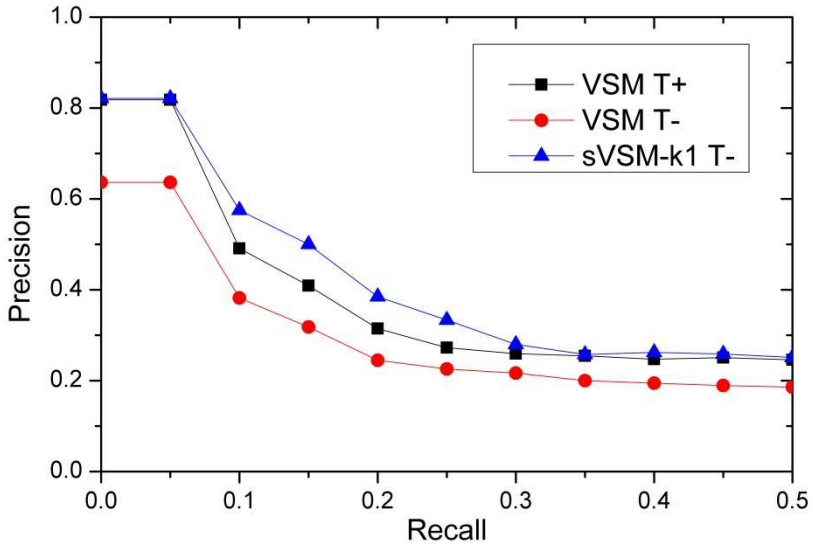


Figure 6.4 Precision and recall of VSM and sVSM with different tests

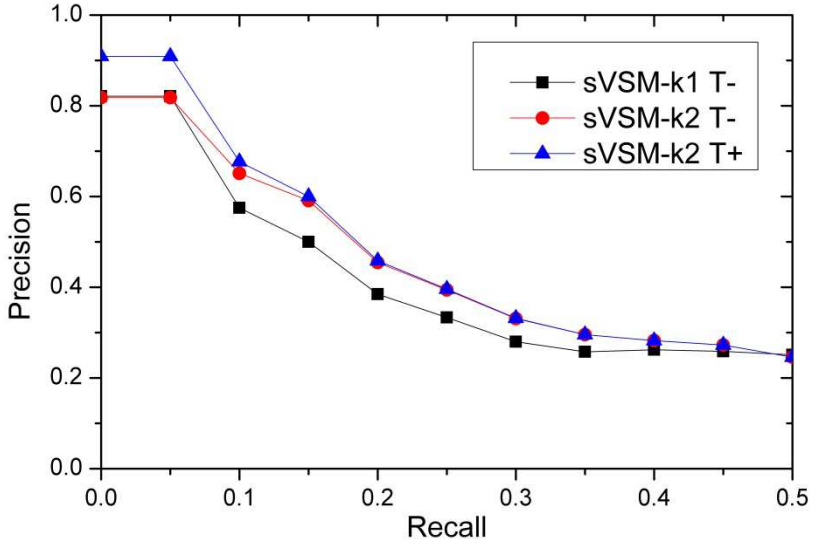


Figure 6.5 Precision and recall of sVSM with different tests

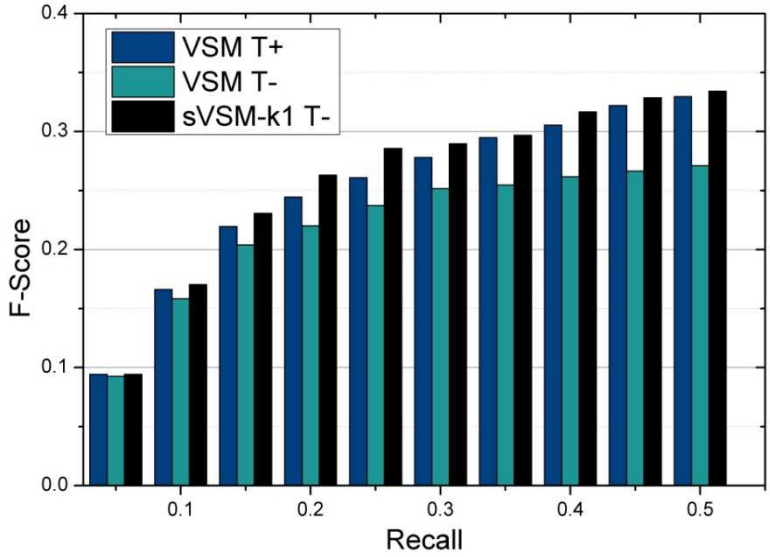


Figure 6.6 F-score of VSM and sVSM with different tests

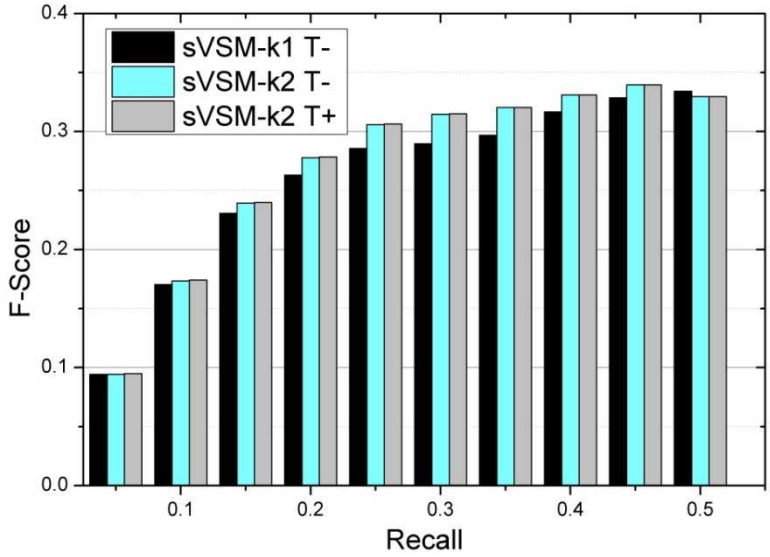


Figure 6.7 F-score of sVSM with different tests

6.6 Summary

This chapter describes a semantic feature matching algorithm based on user-oriented ontologies. The algorithm analyses the observed semantic features in information objects and then using user-oriented ontology, selects the latent semantic features to

enhance their semantic representation. The experiment examines the similarity measure based on the enhanced representation of the information objects. The results show that the approach provides improved retrieval performance due to its use of semantic knowledge. Furthermore, the approach can also be applied to generate the semantic feature-based matrix of prior knowledge, thus it can be integrated into the pre-processing step of other IR mechanisms, such as LSI or LDA, and it can also be directly integrated with the VSM model or other vector-based data clustering methods.

Chapter 7

Ontology-Based Clustering of Stored Memory

This chapter proposes a user-oriented approach for topic identification and clustering of stored memories, i.e. content related to the life of a person. The chapter is organised as follows. Section 7.1 introduces modules and processes related to the approach. Section 7.2 presents dimension reduction using Singular Value Decomposition (SVD). Section 7.3 introduces the approach developed called Onto-SVD and provides illustrative examples. Section 7.4 includes the experimental evaluation of the proposed approach. Section 7.5 concludes the chapter.

7.1 Modules and Process

As shown in Figure 7.1, the proposed approach named Onto-SVD includes modules for natural language processing (NLP), named entity recognition (NER), semantic feature matching, dimensionality reduction (SVD) and clustering. These modules are based on processing textual data which relies on the quality of the annotations provided by the user. The NLP module executes tokenisation, stop-words removal, stemming and corpus building. It removes unimportant terms and symbols, splits textual data into tokens, and convert tokens to their root forms. A lexical ontology is used at this stage to facilitate the processing. The NER module detects and labels the named entities from the information objects. Next, the semantic feature matching uses these named entities to select semantically related named entities from the user-oriented ontologies; it then expands the feature space of each information object by including all these selected entities that are further used to identify and establish latent semantic connections with other information objects. The dimensionality reduction approach further facilitates the identification of latent semantic connections between the information objects. The clustering module processes the data further,

and categorises similar information objects in clusters. The user-oriented ontology provides knowledge support to the semantic feature matching module whose task is to identify and select the most relevant semantic features.

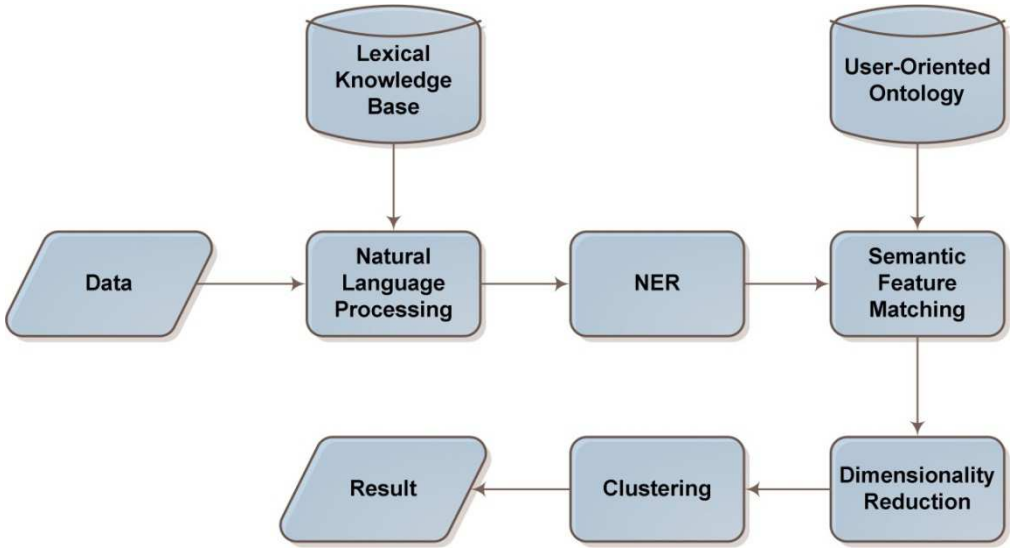


Figure 7.1 Modules and process of Onto-SVD

7.2 Dimensionality Reduction

Dimensionality reduction plays an essential role in multivariate data analysis, e.g. text analysis, image processing, and analysis of gene expressions. One core technology of dimensionality reduction is based on the matrix factorisation method, e.g. SVD and PCA (Principal Component Analysis).

The SVD method can be formally represented as follows: let M be a real $m \times n$ matrix with rank r (i.e. the number of linearly independent rows/columns of M). It can be factored as,

$$M = U\Sigma V^T, \tag{7.1}$$

where U and V are $m \times m$ and $n \times n$ orthogonal matrices respectively, where $I = U^T U = V^T V$; Σ is a diagonal matrix with the same size as M. The nonnegative

entries on the main diagonal are singular values of M, $\Sigma = diag(\sigma_1, \sigma_2, \cdots, \sigma_n)$, where $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0$, if $1 \leq i \leq r$; and $\sigma_{r+1} = \sigma_{r+2} = \cdots = \sigma_n = 0$, if $r+1 \leq i \leq n$.

As a low-rank approximation, truncated SVD converts data from a r-dimensional to k-dimensional projected space, where $1 \leq k \ll r$. In other words, it finds a rank-reduced matrix M_k by minimising the difference of Frobenius norm Δ between the initial matrix M and M_k,

$$\|M\|_{Forbenius} = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}\left|a_{ij}\right|^2}, \Delta = \|M - M_k\|, \tag{7.2}$$

where Δ is information loss, and it is determined by the parameter k, for example, a greater k indicates less information loss.

Let u_i and v_i denote a column of U and a row of V^T, then u_i and v_i are called left singular vector and right singular vector of M, respectively. The rank-k approximation M_k of M is represented as ,

$$M_k = U_k \Sigma_k V_k^T = \sum_{i=1}^{k} u_i \cdot \sigma_i \cdot v_i^T, \tag{7.3}$$

where $k \ll r$.

The following formula is applied to convert a vector q into the k-dimensional space,

$$\tilde{q} = q^T U_k \Sigma_k^{-1}. \tag{7.4}$$

Latent semantic indexing (LSI) uses truncated SVD to partly solve the polysemy and synonymy problem. Besides analysing the most frequent usage of common terms, LSI also considers their co-occurrence in contexts that enables LSI to identify the semantic similarity of information objects. The LSI outperforms vector space model (VSM) on semantic basis for the following reasons. In the VSM, the similarity between the information objects is measured on the basis of the common terms they contain. The limitation of VSM is that it is hard to distinguish the meaning of the

terms based on the context, which is important when dealing with polysemy and synonymy. For instance, VSM cannot distinguish between "*apple*" as a type of fruit and "*apple*" as a brand, as it treats them identically based on their spelling. This causes disordered VSM retrieval results leading to reduced precision. Therefore, users need to filter the results based on their own knowledge. The second problem is *synonymy*. For instance, two documents may contain the words "*sorrowful*" and "*sad*", respectively. If one of them is used in a query, VSM would not return the other one as a retrieval result, although the two terms are synonyms. This has a negative impact on recall. These two problems illustrate the inadequacy of the VSM technique to complete complicated tasks such as topic identification and semantic relation detection.

7.3 Onto-SVD

The proposed Onto-SVD dimensionality reduction approach originates from the idea that the semantic meaning of each information object can be represented through a combination of terms and named entities included in it. As mentioned before, textual data is normally represented in a high-dimensional space where each term or named entity is treated as one dimension, and each dimension is orthogonal to the others. The Onto-SVD approach combines semantic feature selection with user-oriented ontology, and uses SVD as a dimension reduction method to achieve topic identification based on semantic similarity.

Table 7.1 shows a dataset containing 5 information objects d_1 - d_5. The underlined terms are used in this example as semantic features. Two retrieval algorithms, namely VSM and SVD, have been tested with the query "*Queen Elizabeth*". The VSM algorithm retrieves information objects d_1 and d_5 as they both include the word "*Elizabeth*". In the case of using the SVD algorithm, the result also includes information object d_4 because although d_4 does not contain common terms with the query, *Westminster Abbey* provides a cue to a latent semantic relation.

Table 7.1 Information objects

d_1	20 November *1947*, Princess *Elizabeth* and Prince *Philip* leave *Westminster Abbey* after their wedding.
d_2	Prince *Charles* and Princess *Diana* on the balcony of *Buckingham Palace* in *1981*.
d_3	9 April *2005*, *Camilla*, Duchess of *Cornwall*, with Prince *Harry*, Prince *William*, *Laura* and *Tom Parker Bowles*.
d_4	Prince *William* and *Kate Middleton* announced their engagement with plans to marry in summer *2011*, *Westminster Abbey*.
d_5	*Windsor Castle* is the oldest and largest occupied castle in the world and the official residence of Her Majesty Queen *Elizabeth* II.

However, as shown by this example, the SVD-based approach has limitations since the two remaining objects, d_2 and d_3, are also related to Queen Elizabeth's life. It is well known fact that *Elizabeth* and *Philip* are connected to *Charles* by semantic relation <*parent-of*>, and both *Westminster Abbey* and *Windsor Castle* are official residences of the Queen and the Royal family. Therefore, in the context of this example, the retrieval result should ideally also contain d_2 and d_3. However, this type of connection is hidden for SVD which only considers term co-occurrences. If the co-occurrence is very low or zero, it would fail to detect existing connections. An improved approach named Onto-SVD is presented in the next section to address this problem. It uses additional domain knowledge which is specific to each individual, e.g. their family, circle of friends, locations, and important events in life.

7.3.1 Algorithm

Onto-SVD is an integrated algorithm involving the *bag-of-words* model, use-oriented ontologies, semantic feature matching and dimensionality reduction. Let O denote a set of ontologies $O = \{o_1, o_2, ..., o_k\}$ and E_j is the set of semantic features contained by an information object d_j, where $E_j = \{e_{i,j} \mid e_{i,j} \in d_j\}$. The feature set of d_i is represented as,

$$F_j = \{e_{i,j} \mid e_{i,j} \in E_j, e_{i,j} \in o_N\}, \tag{7.5}$$

which indicates that semantic feature $e_{i,j}$ is contained in both an information object and an ontology. $g(x)$ is a term weight normalisation function. d_j is then converted to a feature vector. Using $\vec{f}_j$ represents semantic feature vector of d_j then,

$$\vec{f}_j = [\overbrace{g(e_{1,j}) \cdots g(e_{i1,j})}^{o1}, \overbrace{g(e_{1,j}) \cdots g(e_{i2,j})}^{o2}, \cdots, \overbrace{g(e_{1,j}) \cdots g(e_{iN,j})}^{oN}] \tag{7.6}$$

where i_N is the number of semantic features of d_j belonging to o_N and $l = \sum i_N$ is the total number of named entities in d_j; N is the number of ontologies ($k = 1$ for a single ontology model and $k > 1$ for a multi-ontology model). For example, the information object d_3 in Table 7.1 contains 7 named entities which belong to 3 different ontologies, namely people (*Camilla, Harry, Williams, Laura, Tom Parker Bowles*), location (*Cornwall*), and time (*2005*), thus $l = i_1 + i_2 + i_3 = 5 + 1 + 1 = 7$.

The semantic feature space F_D of dataset D is defined through its feature sets,

$$F_D = \bigcup_{j=1}^{n} F_j, \tag{7.7}$$

where n is the number of information objects contained in D. The feature matrix is constructed based on the feature space and feature vectors. It is a sparse matrix which has the same size as the initial term-information object matrix where all non-entity entries are represented as zeros. The feature matrix of the dataset D is denoted by M_F,

$$M_F = \left[\vec{f}_1^{\,T} \quad \cdots \quad \vec{f}_j^{\,T} \right], \tag{7.8}$$

the entries of M_F are weight of the entities initially contained within the information objects.

Semantic feature matching is then used to enhance the semantic representation of information object, by selecting neighbours of the contained entity (entities) of the information object from related user-oriented ontologies. Assuming the semantic

features in F_j are contained in an ontology o_N, i.e. $\forall e_{i,j} \in F_j : e_{i,j} \in o_N$. The ontology o_N is treated as a weighted undirected graph $G_N = (V, E)$, where V is the vertex set, $\forall e_{i,j} \in F_j : e_{i,j} \in V$, and let E denote the edge (relation) set, which is applied to connect the entities (vertices). The degree $\deg(v_i)$ of a vertex v_i is represented through the number of its connected entities (vertices) in G_N. The adjacency matrix of G_N is denoted as $A(G_N)$, and its entry $a_{i,j}$ is represented as,

$$a_{i,j} = \begin{cases} 1, v_i \ adj \ v_j \\ 0, v_i \ nadj \ v_j \ or \ i = j, \end{cases} \tag{7.9}$$

where i and j indicate vertices v_i and v_j respectively. Then, the degree of v_i is computed as $\deg(v_i) = \sum_j a_{i,j}$. The self-information of a named entity $e_{i,j}$ with an outcome is $I(e_{i,j}) = -\log p(e_{i,j}) = -\log \frac{1}{\deg(e_{i,j})}$; the outcome indicates the probability of this named entity being selected as a feature in the semantic feature selection process. The Shannon entropy of $e_{i,j}$ is written as:

$$H(e_{i,j}) = \sum_{i=1}^{n} p(e_{i,j}) \cdot I(e_{i,j}) = \sum_{i=1}^{n} \frac{1}{\deg(e_{i,j})} \cdot \log \frac{1}{\deg(e_{i,j})} \tag{7.10}$$

where $n = \deg(e_{i,j})$. The entropy measures the semantic information of $e_{i,j}$.

The algorithm includes selecting, for each information object, a named entity $e_{i,j}$ (further referred to in this chapter as an identified entity) from the semantic feature set F_j, and then extracting its related entities from the ontology o_N (i.e. its neighbouring vertices in the graph representing the ontology relations). The selection of the detected entity is based on the maximum entropy principle (Berger et al., 1996), that suggests the identified entity $e_{i,j}$ should be the entity with the highest entropy in the corresponding information object d_j. As shown in formula (7.7), $H(e_{i,j})$ is proportional to $\deg(e_{i,j})$, which means that the named entity with a higher

entropy is the one with a higher degree. The neighbours are the nearest entities of $e_{i,j}$ in G_N. Onto-SVD considers $e_{i,j}$ together with its neighbours as the semantic representation of information object d_j. Formula (7.11) is based on formula (5.2), and it is used to measure the length of the edge between any two entities $e_{i,j}$ and $e_{i+1,j}$ in the graph,

$$edge(e_{i,j}, e_{i+1,j}) = \frac{1}{\ln(weight(e_{i,j}, e_{i+1,j}) + e)} \tag{7.11}$$

where $weight(e_{i,j}, e_{i+1,j})$ is the weight of the relation between $e_{i,j}$ and $e_{i+1,j}$.

Let $N_h = \{n_1, n_2, ..., n_k\}$ denotes the set of selected neighbour of the identified entity e_h, where $k = \deg(e_h)$. The parameter t is an adjustable threshold which limits the number of neighbouring entities of e_h selected from N_h. It is used to adjust the strength of the semantic representation. Let $t(e_h)$ denotes a threshold function,

$$t(e_h) = \begin{cases} \deg(e_h), \textit{ if } t > \deg(e_h) \\ t, \textit{ otherwise} \end{cases}. \tag{7.12}$$

N_t denotes the neighbour set of e_h with threshold t, where $N_t \subseteq N_h$. Assuming the semantic feature selection process picks t neighbours of an identified entity e_h, the process can be represented as a chain, which has joint conditional probability:

$$\begin{aligned} &P(e_h, N_t) = P(n_1 \mid e_h)P(n_2 \mid e_h, n_1)...P(n_t \mid e_h, n_1, ...n_{t-1}) \\ &= \prod_{i=1}^{t} P(n_i \mid n_0, n_1, ...n_{i-1}), \end{aligned} \tag{7.13}$$

where $n_0 = e_h$.

The entropy of the selection result is,

$$\begin{aligned} &H(e_h, N_t) = H(n_1 \mid e_h) + H(n_2 \mid e_h, n_1)...H(n_t \mid e_h, n_1...n_{t-1}) \\ &= \sum_{i=1}^{t} H(n_i \mid n_0, n_1, ...n_{i-1}), \end{aligned} \tag{7.14}$$

where $n_0 = e_h$.

To reduce the computational complexity, a bigram form of formula 7.13 is written as follows,

$$\begin{aligned} P(e_h, N_t) &= P(n_1 \mid e_h)P(n_2 \mid n_1)...P(n_t \mid n_{t-1}) \\ &= \prod_{i=1}^{t} P(n_i \mid n_{i-1}). \end{aligned} \tag{7.15}$$

The entropy of the selection result according to the bigram form is,

$$\begin{aligned} H(e_h, N_t) &\cong H(n_1 \mid e_h) + H(n_2 \mid n_1)... + H(n_t \mid n_{t-1}) \\ &= \sum_{i=1}^{t} H(n_i \mid n_{i-1}). \end{aligned} \tag{7.16}$$

The aim of selecting neighbours is to enhance the semantic representation of the identified entity e_h and reduce the risk of selecting neighbours with weak semantic representation. Therefore, the selecting process needs to satisfy,

$$H(e_h, N_t) = \arg\max \sum_{i=1}^{t} H(n_i \mid n_{i-1}). \tag{7.17}$$

In other words, the result of the semantic feature selection based on the maximum entropy ensures that the selected features have strong semantic value, i.e. the selected entities should have high entropy to maximize $H(e_h, N_t)$. Therefore, the selected neighbours should have the highest degree (entropy).

Let $\vec{f}_j^{\,t}$ and M_F^t denote the semantic feature vector with the t selected feature value(s) and the semantic feature matrix respectively,

$$M_F^t = \left[(\vec{f}_j^{\,t})^T \quad \cdots \quad (\vec{f}_j^{\,t})^T \right] \tag{7.18}$$

where t indicates the threshold of semantic feature selection. After applying the semantic feature matrix M_F^t to the initial matrix M, the semantically enhanced matrix $M_{enhanced}$ is produced,

$$M_{enhanced} = M + M_F^t. \tag{7.19}$$

The next step of the algorithm uses SVD to decompose $M_{enhanced}$,

$$M_{enhanced} \cong U_k \Sigma_k V_k^T, \tag{7.20}$$

where U_k and V_k^T multiplied with their corresponding singular values are treated as terms (entities) and information objects' projections in the k dimensional space, respectively. As the k-dimensional space has the same attributes as the Euclidean space, cosine similarity is applied to compare the vectors' similarity in this space,

$$sim_{\cos} = \frac{v_a \cdot v_b^T}{\|v_a\|_2 \cdot \|v_b\|_2}. \tag{7.21}$$

The algorithm is used to calculate the semantic similarity of information objects as shown in the illustrative example below.

7.3.2 Illustrative Example

This section presents an example of using a simple ontology representing relations within the British Royal Family. The section further develops the example introduced in the previous section (see Table 7.1) to illustrate the process of creating semantic feature matrix using the user-oriented ontology. A part of the ontology *Royal Family* related to the head of the Royal Family is shown in Figure 7.2 and 7.3. Note that only part of the family tree contains labelled entities (e_1-e_{10}) as not all family members are mentioned in the small collection of information objects used in the example shown in Table 7.1.

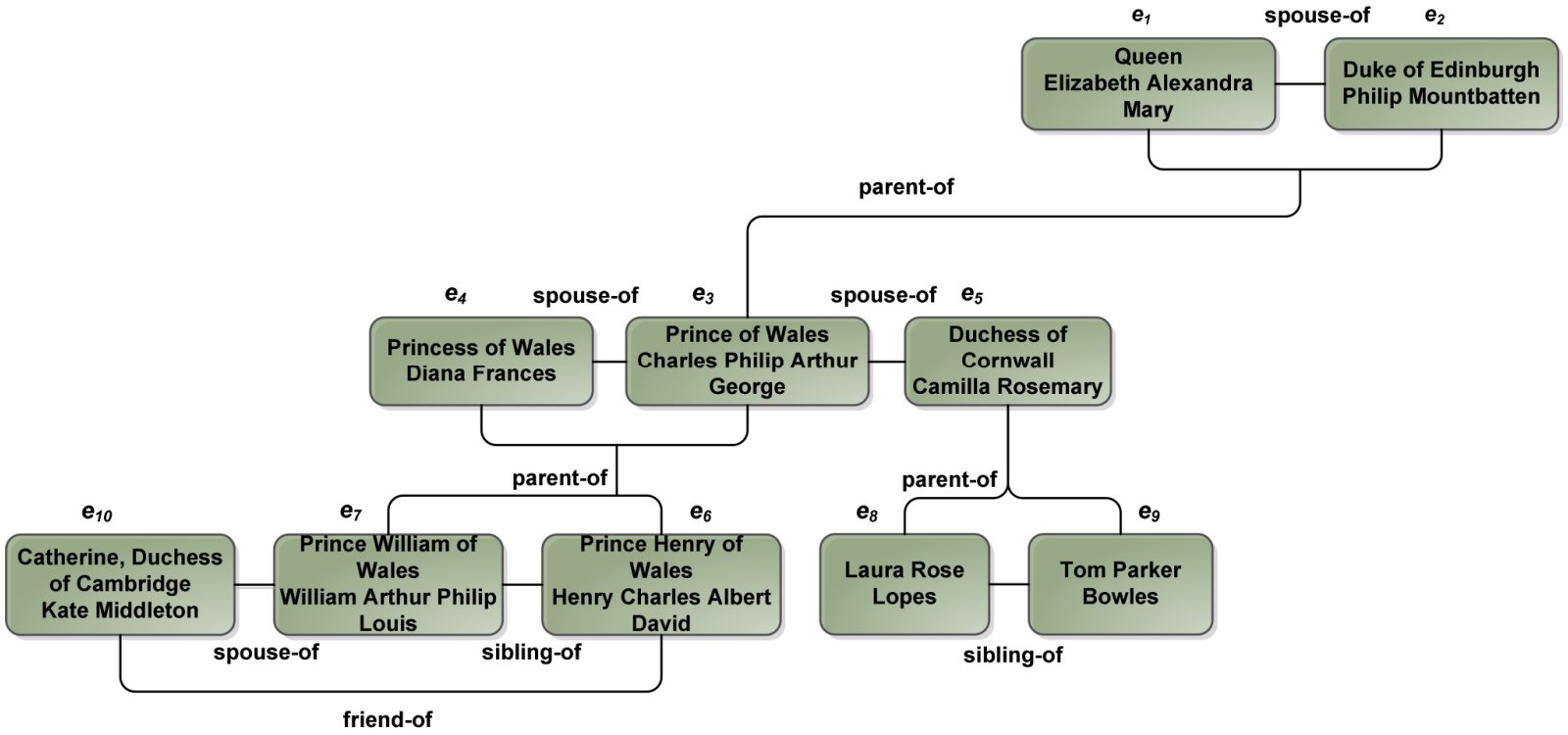


Figure 7.2 Part of a user-oriented ontology with 10 named entities

The next step of the example is to extract the named entities included in the information objects and analyse them. For simplicity, time and locations are not considered in this example. For instance, information object d_1 contains two named entities: *Elizabeth* (e_1) and *Philip* (e_2); information object d_2 also contains two named entities: *Charles* (e_3) and Diana (e_4), etc. The graph representation of the ontology (Figure 7.2) provides background knowledge needed to select the features (neighbours), and then establish the semantic relations between the information objects.

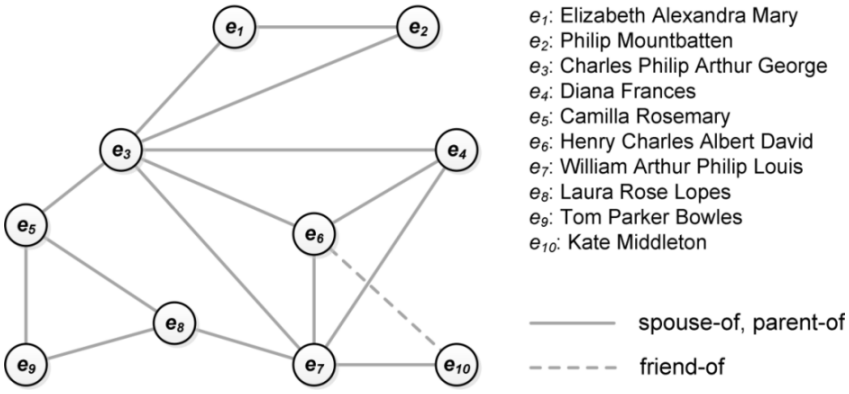


Figure 7.3 Weighted undirected graph representing relations within the user-oriented ontology

To compare the degrees of the named entities (vertices in the ontology graph), the adjacency matrix corresponding to Figure 7.3 is shown below. The adjacency matrix A_N is symmetric, and the entry $a_{i,j}$ indicates the adjacency of e_i and e_j,

$$A_N = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \end{pmatrix}.$$

For example, e_1 and e_2 are adjacent, then elements $a_{12} = a_{21} = 1$; e_1 and e_4 are not adjacent, $a_{14} = a_{41} = 0$. The degree of e_1 is $\deg(e_1) = \sum_{1 \le j \le 10} a_{1j} = 2$.

Table 7.2 Initial matrix and its related semantic feature matrices

	d_1	d_2	d_3	d_4	d_5
e_1	1	0	0	0	1
e_2	1	0	0	0	0
e_3	0	1	0	0	0
e_4	0	1	0	0	0
e_5	0	0	1	0	0
e_6	0	0	1	0	0
e_7	0	0	1	1	0
e_8	0	0	1	0	0
e_9	0	0	1	0	0
e_{10}	0	0	0	1	0

(a) term-information object matrix

	d_1	d_2	d_3	d_4	d_5
e_1	1	0	0	0	1
e_2	1	0	0	0	0
e_3	.5	1	.5	.5	.5
e_4	0	1	0	0	0
e_5	0	0	1	0	0
e_6	0	0	1	0	0
e_7	0	.5	1	1	0
e_8	0	0	1	0	0
e_9	0	0	1	0	0
e_{10}	0	0	0	1	0

(b) semantic feature matrix, t=1

	d_1	d_2	d_3	d_4	d_5
e_1	1	0	0	0	1
e_2	1	0	0	0	.25
e_3	.25	1	.25	.25	.25
e_4	0	1	.25	0	0
e_5	0	0	1	0	0
e_6	0	.25	1	.25	0
e_7	0	.25	1	1	0
e_8	0	0	1	0	0
e_9	0	0	1	0	0
e_{10}	0	0	0	1	0

(c) semantic feature matrix, t=2

	d_1	d_2	d_3	d_4	d_5
e_1	1	0	0	0	1
e_2	1	0	0	0	.125
e_3	.125	1	.125	.125	.125
e_4	0	1	.125	.125	0
e_5	0	.125	1	0	0
e_6	0	.125	1	.125	0
e_7	0	.125	1	1	0
e_8	0	0	1	0	0
e_9	0	0	1	0	0
e_{10}	0	0	.125	1	0

(d) semantic feature matrix, t=3

Table 7.2 is constructed to show the use of a user-oriented ontology to enhance the semantic representation of the dataset employed in this example. Table 7.2 represents the raw dataset shown in Table 7.1 through the named entities included in the ontology *Royal Family*. Table 7.2 (a)-(d) are different representations of the same data set. Table 7.2 (a) shows the occurrence of the identified named entities; it only

represents direct relations between the information objects. In other words, the correlation of information objects depends on their shared named entities. For example, d_1 and d_5 are linked as they share named entity e_1; d_2 is not correlated to any other information object as there is no shared named entity between it and the rest of the dataset. Table 7.2 (b) shows the semantic feature matrix built with a threshold of t=1. It includes all named entities identified in the information objects, as well as one selected feature (neighbour entity) from the ontology shown in Figure 7.3. It represents direct and indirect relations. For example, d_2 is now linked with all information objects in the dataset through its entity e_3. Table 7.2 (c) and (d) show the semantic feature matrix computed with a threshold of t=2 (all values are $1/2^2$) and 3 (all values are $1/2^3$). As illustrated in Figure 7.3, degree set of the entities in this example is: {deg(e_1)=2, deg(e_2)=2, deg(e_3)=6, deg(e_4)=3, deg(e_5)=3, deg(e_6)=3, deg(e_7)=5, deg(e_8)=3, deg(e_9)=2, deg(e_{10})=1}. Note that the <*friend-of*> relation is ignored for simplicity. Using the degree set, the additional values (all $1/2^t$=$1/2^2$=1/4 in Table 7.2 (b) are determined in the following way:

- d_1: the detected entity is e_1 (it can also be e_2, as they both have the same degree 2), which according to the graph has two neighbours, e_2 and e_3, i.e. its neighbour set is $N_1 = \{e_2, e_3\}$. The threshold t=2 requires two neighbouring entities to be selected. However, one of the two entities is already contained in d_1. Therefore, only e_3 is added to the semantic feature matrix.
- d_2: the detected entity is e_3 and its neighbour set is $N_3 = \{e_1, e_2, e_4, e_5, e_6, e_7\}$. At t=2, e_7 (degree=5) is first selected because it has the highest degree. One more named entity needs to be selected among the three candidates having the same degree (degree=3); these are e_4, e_5 and e_6 and they all carry equal amounts of semantic information. In this particular case, e_7 is selected together with e_6.
- d_3: the neighbour set of the detected entity e_7 is $N_7 = \{e_3, e_4, e_6, e_8, e_{10}\}$. At t=2, the selection principle is similar with the above case, and e_3 (degree=6) is selected as the first neighbour. The second neighbour selection has candidates,

e_4 and e_5 (degree=3). The reason of removing e_6 from the candidates is that e_6 is already contained in d_3. In this example, e_3 and e_4 are selected.

- d_4: the detected entity is e_7 and its neighbour set is $N_7 = \{e_3, e_4, e_6, e_8, e_{10}\}$. At $t=2$, e_3 (degree=6) is selected as the first neighbour. The candidates for the second neighbour selection are e_4, e_6 and e_8 (degree=3). In this example, e_3 is selected with e_6.

- d_5: the only entity of ontology *Royal Family* in it is e_1; its neighbour set is $N_1 = \{e_2, e_3\}$. At $t=2$, e_2 (degree=2) and e_3 (degree=6) are both selected.

Indirect relations indicate the semantic similarity between information objects detected using an ontology. The Onto-SVD approach analyses the information objects using both direct and indirect relations that facilitate semantic similarity measure.

7.4 Experiment and Evaluation

The experiment shown in this section aims to evaluate the topic identification performance using the k-means clustering algorithm in conjunction with the Onto-SVD (*Onto-SVDK*) in comparison with k-means with SVD (*SVDK*). The dataset used in the evaluation is a collection of 2065 high quality English articles. The collection is manually tagged with 8 labelled groups (topics): Health (426 articles), Dementia Disease (420), Olympics (384), Finance (365), British Royal Family (184), FIFA (106), Celebrity (95), and Politics (94). A user-oriented ontology o_p: *Royal Family* is built to guide Onto-SVD in identifying information objects related to the British Royal Family. The relation set selected is $R = \{r_1$: <*spouse-of*>; r_2: <*parent-of*>, r_3: <*sibling-of*>, r_4: <*friend-of*>$\}$. As a relative measure of semantic similarity, the weight of all these connections is set as $W=\{w_1{:}c_0, w_2{:}c_0, w_3{:}c_0, w_4{:}c_0\}$, where c_0 is a positive constant (in this experiment, $c_0=1$, and <*friend-of*>relations are ignored).

In the experiment, truncated SVD factorises the term-information object and semantic feature matrices, and then projects information object vectors into a lower dimensional space. The algorithm then uses cosine similarity to measure the similarity

between information objects based on the position of projections. Then k-means algorithm is applied to classify the information objects, and he output produces a clustering solution of k clusters. The most suitable clustering setting of k-means is k=8, since the data has eight groups. When k=8, the clusters are supposed to match the groups (topics). By comparing the difference between groups and clusters, the experiment could examine the performance of topic identification (clustering) of Onto-SVDK and SVDK.

The evaluation methods employed are listed in Table 7.3.

Table 7.3 Evaluation methods applied in the experiment

Precision	$precision(C_j) = \frac{\mid L_l \cap C_j \mid}{\mid C_j \mid}$
Recall	$recall(C_j) = \frac{\mid L_l \cap C_j \mid}{\mid L_l \mid}$
F-score	$f\text{-}score_{local} = \frac{2 \cdot precision \cdot recall}{precision + recall}$
	$f\text{-}score_{global} = \sum_{l=1}^{k} \frac{\mid L_l \mid}{\mid D \mid} \cdot \max_{1 \le j \le k} \frac{2 \cdot precision(C_j) \cdot recall(C_j)}{precision(C_j) + recall(C_j)}$
Purity	$purity_{local}(C_j) = \frac{1}{\mid C_j \mid} \cdot \max_l(\mid C_j \mid_{label=l})$
	$purity_{global} = \sum_{j=1}^{k} \frac{\mid C_j \mid}{\mid D \mid} \cdot purity(C_j)$
Entropy	$entropy = \sum_{j=1}^{k} \frac{\mid C_j \mid}{\mid D \mid} (-\frac{1}{\log k} \cdot \sum_{l=1}^{k} \frac{\mid L_l \cap C_j \mid}{\mid C_j \mid} \cdot \log \frac{\mid L_l \cap C_j \mid}{\mid C_j \mid})$
Normalised mature information	$NMI = \frac{\sum_{l,j} \mid L_l \cap C_j \mid \cdot \log \frac{\mid D \mid \cdot \mid L_l \cap C_j \mid}{\mid L_l \mid \cdot \mid C_j \mid}}{\sqrt{(\sum_l \mid L_l \mid \cdot \log \frac{\mid L_l \mid}{\mid D \mid})(\sum_j \mid C_j \mid \cdot \log \frac{\mid C_j \mid}{\mid D \mid})}}$

The table uses the following notations: D denotes the dataset; $\mid D \mid$ is the number of information objects; each information object has an initial label describing its topic;

k is the number of labelled groups; L_l and C_j denote information objects in a labelled group l and a cluster j respectively; $|L_l|$ and $|C_j|$ are their total numbers; $|L_l \cap C_j|$ is the number of correctly classified information objects (i.e. documents which are members of both a group L_l and a cluster C_j.

In the context of this experiment, precision is correctly classified information objects $|L_l \cap C_j|$ within a cluster C_j, and recall is the correctly classified information objects $|L_l \cap C_j|$ within a group L_l. F-score is harmonic mean of precision and recall. The local and global F-score evaluate the particular cluster (related with the ontology) and all the clusters respectively. The *local* and *global purity* evaluate the particular cluster (related with the ontology) and all the clusters respectively. *Entropy* represents the uncertainty of the global clustering result. It relies on the summation of cluster entropy and the proportion of cluster size with dataset size, i.e. $\frac{|C_j|}{|D|}$.

The conducted experiment explores the benefit of semantic feature selection, i.e. establishing indirect relations between information objects enhanced by an ontology. As mentioned before, the threshold t is the number of selected neighbouring entities (vertices) of an identified e_h of d_j from G_N. Therefore, the construction of a term–information object (with the selected semantic features) matrix of dataset is dependent on the value of t. In the experiment, t is set as 2, 4, 6 and 8. To avoid overfitting, the maximum threshold is limited to 8. Tables 7.4 and 7.5 list the global and local clustering performance of SVDK with the dataset. For global clustering performance (Table 7.4), when d=100, SVDK has the best purity (0.4610) and F-score (0.4080), as well as acceptable entropy 0.5393 and NMI 0.2462. Moreover, for local clustering performance (Table 7.5), when d=100, SVDK shows adequate precision (0.4018), recall (0.4783) and F-score (0.4367). Dimension 100 is therefore considered as the optimal dimension of SVDK, and the results achieved when d is 100 are used to

compare the performance of the two algorithms. Furthermore, the nearby dimensions are also selected as testing dimensions.

Table 7.4 Clustering performance of SVDK with different dimension (global)

SVDK	***d=80***	***d=90***	***d=100***	***d=110***	***d=120***
Purity	0.4383	0.4281	0.4610	0.4591	0.4368
F-score	0.3730	0.3696	0.4080	0.4027	0.3571
Entropy	0.4667	0.5041	0.5393	0.5418	0.5179
NMI	0.2362	0.2171	0.2462	0.2345	0.2540

Table 7.5 Clustering performance of SVDK with different dimension (local)

SVDK	***d=80***	***d=90***	***d=100***	***d=110***	***d=120***
Precision	0.4077	0.4082	0.4018	0.4546	0.2449
Recall	0.4446	0.5435	0.4783	0.4052	0.6467
F-score	0.4254	0.4662	0.4367	0.4285	0.3553

To compare with the optimal result of SVDK, the dimension of Onto-SVDK is set as SVDK's optimal dimension d=100. Table 7.6 shows the impact of the semantic feature selection on the global clustering performance, when the dimension is 100. When the threshold t equals 2, 4, 6, 8, Onto-SVDK displays different performance. As expected, all the results of Onto-SVDK outperform the optimal result of SVDK. Onto-SVDK has its best performance purity (0.7698), F-score (0.7910), entropy (0.4069) and NMI (0.5484) when the threshold is 4. In other words, within the same dimension, when a different number of semantic features is used, the topic identification performance of Onto-SVDK is better than SVDK.

Table 7.7 shows that local clustering performance has a high recall (0.9239), when threshold t=2, but a low precision (0.3872.) In addition, Onto-SVD identifies 92.39%

of the information objects related to "Royal Family", but some information objects within other topics are not identified correctly. When t=4, Onto-SVDK also reaches its best local clustering performance (precision 0.6693, recall 0.9239 and F-score 0.7763). In case of t={6, 8}, the performance slightly deteriorates. The results indicate that the clustering performance is not directly proportional to the value of the threshold t. This can be explained by the following two reasons. Firstly, some information objects that belong to different topics have shared content, and thus a low threshold is not sufficient to enhance the semantic representation of the information objects to a proper level. This indicates that the information objects are still undistinguishable semantically (see the local clustering result when t=2). Secondly, a high threshold causes overfitting, which involves using excess of features in cases of semantic ambiguity, e.g. when the same name is shared by different persons. If such type of semantic features are overly utilised, the possibility of bringing irrelevant information objects to a cluster will be significant (see the local clustering result when t=8).

Table 7.6 Onto-SVDK with different threshold t when d=100 (global)

Onto-SVDK	***t=2***	***t=4***	***t=6***	***t=8***
Purity	0.7036	0.7698	0.6261	0.6015
F-score	0.7327	0.7910	0.5664	0.5423
Entropy	0.3556	0.4069	0.4451	0.4693
NMI	0.4846	0.5484	0.4278	0.4207

Table 7.7 Onto-SVDK with different threshold t when d=100 (local)

Onto-SVDK	***t=2***	***t=4***	***t=6***	***t=8***
Precision	0.3872	0.6693	0.5339	0.5401
Recall	0.9239	0.9239	0.7283	0.8043
F-score	0.5457	0.7763	0.6161	0.6462

The performance of local clustering is assessed by the results obtained from the same tests (i.e. a dimension d=100 and thresholds t={2, 4, 6, 8}), the local clusters generated by the two algorithms SVDK and Onto-SVDK are examined in terms of number of information objects from different topics (e.g., "Finance", "Dementia", etc.) included in the local clusters (i.e. related to the "Royal Family"). Table 7.8 shows that clusters generated by the SVDK include 219 information objects, 88 out of which have been originally tagged as "Royal Family". However, this cluster also contains 3 information objects from "Finance", 1 from "FIFA", etc. Figure 7.4, shows plots which are produced using data from Table 7.8. From the same figure it can be concluded that the local clustering performance of Onto-SVDK is better than SVDK, especially when t={4, 6, 8}. Table 7.8 and Figure 7.4 demonstrate that it is hard to distinguish some of the topics, e.g. some of the information objects in clusters "Politics", "Celebrity" and "Olympics" since they are also related to the "Royal Family". In addition, this is the reason why SVDK fails to identify the semantic difference those information objects. Onto-SVDK employs ontology to enhance the semantic representation of information objects related to the "Royal Family", and thus it successfully establishes existing relations between the information objects which addresses the problem well.

Table 7.8 Details of the local cluster, when dimension d=100 (local)

	SVDK	**Onto-SVDK**			
		***t*=2**	***t*=4**	***t*=6**	***t*=8**
Finance	3	5	2	2	2
Dementia	0	1	2	2	10
FIFA	1	0	7	17	18
Health	7	19	7	7	7
Politics	36	63	16	17	17
Olympics	24	93	24	47	47
Celebrity	60	89	26	25	25
Royal Family	88	169	170	134	148
Total	219	439	254	251	274

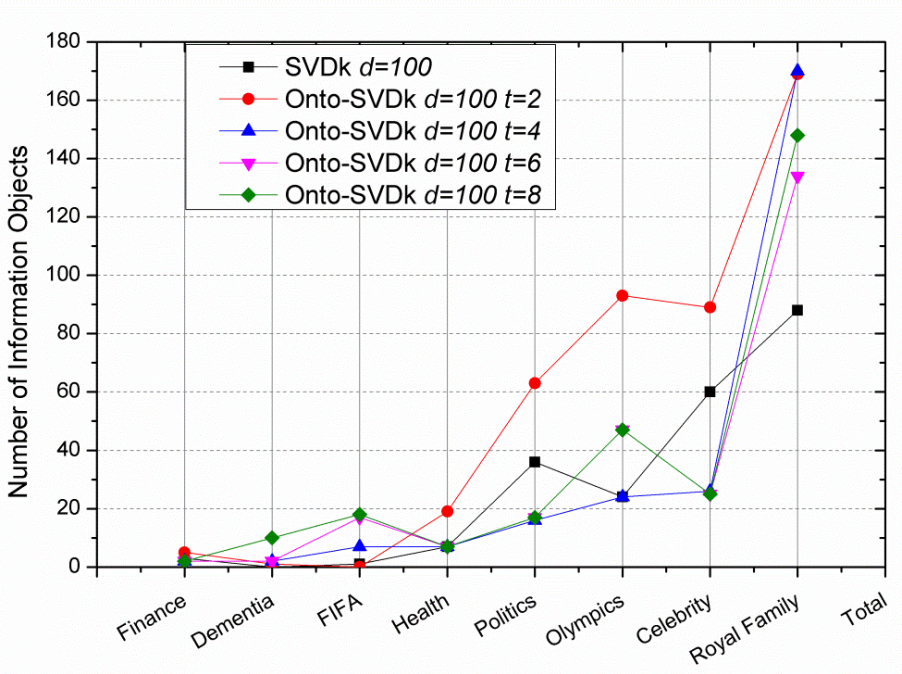


Figure 7.4 Details of the local cluster, when d=100 (local)

As mentioned before, Onto-SVDK has an optimal threshold of t=4, when the dimension d=100. To compare the global and local clustering performance of Onto-SVDK and SVDK with different dimensions, the threshold of Onto-SVDK is fixed to t=4. Table 7.9 shows the global clustering performance of Onto-SVDK when tested using different dimensions, from 80 to 120. The comparison with Table 7.4 shows that the performance of Onto-SVDK outperforms SVDK with all the testing dimensions. In other words, within a set of selected semantic features, testing with different dimensions, the topic identification performance of Onto-SVDK is better than SVDK. Figure 7.5 shows that the performance improvement of Onto-SVDK over SVDK with different dimensions, and the average improvement is purity 32.82%, F-score 41.62%, entropy (reduced) 21.38% and NMI 32.36%.

Table 7.9 Onto-SVDK with different dimensions, when t=4 (global)

Onto-SVDK	*d=80*	*d=90*	*d=100*	*d=110*	*d=120*
Purity	0.8029	0.7690	0.7698	0.7603	0.7622
F-score	0.8435	0.7932	0.7910	0.7757	0.7881
Entropy	0.3951	0.4325	0.4069	0.3878	0.3917
NMI	0.6148	0.5472	0.5484	0.5475	0.5479

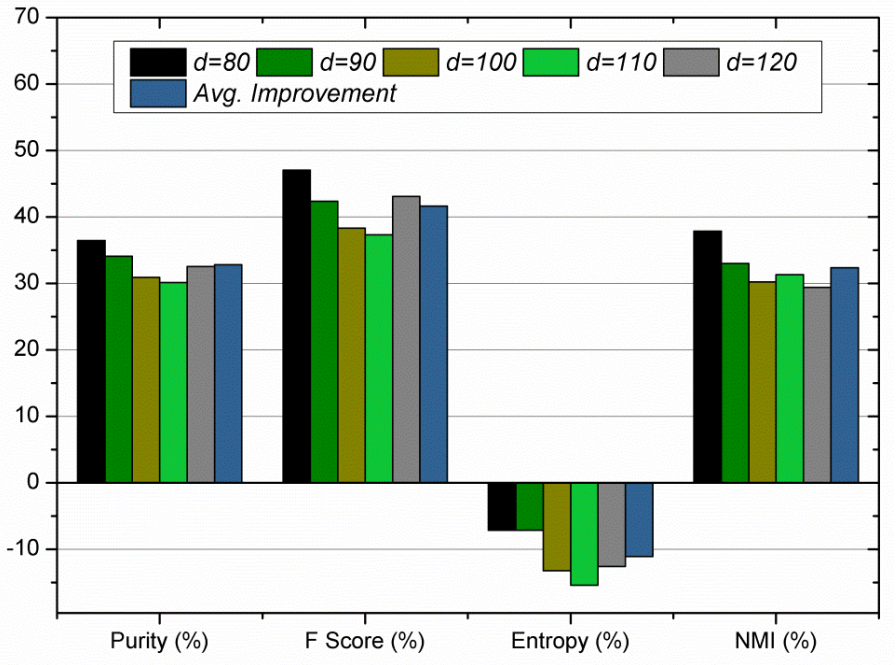


Figure 7.5 Clustering performance improvement of Onto-SVDK (global)

Table 7.10 shows the local clustering performance of Onto-SVDK, with different dimension from 80 to 120, when t=4. The comparison with Table 7.5 shows that the performance of Onto-SVDK outperforms SVDK. The best local clustering performance of Onto-SVDK is, precision 0.6883, recall 0.9239 and F-score 0.7889. Figure 7.6 represents the performance improvement of Onto-SVDK over SVDK, and the average improvement is precision 29.37%, recall 41.90% and F-score 35.86%.

Table 7.10 Onto-SVDK with different dimensions, when t=4 (local)

Onto-SVDK	*d=80*	*d=90*	*d=100*	*d=110*	*d=120*
Precision	0.6842	0.6693	0.6693	0.6883	0.6746
Recall	0.9185	0.9231	0.9239	0.9239	0.9239
F-score	0.7842	0.7760	0.7763	0.7889	0.7798

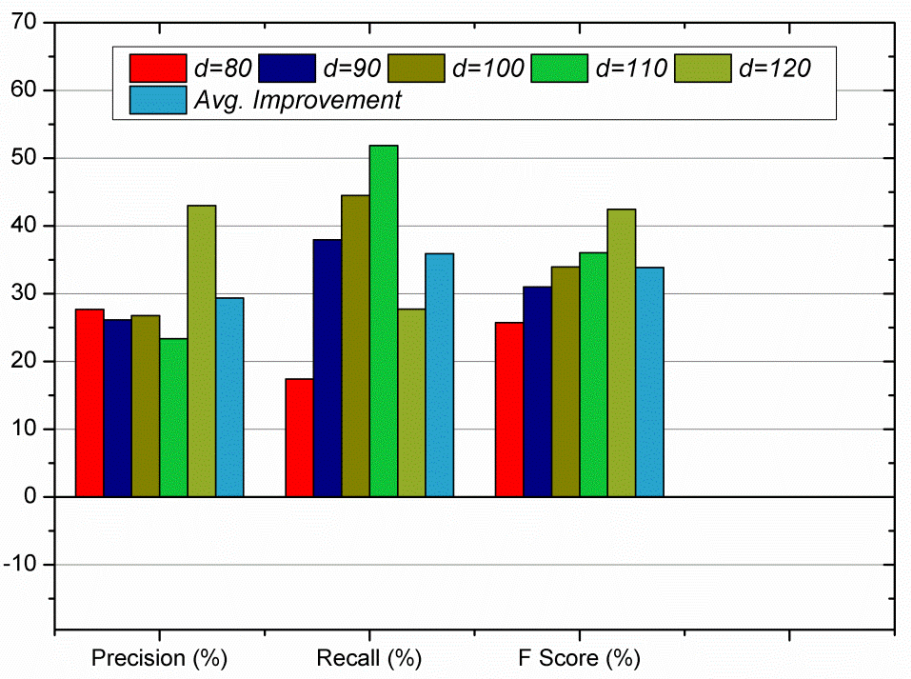


Figure 7.6 Clustering performance improvement of Onto-SVDK (local)

Figure 7.7 shows part of the clustering result of Onto-SVD when tested with ontology formalising relation with regards to the "Royal Family". It displays 4 topics, namely "Royal Family", "Politics", "Finance" and "Celebrity". The result shows that some of the information objects are not related to the topic "Royal Family" but still share similar content, thus they can be successfully identified by Onto-SVD as related to each other. This approach facilitates the information management process of Sem-LSB system. In addition, the cluster-based data also provides a story-like retrieval mechanism for the system. For example, if there is a query about Queen Elizabeth II, the retrieval mechanism is able to generate a result from the different clusters

(topics), with regards to her family, significant politics events, dignitaries, etc. The clusters (topics) are not independent from each other; therefore it is possible to generate a retrieval result in the form of a story.

Figure 7.7 Cluster result by Onto-SVD

7.5 Summary

The Sem-LSB employs a user-oriented ontology that in an interactive pattern enables users to apply their background knowledge. The ontology supports the system to understand the user's personal life events on semantic basis, which leads to improved performance in terms of correctly establishing connections between objects, events and facts. Onto-SVD provides an advanced topic identification approach relying on user-oriented ontologies. It uses terms and essential semantic features to establish connections between similar information objects. Due to the indirect connections identified with the help of the ontology, Onto-SVD is more useful for topic identification and semantic similarity than the traditional SVD based method.

The evaluation shows that the topic-based clustering performance of Onto-SVD outperforms the SVD-based method, the average improvements of global clustering performance are purity 32.82%, F-score 41.62%, entropy (reduced) 21.38% and NMI 32.36%. With regard to the cluster produced using a user-oriented ontology, the average improvements of the local clustering performance are precision 29.37%, recall 41.90%, F-score 35.86%. This means that Onto-SVD has proven its ability to distinguish information objects based on their topics, and successfully detects (on semantic basis) the indirect relations between them.

Chapter 8

Ontology-Based Personalised Retrieval

This chapter proposes an ontology-based personalised retrieval mechanism for Sem-LSB, to achieve its dynamic content generation, for instance, providing customised retrieval results to different persons based on their background. The chapter is organised as follows. Section 8.1 introduces the main modules and operation of the mechanism. Section 8.2 formally defines a multiple ontology model for the mechanism named a user profile space. A tree-based personalised retrieval algorithm with a user-oriented ontology and user profile space is proposed in Section 8.3. Section 8.4 shows the experimental evaluation of the proposed algorithm. Section 8.5 summarises the chapter.

8.1 Modules and Process

The modules and operation of the proposed mechanism are shown in Figure 8.1. During a search session, the user provides queries, each of which can contain semantic features or not. The query is processed by the pre-processing module, which conducts automatic spelling correction and text processing. The spelling correction function aims to assist the users, especially the elderly and those with cognitive impairment. The NLP module performs tokenisation, stemming and stop-words removal. The semantic feature extraction module uses automatic named entity recognition to extract semantic features from the query. If an extracted semantic feature cannot be located in the ontology, it is collected and stored, and further human decision would be needed during/after the search session. The semantic feature selection is the key module that selects the relevant semantic features from the ontology, and then sends them to the query expansion module. The expanded query is passed to the data retrieval module. Next, the retrieved result is grouped by the clustering module. The

final result is cluster-based with a story-like format that facilitates users' browsing and re-discovering their life events. The relevance feedback module collects explicit and implicit feedback which is further applied to enrich the knowledge base. The explicit feedback collection is a manual process based on direct submissions from the users. The implicit feedback collection is an automatic process based on analysing user behaviour (e.g. clicking sequence, browsing/dwell time), and pseudo-relevance feedback, which is not explicitly semantically related.

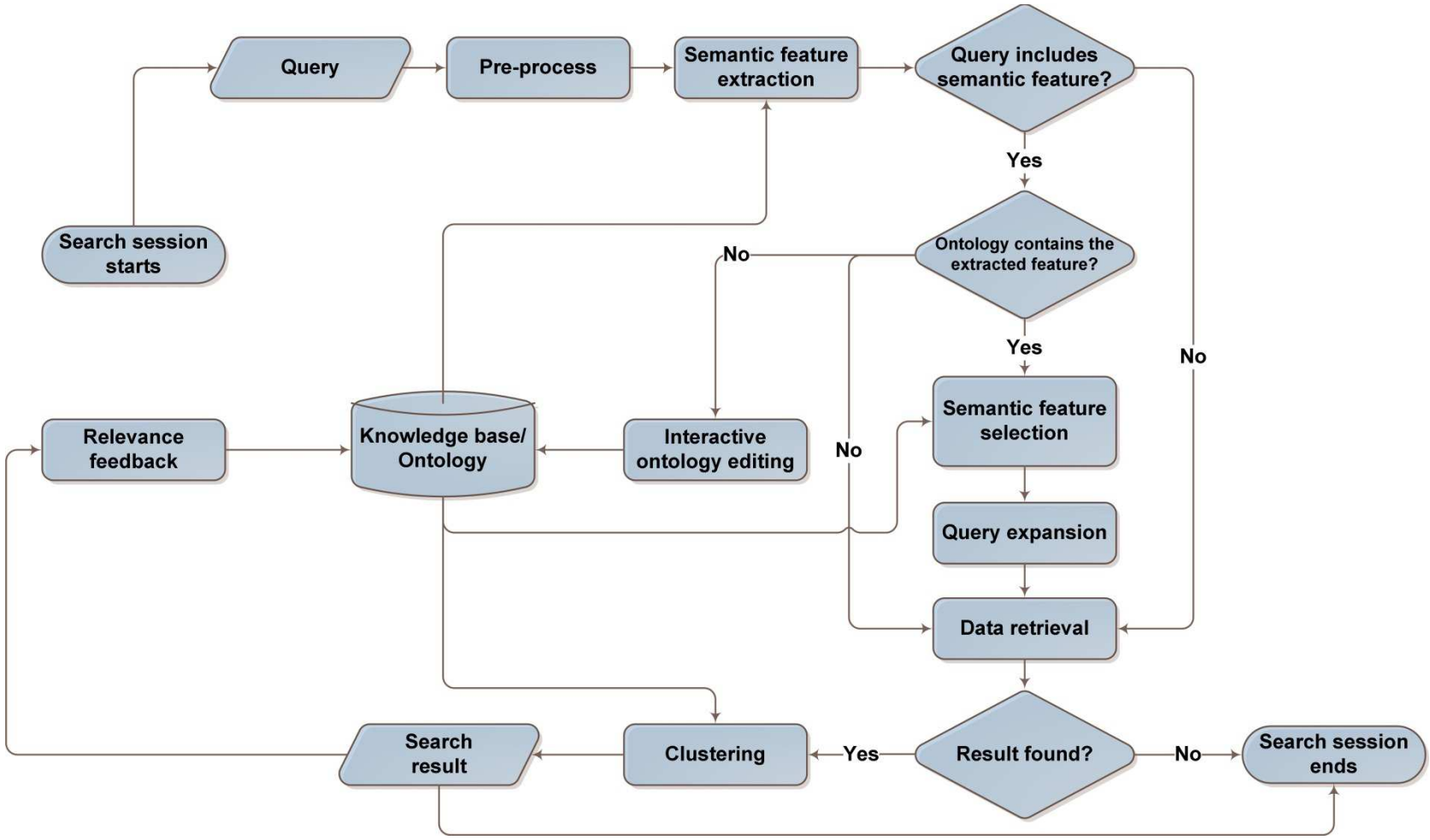


Figure 8.1 Modules and operation of the personalised retrieval mechanism

8.2 User Profile Space

The user-oriented ontology is domain-dependent, thus a single ontology is not able to handle cross-domain knowledge. For example, a query "*Queen Elizabeth II*" might be implicitly related to places and historical dates, such as "*Buckingham Palace*", "*Windsor Castle*", "*2 June 2012*", and these features cannot be selected from a single user-oriented ontology.

To achieve cross-domain search in personalised retrieval, a multi-ontology model, named a user profile space (UPS) is formally proposed, which integrates all user-oriented ontologies of a single user, thus providing more comprehensive knowledge coverage than the single ontology. Figure 8.2 shows the relation diagram of this model. It contains four components: user profile space, ontology, user feedback and semantic feature.

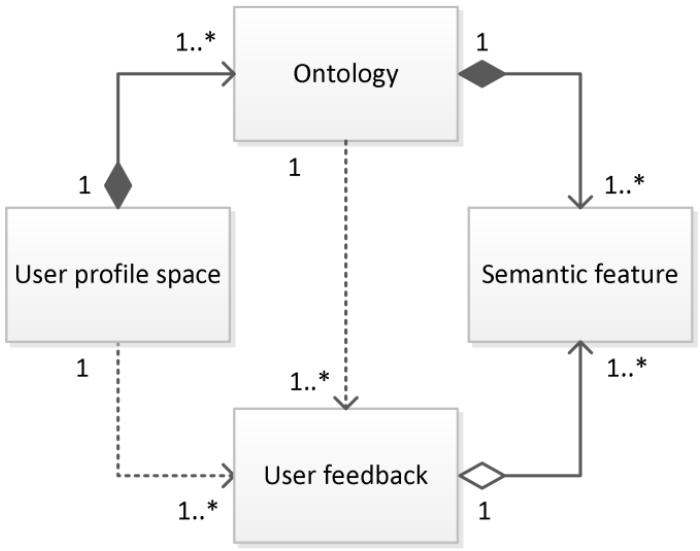


Figure 8.2 Construction of a user profile space

The relation between the semantic features and ontology is *composition relationship* with *many-to-one* attribute. It means that any single ontology consists of many semantic features, and the semantic feature is a component of the ontology. Similarly, *composition relationship* is used between the ontologies and UPS with

many-to-one attribute. The user feedback reflects the user's satisfaction with the previous retrieval results. It has *aggregation relationship* between semantic features with *one-to-many* attribute. Furthermore, the user feedback impacts on the ontology editing and UPS construction, thus the ontology and UPS both have *dependency relationship* with the user feedback. This means that if any new user feedback is generated, the ontology and UPS could be changed as a result.

8.3 Knowledge Spanning Tree Generation

The scale of the ontology employed influences the computational complexity of the system. The complexity is reduced by converting an ontology to ontology graph, and the graph structure can be further simplified by generating knowledge spanning trees.

In a search session, the semantic feature selection retrieves relevant semantic features from the ontology or user profile space, and then these features are used to generate various sub-trees. During the query expansion, each sub-tree can be used as a bundle of related semantic features with relations, which are applied to reform and extend user's search query. This section discusses in more detail the developed knowledge spanning tree generation, graph-based semantic feature selection, and query expansion with the generated knowledge spanning trees.

Let $G_N = (V, E)$ denote the graph of user-oriented ontology o_N, the vertices V and edges E represent the ontology concepts and relations, respectively. $|V|$ is the number of vertices that describes the semantic feature capability of o_N, i.e. a large $|V|$ indicates a rich ontology with many semantic features. $|E|$ is the number of edges indicating the knowledge structure complexity of o_N, e.g. two ontologies may have the same number of vertices $|V|$ but a different number of edges $|E|$. The one with greater edge number is considered as having a more complicated knowledge structure than the other.

Let $edge\_dist(v_a, v_b)$ denote the edge distance between vertices v_a and v_b, the similarity between any two ontology concepts is measured through the edge distance, which is based on formula 5.2,

$$edge\_dist(v_a, v_b) = \frac{1}{\ln(w_{r(a,b)} + e)}, \tag{8.1}$$

where $w_{r(a,b)}$ is a non-negative number representing the relation weight of v_a and v_b. As mentioned before, the distance is monotonically decreasing as $w_{r(a,b)}$ increases.

8.3.1 Knowledge Spanning Tree in a Single User-Oriented Ontology

In personalised retrieval, the semantic feature(s) contained in a query is the essential cue in identifying the search intention (Guo et al., 2009). Furthermore, the semantic representation with personalisation of the query can be enhanced by the user-oriented ontology. The semantic feature contained in the query is extracted using automatic named entity recognition and knowledge bases. If the extracted feature can be located in a user-oriented ontology, then the semantic feature selection module selects more relevant features from the ontology, which then are considered as expansion terms of the query. There are two crucial factors in the process: (i) the user search intention needs to be first identified based on the query content; and (ii) the selected features should be highly correlated with the search intention identified.

Given a query q and ontology o_N, the extracted semantic features are represented as $\{s_1, s_2, \ldots, s_n\} \in q \cap o_N$. The term importance is typically measured using term weight normalisation methods such as TF-IDF or Okapi BM25. However, these approaches cannot properly assess the importance of the semantic feature in the query, as the query has very limited context. The solution is to measure the term importance using knowledge from the ontology. Based on the ontology graph, the information entropy of an ontology concept is measurable, which indicates the amount of semantic information carried within the concept. Similar to information objects, the

semantic feature of a query is linked to the ontology concept, thus the concept entropy indicates the significance of the semantic feature in the query. The concept entropy of s_i with G_N is represented as,

$$H(s_i)_{s_i \in o_N} = -\sum_{i=1}^{n} p(s_i) \log p(s_i), \tag{8.2}$$

where n is the edge number of s_i; $p(s_i)$ denotes the probability of selecting s_i from any of its adjacent vertices. Higher concept entropy indicates that the concept carries more semantic information. Semantic feature with the highest concept entropy is considered as the most important one for the query. It is named as *identified* feature, and denoted by s_h,

$$\arg\max_{s_n \in q, o_N} H(s_n) \rightarrow s_h, \tag{8.3}$$

where s_h, o_N are utilised to represent the search intention of the query.

The semantic feature selection then selects more features based on the identified feature. The process is to get the nearby vertices of s_h from G_N, and the criterion used is the short edge distance principle[1]. Let $N_h = \{n_1, n_2, ..., n_i\}$ denote the adjacent vertex set of s_h, and $N_{h,k}$ denotes selected k features from N_h based on s_h, where $N_{h,k} \subseteq N_h$. Then the selected feature set satisfies,

$$\arg\min \sum_{n_i \in N_h}^{k} edge\_dist(s_h, n_i) \rightarrow N_{h,k}. \tag{8.4}$$

In accordance with the short edge distance principle, the process of semantic feature selection is similar to the generation of the minimum spanning tree (MST) of the ontology graph. Based on graph theory, the minimum spanning tree is defined as the tree of a graph, which connects all the vertices with the minimum edge weight

[1] It means that the selected k vertices have shorter edge distances with s_h than the other vertices. If there are several edges between the two vertices, only the one with the shortest distance needs to be considered.

cost. The primary difference between G_N and its minimum spanning tree MST_N is the knowledge structure complexity $|E|$. As shown in Figure 8.3, all vertices of G_N remain in MST_N, but the minor edges are not included. The edges in MST_N indicate the essential relations among the ontology concepts, thus it can be treated as an optimisation of the ontology. MST_N has a lower knowledge structure complexity, and therefore its application in the semantic feature selection could reduce the computational cost of the algorithm.

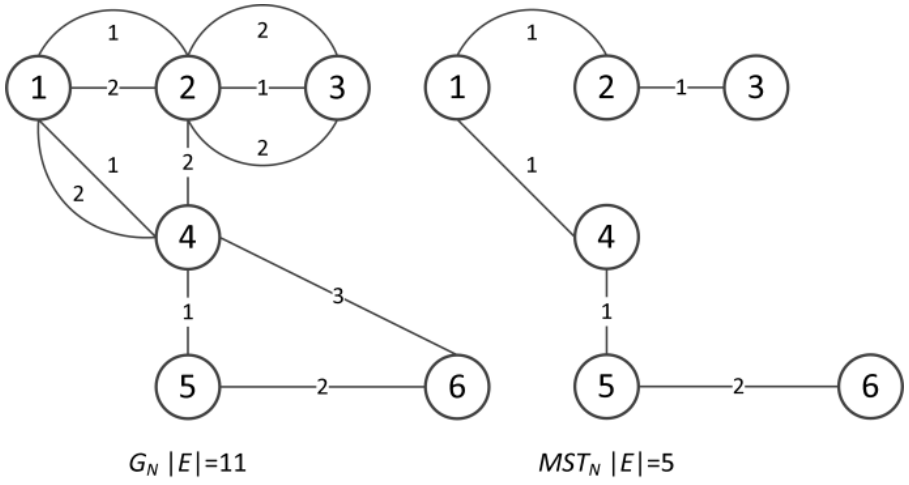


Figure 8.3 An ontology graph and its Minimum Spanning Tree (MST)

However, the minimum spanning tree may not be suitable for some retrieval requirements. In the case of a large ontology o_N, the user's search intentions always refer to certain ontology concepts, rather than the entire ontology. The use of MST_N could involve an excessive number of semantic features causing over-fitting problem. Using the example from Figure 8.3, if s_2 represents an identified feature of a query, the concept s_6 has lower concept similarity with s_2 than the others and therefore its probability of bringing relevant results for s_2 is also lower. To prevent this, the quantity of selected features should be controllable.

Considering this issue, an improved approach is proposed in this work which uses the concept of the K-minimum spanning tree (KMST). Giving an ontology o_N, a

query with the identified feature s_h, the semantic feature selection is to generate a KMST of the ontology graph. In graph theory, the KMST is defined as a tree of G_N, which contains k vertices and $k-1$ edges with the minimum edge weight cost. The example shown in Figure 8.4 is converted from the family tree in Figure 8.7; it includes 16 named entities and 3 relation types. To simplify the example, the relation weights are set as before. Figure 8.4 shows six K-minimum spanning trees of identified feature s_9, with $k=5$.

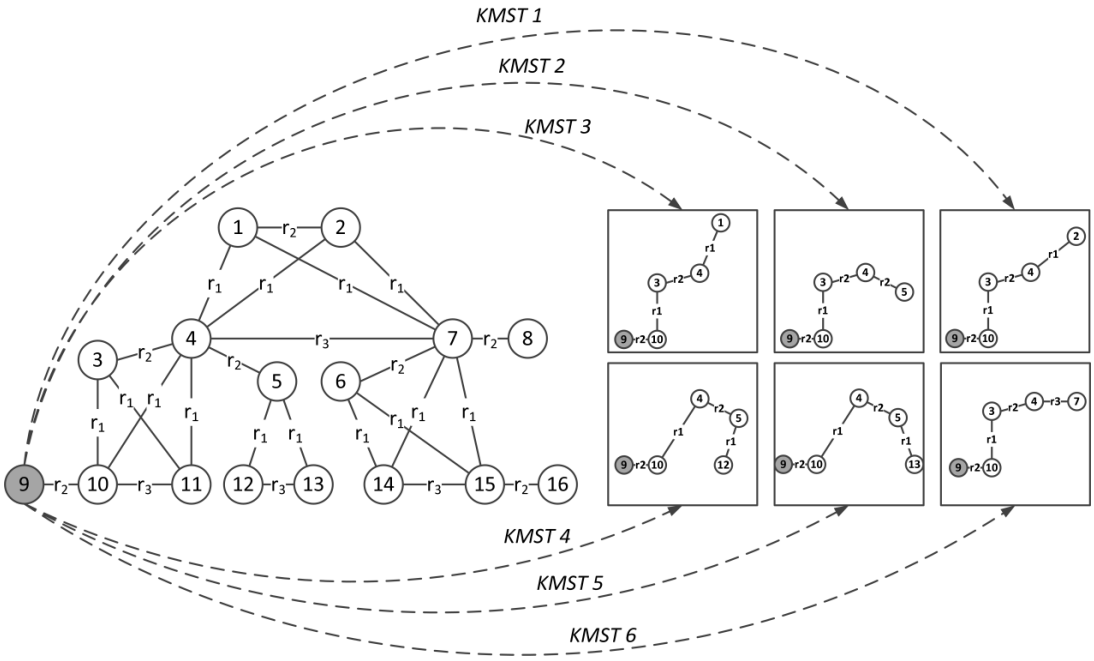


Figure 8.4 Weighted undirected ontology graph and K-minimum spanning trees of s_9, with k=5

Let $t_{h,k}$ denote a tree, where $s_h \in t_{h,k}$ and $s_h \in o_N$. Based on G_N, the generation of the KMST is,

$$\arg\min cost(t_{h,k}) = \sum_{i=1}^{k-1} edge_i \rightarrow KMST_{h,N} \tag{8.5}$$

where k is the number of vertex in $t_{h,k}$.

As shown in Figure 8.4, the ontology graph contains identical edge distances, thus the KMST with a given vertex could not be unique. It consists of vertices and edges,

and different combinations of vertices and edges determine different semantic representations. The strength of the semantic representation of a K-minimum spanning tree depends on the amount of the semantic information carried through its contained ontology concepts, which is measured by cumulative concept entropy,

$$H(KMST_{h,k}) = -\sum_{i=1}^{k}\sum_{j=1}^{n} p(s_{i,N}) \log p(s_{i,N}) \tag{8.6}$$

where k is the number of vertices in $KMST_{h,k}$.

The $KMST_{h,k}$ with high cumulative concept entropy indicates strong semantic representation, thus the result of the semantic feature selection is the KMST with the highest cumulative concept entropy. For example, in Figure 8.4, the cumulative concept entropy of the K-minimum spanning trees are sorted, i.e.

$H(KMST\ 4) = H(KMST\ 5) = 2.225 < H(KMST\ 1) =$
$H(KMST\ 2) = H(KMST\ 3) = 2.401 < H(KMST\ 6) = 2.769.$

Therefore, if a query contains s_9, the selected relevant 5 semantic features can be represented by $KMST\ 6 = \{s_9, s_{10,} s_3, s_4, s_7\}$.

8.3.2 Spatial Knowledge Spanning Tree in the User Profile Space

The user-oriented ontology is the component of user profile space, and the parallel ontologies in UPS are connected by their inner relationships. The structure depends on the semantic feature correlation and ontology correlation. These two concepts are explained in more details next.

Let $O = \{o_1, o_2, ..., o_N\}$ denote an ontology set, $X = \{x_1, x_2, ..., x_n\}$ represents a training sample set. Each training sample $x_n = \{w_1, w_2, ..., w_i; s_1, s_2, ..., s_j\}$ is the feedback that includes arbitrary words and semantic features. The essential information object of the user's significant experience is used as explicit feedback.

The semantic feature correlation measure identifies the correlation of the semantic features which are located in different ontologies but are included in the same training

samples. The measure is based on the semantic feature co-occurrence in the training samples. For example, two features are considered as correlated if both of them are contained in one or more training samples. The correlation value is in direct proportion to the co-occurrence rate of the features in the training samples, and the number of training samples containing the features. The former indicates the importance of the feature for the particular training sample. In addition, the latter indicates the importance of the features combination for the entire training set. The semantic feature correlation between s_a and s_b is represented as,

$$\vartheta(s_a, s_b) = \sum_{i=1}^{n} \frac{| s_a \in x_i || s_b \in x_i |}{n \cdot \|x_i\|^2} \tag{8.7}$$

where $s_a \in o_M$, $s_b \in o_N$, $o_M \neq o_N$; n is the number of the training sample in X ; $| s_a \in x_i |$ is the occurrence rate of s_a in x_i ; $| s_a \in x_i |$ is the occurrence rate of s_b in x_i ; $\|x_i\|$ is the length of the vector of x_i. For example, a sample UPS is constructed by four ontologies and its graph representation is shown in Figure 8.5. In the figure, s_{2,o_1} and s_{5,o_2} are correlated, as they are contained in the same training sample x_1. The semantic feature correlation of s_{2,o_1} and s_{5,o_2} is 0.125. Similarly, s_{4,o_1}, s_{3,o_2} and s_{2,o_3} are correlated, as they are contained in x_2 .

The ontology correlation depends on the semantic feature correlation, and its value is in direct proportion to the number of correlated semantic features contained in the ontologies. For ontologies o_M and o_N , if $s_a \in o_M$, $s_b \in o_N$ and $\vartheta(s_a, s_b) > 0$, then these ontologies are correlated. Let $S_{M,N}$ represent the correlated semantic feature contained in o_M and o_N , and the ontology correlation is represented as,

$$\varphi(o_M, o_N) = \frac{| S_{M,N} |}{| o_M | + | o_N |} \tag{8.8}$$

where $| S_{M,N} |$ is the number of correlated semantic features in $S_{M,N}$. $| o_M |$ and $| o_N |$ is the number of vertices in o_M and o_N , respectively. For example, in Figure 8.5, o_1 and o_2 are correlated, where $S_{o_1,o_2} = \{s_{2,o_1}, s_{4,o_1}, s_{3,o_2}, s_{5,o_2}\}$; o_2 and o_3 are correlated,

where $S_{o_2,o_3} = \{s_{3,o_2}, s_{2,o_3}\}$. The ontology correlation of o_1 and o_2 is 0.400, and the ontology correlation of o_2 and o_3 is 0.250.

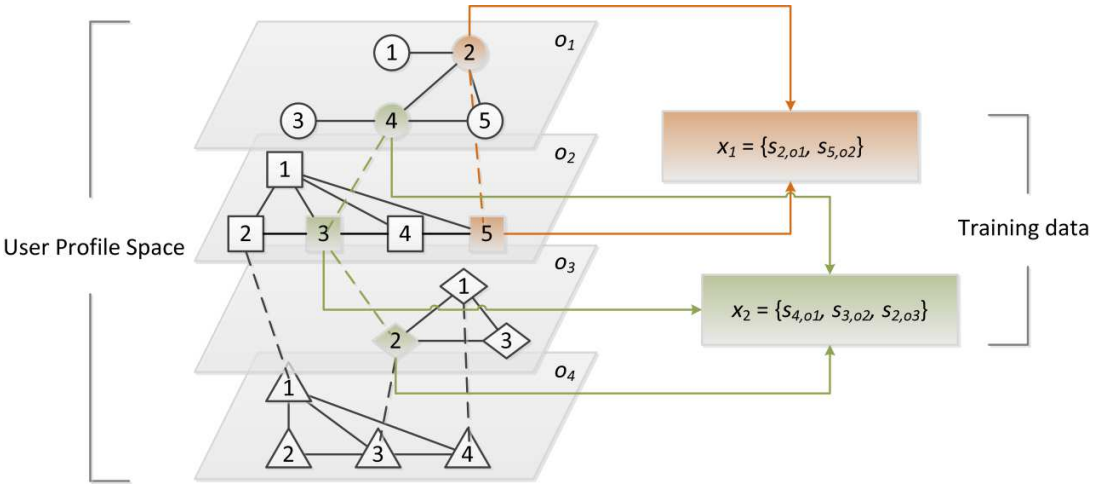


Figure 8.5 Semantic feature correlation and ontology correlation

A relationship between the ontologies in a UPS is named in this research *spatial edge*. It is established by the correlated features from different ontologies. The spatial edges distance is in inverse proportion to the semantic feature correlation, and it is represented as,

$$spatial\_edge\_dist(s_a, s_b) = \vartheta(s_a, s_b)^{-1} \tag{8.9}$$

where s_a, s_b are the correlated feature establishing the spatial edge.

UPS can be converted to a graph denoted by G_{UPS}, which contains vertices, edges of the ontologies and spatial edges between the ontologies. Similar to the minimum spanning tree of the ontology graph, the UPS graph also has a simplified knowledge structure, and it is represented by the spatial minimum spanning tree (sMST). In this work, a sMST of G_{UPS} is defined as a tree that contains all the vertices of the ontologies with the minimum edge weight cost and spatial edge cost. In other words, sMST is the simplified knowledge structure of a UPS, which only contains the

minimum spanning trees of the contained ontologies and the spatial edges between the ontologies.

Given a UPS consisting of N ontologies, with an identified feature $s_h \in o_N$, then the construction process of the sMST is as follows,

- generate the minimum spanning tree of o_1:
  $o_1 \rightarrow MST_1 = \{s_{1,o_1}, s_{2,o_1}, ..., s_{n,o_1}\}$;
- if $\exists \vartheta(s_{a,o_1}, s_{b,o_2}) > 0 \rightarrow \varphi(o_1, o_2) > 0$, o_1, o_2 are correlated;
- generate a spatial edge, $\vartheta(s_{a,o_1}, s_{b,o_2}) > 0 \rightarrow spatial\_edge(s_{a,o_1}, s_{b,o_2})$;
- generate the spatial edge set of o_1, o_2:
  $spatial\_edges(o_1, o_2) =$
  $\{spatial\_edge(s_{a,o_1}, s_{b,o_2}), ..., spatial\_edge(s_{i,o_1}, s_{j,o_2})\}$;
- generate the minimum spanning tree of o_2:
  $MST_2 = \{s_{1,o_2}, s_{2,o_2}, ..., s_{n,o_2}\}$;
- iterate ...;
- if $\exists \vartheta(s_{a,o_{N-1}}, s_{b,o_N}) > 0 \rightarrow \varphi(o_{N-1}, o_N) > 0$, o_{N-1}, o_N are correlated;
- generate a spatial edge: $\vartheta(s_{a,o_{N-1}}, s_{b,o_N}) > 0 \rightarrow spatial\_edge(s_{a,o_{N-1}}, s_{b,o_N})$;
- generate the spatial edge set of o_{N-1}, o_N:
  $spatial\_edges(o_{N-1}, o_N) =$
  $\{spatial\_edge(s_{a,o_{N-1}}, s_{b,o_N}), ... spatial\_edge(s_{i,o_{N-1}}, s_{j,o_N})\}$;
- generate the minimum spanning tree of o_N: $MST_N = \{s_{1,o_N}, s_{2,o_N}, ..., s_{n,o_N}\}$;
- generate the spatial minimum spanning tree of the UPS:
  $sMST =$
  $(MST_1 ... \cup MST_N) \cup spatial\_edges(o_1, o_2) ... \cup spatial\_edges(o_{N-1}, o_N)$,

where $spatial\_edges(o_{N-1}, o_N)$ represents the spatial edge set between the ontologies, and the spatial edges could be multiple.

In cross-domain search, semantic features are selected from UPS. To control the quantity of them, the feature selection process is based on a spatial K-minimum spanning tree (sKMST). In this research, sKMST is defined as a tree of G_{UPS} that

contains K vertices with minimum edge weight cost. Let $K = \{k_1, k_2, ..., k_N\}$ denote a set of tuning parameters, one of which is the number of selected features from a particular ontology, and it constrains the number of vertices of the particular ontology's KMST. $| K |= \sum_{i=1}^{N} k_i$ is the maximum number of selected features from the UPS. As mentioned above, the spatial edge between two ontologies in sMST can be multiple, however, in sKMST, there is only one spatial edge between any two ontologies, and it is established based on the pair of semantic features with the highest feature correlation.

Given a UPS consisting of N ontologies, with an identified feature $s_h \in o_N$ and $K = \{k_1, k_2, ..., k_N\}$, the construction process of the sKMST is as follows,

- if $s_h \in o_N$, generate the K-minimum spanning tree of o_1:
  $o_1 \rightarrow KMST_1 = \{s_h, s_{2,o_1}, \ldots, s_{k_1,o_1}\}$;
- if $\exists \vartheta(s_{a,o_1}, s_{b,o_2}) > 0 \rightarrow \varphi(o_1, o_2) > 0$, o_1, o_2 are correlated;
- generate a spatial edge, $\vartheta(s_{a,o_1}, s_{b,o_2}) > 0 \rightarrow spatial\_edge(s_{a,o_1}, s_{b,o_2})$;
- generate the spatial edge set of o_1, o_2.
  $spatial\_edges(o_1, o_2) =$
  $\{spatial\_edge(s_{a,o_1}, s_{b,o_2}), \ldots, spatial\_edge(s_{i,o_1}, s_{j,o_2})\}$;
- if the shortest spatial edge in $spatial\_edges(o_1, o_2)$ is $spatial\_edge(s_{i,o_1}, s_{j,o_2})$, then $spatial\_edge(s_{i,o_1}, s_{j,o_2})$ is the unique spatial edge of o_1, o_2, and s_{j,o_2} is treated as the identified semantic feature of o_2;
- with s_{j,o_2}, generate the K-minimum spanning tree of o_2:
  $o_2 \rightarrow KMST_2 = \{s_{j,o_2}, s_{2,o_2}, s_{3,o_2}, \ldots, s_{k_2,o_2}\}$;
- iterate ...;
- if $\exists \vartheta(s_{a,o_{N-1}}, s_{b,o_N}) > 0 \rightarrow \varphi(o_{N-1}, o_N) > 0$, o_{N-1}, o_N are correlated;
- generate a spatial edge, $\vartheta(s_{a,o_{N-1}}, s_{b,o_N}) > 0 \rightarrow spatial\_edge(s_{a,o_{N-1}}, s_{b,o_N})$;
- generate the spatial edge set of o_{N-1}, o_N:

$spatial\_edges(o_{N-1}, o_N) =$
$\{spatial\_edge(s_{a,o_{N-1}}, s_{b,o_N}), \ldots, spatial\_edge(s_{i,o_{N-1}}, s_{j,o_N})\}$;

- if the shortest spatial edge in $spatial\_edges(o_{N-1}, o_N)$ is $spatial\_edge(s_{i,o_{N-1}}, s_{j,o_N})$, then $spatial\_edge(s_{i,o_{N-1}}, s_{j,o_N})$ is the unique spatial edge of o_{N-1}, o_N, and s_{j,o_N} is treated as the identified semantic feature of o_N;
- with s_{j,o_N}, generate the K-minimum spanning tree of o_N;
  $o_N \rightarrow KMST_N = \{s_{j,o_N}, s_{2,o_N}, s_{3,o_N}, \ldots, s_{k_N,o_N}\}$;
- generate the spatial K-minimum spanning tree of the UPS:
  $sKMST = KMST_1 \ldots \cup KMST_N \cup$
  $spatial\_edge(s_{i,o_1}, s_{j,o_2}) \ldots \cup spatial\_edge(s_{i,o_{N-1}}, s_{j,o_N})$,

where $spatial\_edge(s_{i,o_{N-1}}, s_{j,o_N})$ represents the shortest spatial edge between the ontologies.

Figure 8.6 shows an example of sKMST based semantic feature selection.

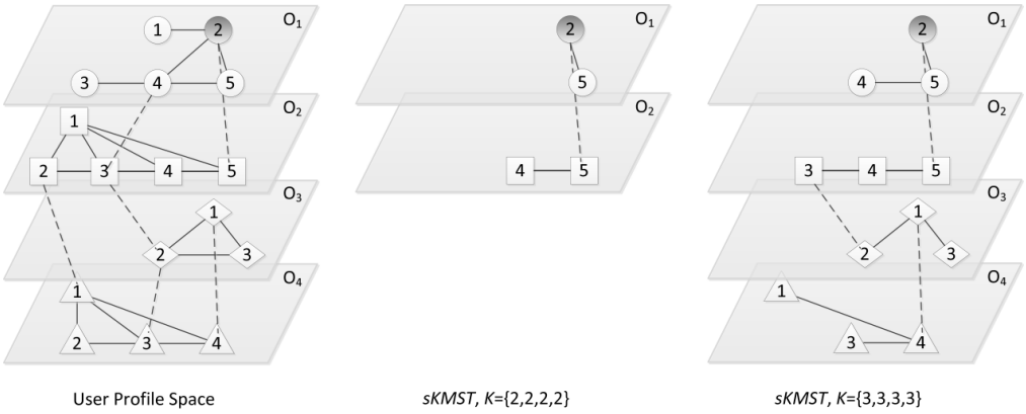


Figure 8.6 Spatial K-minimum spanning tree in the user profile space

The UPS in the figure involves four ontologies o_1, o_2, o_3, o_4 and the identified feature is s_2, where $s_2 \in o_1$,

- If $K = \{2,2,2,2\}$, the sKMST of UPS is:

  $\{s_{2,1}, s_{5,1}\} \cup \{s_{4,2}, s_{5,2}\} \cup \{spatial\_edge(s_{2,1}, s_{5,2})\}$.

- If $K = \{3,3,3,3\}$, the sKMST of UPS is:

  $\{s_{2,1}, s_{4,1}, s_{5,1}\} \cup \{s_{3,2}, s_{4,2}, s_{5,2}\} \cup \{s_{1,2}, s_{2,2}, s_{3,3}\} \cup \{s_{1,4}, s_{3,4}, s_{4,4}\} \cup$
  $\{spatial\_edge(s_{2,1}, s_{5,2})\} \cup \{spatial\_edge(s_{3,2}, s_{2,3})\} \cup \{spatial\_edge(s_{1,3}, s_{4,4})\}$.

8.3.3 Query Expansion

In a search session, a query with the selected semantic features is passed to the query expansion module. The semantic features are organised as a spatial K-minimum spanning tree, and therefore need to be further processed. $q = [t_1, t_2, \ldots, t_i, s_h]$ denotes vector of the original query, with $s_h \in o_N$. Let V_N represent the vertex set of KMST of o_N , and Q denotes expanded query set. Table 8.1 shows the pseudo code of the query expansion process.

Table 8.1 Pseudo code of query expansion

```
1:   input q, sKMST
2:   Q ← ∅
3:   for i = 1 to N
4:   convert KMST_i to V_i
5:       if i = 1
6:           λ_1 ← 1
7:       else
8:           λ_i ← φ(o_{i-1}, o_i)
9:       end if
10:      Q ← q ∪ λ_i · V_i
11:  end for
12:  return Q
```

where the return result is $Q = q \cup \lambda_1 \cdot V_1 \ldots \cup \lambda_N \cdot V_N$, where λ_N represents the ontology correlation.

The data retrieval module retrieves information objects using the expanded query set Q instead of the original query. The expanded query set Q contains several features sets, each of which corresponds to an ontology. Let $\hat{q}_N$ represent the expanded features related to ontology o_N , and $r(\hat{q}_N)$ denotes the retrieval result of $\hat{q}_N$. R is the retrieval result of Q. Table 8.2 shows the pseudo code of the data retrieval process, and the final return result is union of the respective retrieval result of $\hat{q}_N$.

Table 8.2 Pseudo code of data retrieval

1:	**input** Q
2:	$R \leftarrow \varnothing$
3:	**convert** Q **to** $\hat{q}_i$:
4:	**for** $i = 1$ **to** N
5:	**if** $i = 1$
6:	$\hat{q}_1 \leftarrow q \cup \lambda_1 \cdot V_1$
7:	**else**
8:	$\hat{q}_i \leftarrow \lambda_i \cdot V_i$
9:	**end if**
10:	**retrieve**: $R \leftarrow R \cup r(\hat{q}_i)$
11:	**end for**
12:	**return** R

where the final result is $R = r(\hat{q}_1) \ldots \cup r(\hat{q}_N)$.

Ontology correlation facilitates further processing of the result, e.g. ranking, data clustering. The correlations implicitly reflect the relevance between expanded queries and original query. In ranking based retrieval, if one expanded query has a low correlation with the original query, then the result of the expanded query could be ranked in a low position. In cluster-based retrieval, the retrieval results are grouped by the semantic topics, and the similarities between the groups are measured based on the ontology correlations.

8.4 Experiment and Evaluation

This experiment aims to evaluate and demonstrate the proposed algorithms. The data is collected from news websites in English that contain 42,824 documents (BBC: 19,353; Telegraph: 7,187; Daily Mail: 16,284). The collection includes several topics, i.e. health, celebrity, sports, financial, technology. For this experiment, 2,050 documents are selected from the celebrity topic; all documents are manually labelled based on the content. The building of the user-oriented ontology follows its specification; four ontologies are built for the purpose of this experiment: British Royal family (o_1), British place names (o_2), historic dates related to the British Royal family (o_3) and recent politicians (o_4)[1]. Figure 8.7 shows part of a user-oriented ontology applied in this chapter. Its topic is "British Royal family" and the ontology concepts are named entities related to this topic. The concepts are connected by relations, e.g. r_1: <*parent-of*>/<*child-of*>, r_2: <*spouse-of*> and r_3: <*sibling-of*>.

DBpedia provides essential information to build o_2, for example, Table 8.3 shows an entity of English place name from DBpedia, i.e. "Buckingham Palace", which includes its description, attributes (relations) and values(related entities).

[1] o_1 includes 27 concepts, the information is from www.royal.gov.uk, and www.britroyals.com; o_2 includes 32 concepts, part of the information is from DBpedia, http://dbpedia.org/page/English_Place-Name_Society; o_3 is based on www.royal.gov.uk; o_4 includes recent politicians of UK and US, e.g. prime ministers, chancellors and presidents.

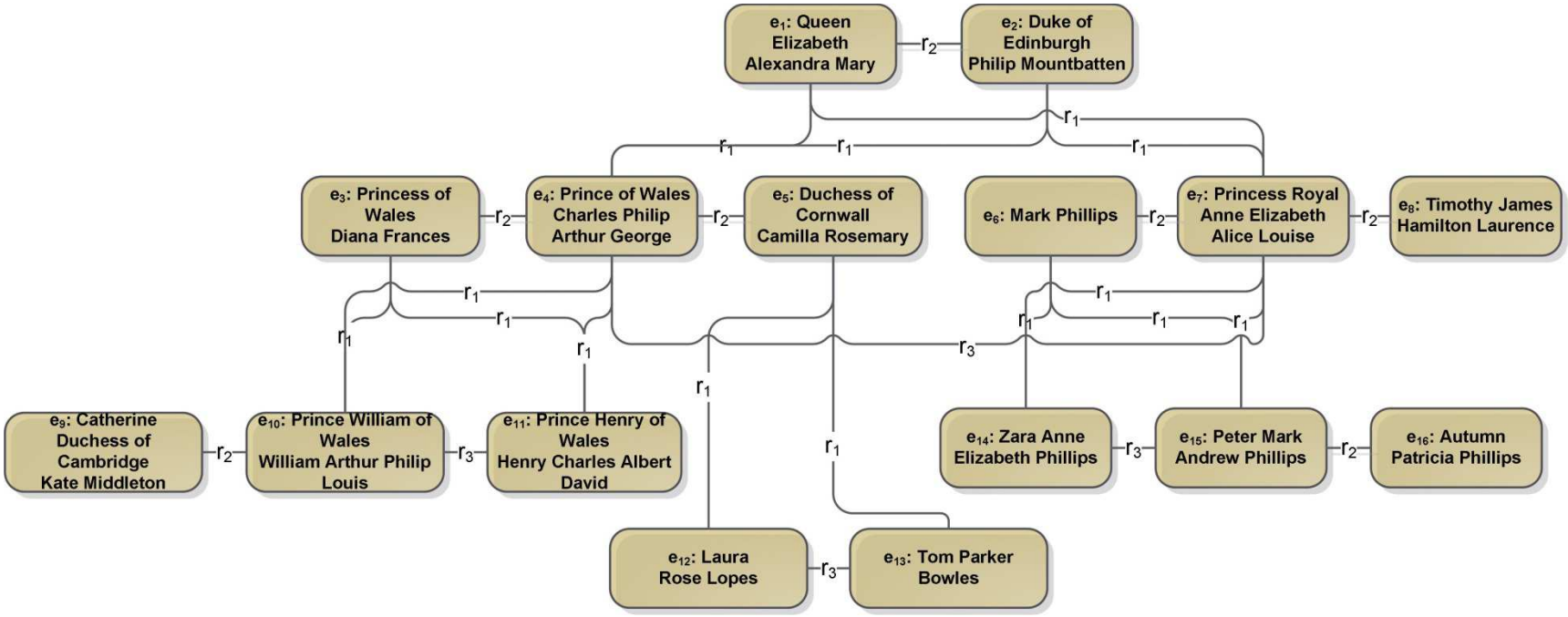


Figure 8.7 Part of a user-oriented ontology with 16 named entities

Table 8.3 Named entity "Buckingham Palace" with its description, related attributes and values in DBpedia

Buckingham Palace *is the official London residence and office of the British monarch. Located in the City of Westminster, the palace is a setting for state occasions and royal hospitality. It has been a focus for the British people at times of national rejoicing and crisis. Originally known as Buckingham House, the building which forms the core of today's palace was a large townhouse built for the Duke of Buckingham in 1705 on a site which had been in private ownership for at least 150 years.*

Attributes	**Values**
is sameAs of	• Buckingham Palace
is death place of	• Edward VII • Princess Alice of Battenberg • Princess Beatrice of the United Kingdom
is birth place of	• Edward VII • Prince Adolphus, Duke of Cambridge • Princess Sophia of the United Kingdom • Prince Arthur, Duke of Connaught and Strathearn • William IV of the United Kingdom • Prince Augustus Frederick, Duke of Sussex • Princess Helena of the United Kingdom
is Wikipage disambiguates of	Buckingham (disambiguation)
is Wikipage redirect of	• Buck House • Buckingham palace • The Queens Private Apartments. • Buckingham Palace Act 1832 • The Queens Private Apartments

The evaluation of the proposed approach uses precision, recall and f-score. The approach is compared with TF-IDF-based retrieval. For a given query, the search intention is predefined by an identified feature contained in it, which refers to one of the ontologies. The query related K-minimum spanning tree is constructed by the

identified feature and k features from the ontology. For example, for $k_1 = 3$, the query "*Philip Mountbatten's life*" has an identified feature "*Philip Mountbatten*" in o_1, and then the KMST could be "*Philip Mountbatten*, <*spouse-of*> *Queen Elizabeth II*, <*parent-of*> *Charles Philip*". Ontologies are not used in the TF-IDF-based retrieval, but the predicated search intention is appended to the query. For example, the same query "*Philip Mountbatten's life*" is converted to "*British Royal family; Philip Mountbatten's life*". The adjustment reduces the ambiguity of some short queries and improves the performance of the TF-IDF for the purposes of the evaluation. It is based the following considerations: (1) in practice, the users would use hypernyms to improve the precision by reducing query ambiguity; (2) the use of hypernyms also produces positive impact on recall (Snow et al., 2004, Ritter et al., 2009). Considering that the ontology concepts in o_3 (historic dates) are linked by a smaller number of semantic relations, this evaluation only involves the other three ontologies, i.e. o_1: British Royal family, o_2: British place names and o_4: politicians. The test query set is generated from the title of the documents, e.g. "*Duchess of Cambridge visits Liverpool charities*". For the evaluation of ontologies o_1, o_2 and o_4, each evaluation employs K-fold cross-validation, so that the dataset is split in 5 subsets of equal size. Five tests are conducted for each of the ontologies. In each test, one of the subsets (20% of the data) is used to generate the testing queries, and the remaining data (80%) is used as a validation dataset. The final test result with a particular ontology is the average of its 5 tests.

The precision-recall results measured during the experiments are shown in Table 8.4 to 8.6, and the precision-recall curves and f-score are shown in Figure 8.8 to 8.13. Clearly, the ontology-based personalised retrieval consistently outperforms TF-IDF-based retrieval. For the evaluation with o_1: British Royal family (see Table 8.4, Figure 8.8 and 8.9), the precision improvements of ontology-based personalised retrieval are +17.31% (KMST K=1), +22.73% (KMST K=2) and +24.87% (KMST K=3); the average f-score improvements are +7.46% (KMST K=1), +8.66% (KMST K=2) and +10.48% (KMST K=3). For the evaluation with o_2: British place names (see Table

8.5, Figure 8.10 and 8.11), the precision improvements are +6.64% (KMST K=1), +8.17% (KMST K=2) and +12.80% (KMST K=3); the average f-score improvements are +2.73% (KMST K=1), +3.07% (KMST K=2) and +4.89% (KMST K=3). For the evaluation with o_4: politicians (see Table 8.6, Figure 8.12 and 8.13), the precision improvements are +12.44% (KMST K=1), +13.94% (KMST K=2) and +16.47% (KMST K=3); the average f-score improvements are +6.04% (KMST K=1), +6.96% (KMST K=2) and +8.06% (KMST K=3). It can be concluded that the K-minimum spanning tree-based semantic feature selection effectively enhances the semantic representation of the query. It avoids the short query problem, and the selected features are highly correlated with the query, which is the required characteristic in personalised retrieval.

Table 8.4 Precision-Recall of TF-IDF and KMST, o_1

Recall	Precision			
	TFIDF	**KMST O1 K=1**	**KMST O1 K=2**	**KMST O1 K=3**
0	1.0000	1.0000	1.0000	1.0000
0.1	0.6364	0.7982	0.8182	0.8338
0.2	0.5482	0.6209	0.6591	0.6619
0.3	0.4221	0.5073	0.5527	0.5727
0.4	0.3521	0.4113	0.4208	0.4321
0.5	0.2727	0.3134	0.3327	0.3353
0.6	0.2273	0.2653	0.2873	0.2901
0.7	0.1848	0.2228	0.2248	0.2318
0.8	0.1705	0.1795	0.1835	0.1845
0.9	0.1475	0.1555	0.1557	0.1558
1.0	0.1205	0.1205	0.1205	0.1205

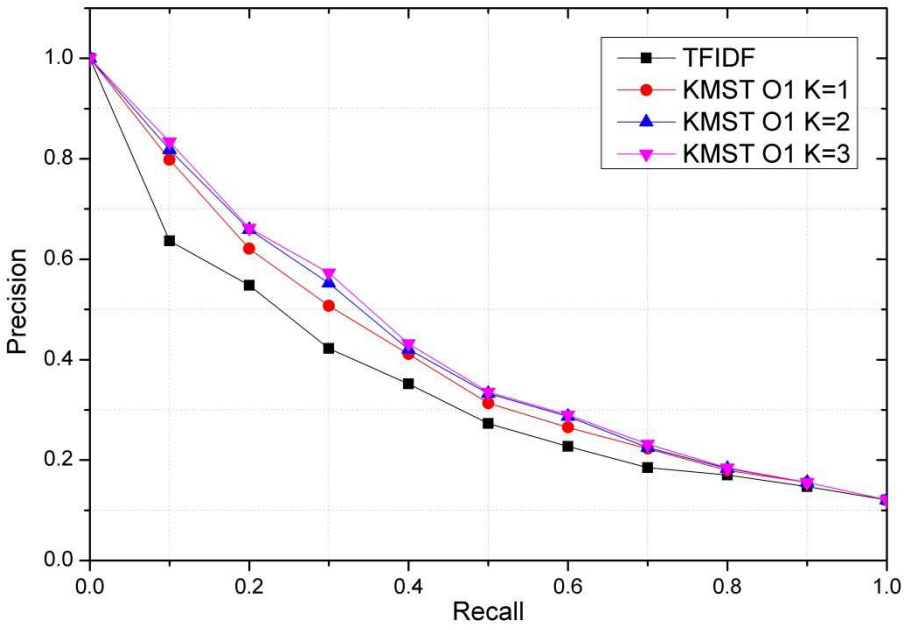


Figure 8.8 Precision-Recall of TF-IDF and KMST, o_1

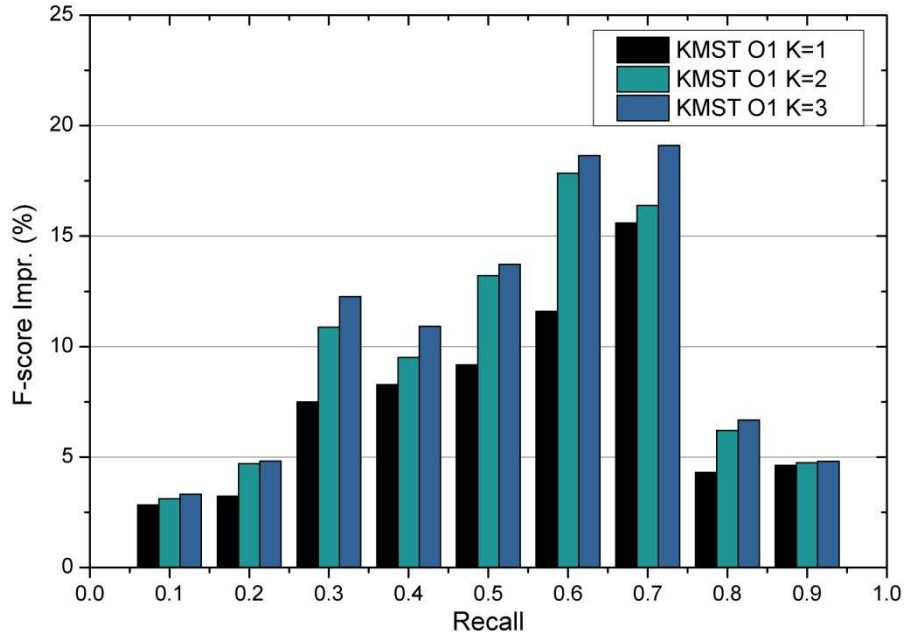


Figure 8.9 F-score improvement of KMST, o_1

Table 8.5 Precision-Recall of TF-IDF and KMST, o_2

Recall	Precision			
	TFIDF	**KMST O2 K=1**	**KMST O2 K=2**	**KMST O2 K=3**
0	1.0000	1.0000	1.0000	1.0000
0.1	0.7054	0.7776	0.7976	0.8176
0.2	0.6136	0.6521	0.6591	0.6891
0.3	0.5735	0.5957	0.6057	0.6426
0.4	0.5043	0.5399	0.5478	0.5678
0.5	0.4688	0.5098	0.5168	0.5368
0.6	0.3927	0.4114	0.4173	0.4573
0.7	0.3418	0.3578	0.3688	0.3829
0.8	0.2413	0.2600	0.2525	0.2566
0.9	0.1587	0.1615	0.1615	0.1616
1.0	0.1366	0.1366	0.1366	0.1366

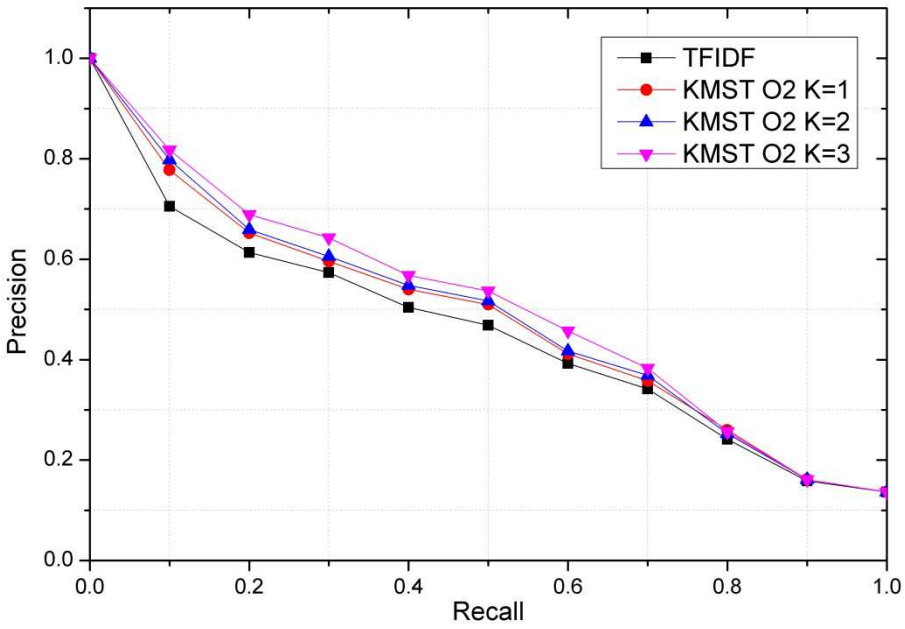


Figure 8.10 Precision-Recall of TF-IDF and KMST, o_2

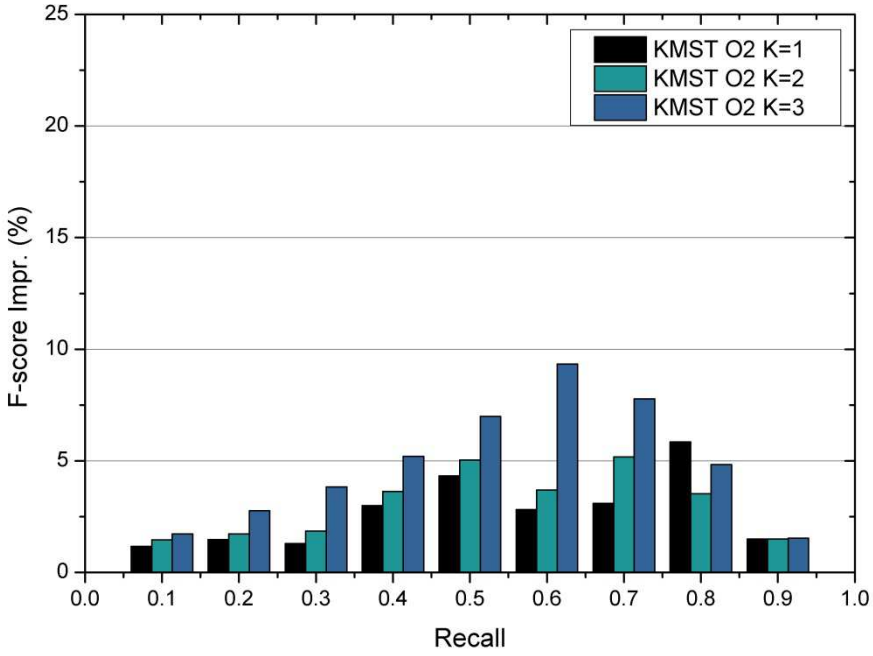
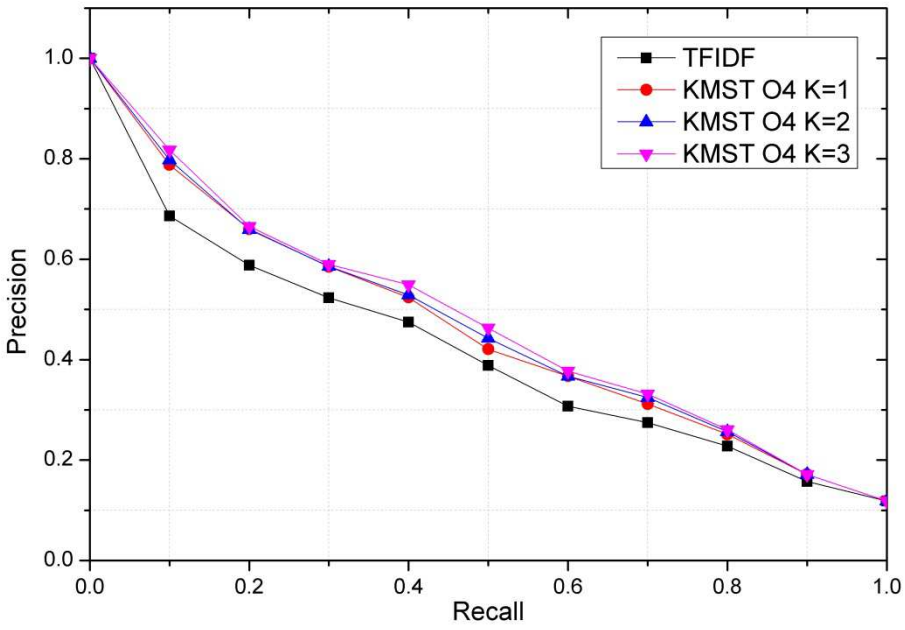


Figure 8.11 F-score improvement of KMST, o_2

Table 8.6 Precision-Recall of TF-IDF and KMST, o_4

Recall	Precision			
	TFIDF	**KMST O4 K=1**	**KMST O4 K=2**	**KMST O4 K=3**
0	1.0000	1.0000	1.0000	1.0000
0.1	0.6864	0.7876	0.7976	0.8176
0.2	0.5882	0.6601	0.6591	0.6650
0.3	0.5234	0.5851	0.5857	0.5899
0.4	0.4746	0.5240	0.5288	0.5492
0.5	0.3887	0.4207	0.4427	0.4628
0.6	0.3073	0.3673	0.3673	0.3773
0.7	0.2746	0.3118	0.3248	0.3318
0.8	0.2277	0.2515	0.2565	0.2605
0.9	0.1575	0.1715	0.1717	0.1718
1.0	0.1186	0.1186	0.1186	0.1186



9

Figure 8.12 Precision-Recall of TF-IDF and KMST, o_4

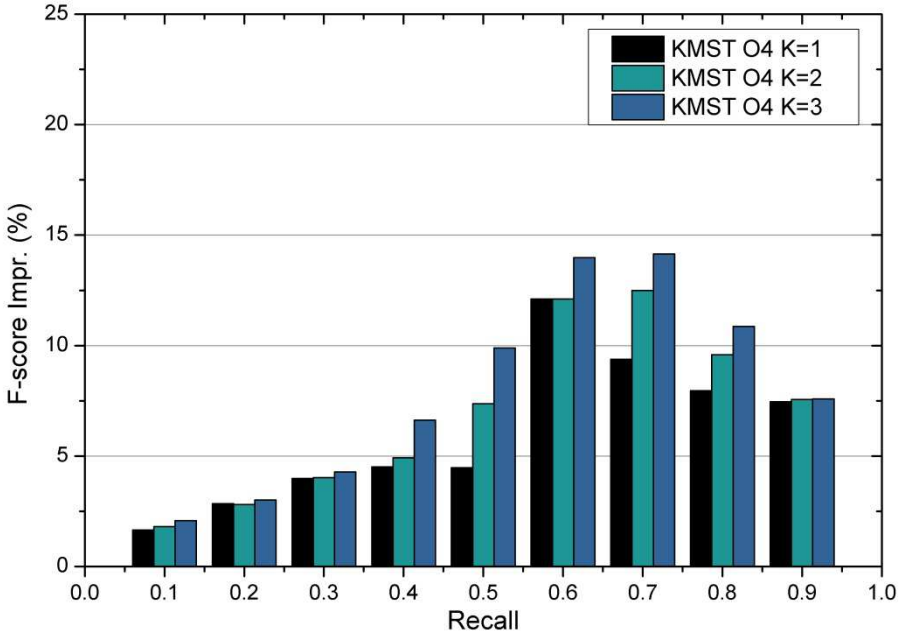


Figure 8.13 F-score improvement of KMST, o_4

To demonstrate the multi-ontology based approach (sKMST), the dataset is also split into five subsets of equal size, four of which are used as training datasets $X = \{x_1, x_2, x_3, x_4\}$. Each training dataset stands for one virtual user's feedback. The remaining subset is used as the validation dataset that represents a shared personal data collection of the virtual users. All ontologies used in the study (o_1, o_2, o_3 and o_4) are applied. With the virtual user's feedbacks, the ontology correlations can be calculated using formula 8.7 and 8.8. The user profile space (UPS) is constructed using the top 3 correlated ontologies. Table 8.7 shows the ontology correlation results of the virtual users, e.g. $UPS_1 = \{o_1, o_2, o_4\}$, as o_1, o_2 and o_2, o_4 have the higher ontology correlations than the others, i.e. $\varphi(o_1, o_2) = 0.2444$, $\varphi(o_2, o_4) = 0.1053$. Similarly, $UPS_2 = \{o_1, o_4, o_2\}$, $UPS_3 = \{o_1, o_3, o_4\}$ and $UPS_4 = \{o_1, o_3, o_2\}$.

Table 8.7 Ontology correlations generated from the training dataset

(a) ontology correlations with x_1

	o_1	o_2	o_3	o_4
o_1	1	0.2444	0.1282	0.1915
o_2	0.2444	1	0.0667	0.1053
o_3	0.1282	0.0667	1	0.1875
o_4	0.1915	0.1053	0.1875	1

(b) ontology correlations with x_2

	o_1	o_2	o_3	o_4
o_1	1	0.1778	0.1026	0.2553
o_2	0.1778	1	0.0333	0.1395
o_3	0.1026	0.0333	1	0.1313
o_4	0.2553	0.1395	0.1313	1

(c) ontology correlations with x_3

	o_1	o_2	o_3	o_4
o_1	1	0.1556	0.1795	0.1702
o_2	0.1556	1	0.0533	0.1579
o_3	0.1795	0.0533	1	0.1250
o_4	0.1702	0.1579	0.1250	1

(d) ontology correlations with x_4

	o_1	o_2	o_3	o_4
o_1	1	0.1467	0.2051	0.1064
o_2	0.1467	1	0.1167	0.0789
o_3	0.2051	0.1167	1	0.1094
o_4	0.1064	0.0789	0.1094	1

For different user profile spaces, given the same query, the selected features could be different. A query q: {Prince William} has an identified semantic feature s_h: *William*. To demonstrate the spatial K-minimum spanning tree-based approach,

UPS_1 and UPS_4 are selected. To simplify the example, the tuning parameters are set to be identical in this demonstration, i.e. $k_1 = k_2 = k_3 = k_4 = 2$. As mentioned before, the correlated semantic features from different ontologies are used to establish the spatial edges of sKMST. Table 8.8 shows the construction of sKMST in UPS_1 with s_h : *William*. The selected feature from o_1 is $s_{2,o1}$: *Charles*, which has the shortest edge distance and a high concept entropy with s_h. Furthermore, according to the training data, s_{2,o_1} : *Charles* has the highest feature correlation with $s_{1,o2}$: *Windsor Castle*, thus one spatial edge is established between o_1 and o_2. Furthermore, $s_{2,o2}$: *Buckingham Palace* and $s_{1,o4}$: *David Cameron* construct another spatial edge between o_2 and o_4. The vertex set of sKMST in UPS_1 is {*William, Charles; Windsor Castle, Buckingham Palace; David Cameron, Nick Clegg*}.

Table 8.8 sKMST in UPS_1 with $k_1 = k_2 = k_4 = 2$

q	s_h : William	
		$edge\_dist(s_h, s_{2,o1}) = 0.4893$ ↓
o_1	s_h : William	$s_{2,o1}$: Charles •
	$\vartheta(s_{2,o1}, s_{1,o2}) = 0.0032$ •; $edge\_dist(s_{1,o2}, s_{2,o2}) = 0.4893$ ↓	
o_2	$s_{1,o2}$: Windsor Castle •	$s_{2,o2}$: Buckingham Palace ◊
	$\vartheta(s_{2,o2}, s_{1,o4}) = 0.0015$ ◊; $edge\_dist(s_{1,o4}, s_{2,o4}) = 0.4893$ ↓	
o_4	$s_{1,o4}$: David Cameron ◊	$s_{2,o4}$: Nick Clegg

Using the same query, Table 8.9 shows the construction of sKMST in UPS_4. The sKMST in UPS_4 has different structure with the previous one in UPS_1, as the correlated ontologies are different in the two user profile spaces. Between o_1 and o_3, s_h : *William* and $s_{1,o3}$: *29 April 2011* establish the spatial edge. Meanwhile, $s_{1,o3}$: *29 April 2011* and $s_{1,o2}$: *Westminster Abbey* establish another spatial edge between o_3 and o_2. The vertex set of sKMST in UPS_4 is {*William, Charles, 29 April 2011, 2 June 2012, Westminster Abbey, Buckingham Palace*}. Obviously, given the same query

but different UPSs, the spatial K-minimum spanning trees have dissimilar structures, as the correlated ontologies are different in the user profile spaces.

Table 8.9 sKMST in UPS_4 with $k_1 = k_3 = k_2 = 2$

q	s_h : William	
		$edge\_dist(s_h, s_{2,o1}) = 0.4893$ ↓
o_1	s_h : William ●	$s_{2,o1}$: Charles
	$\vartheta(s_h, s_{1,o3}) = 0.0025$ ●;	$\text{edge}(s_{1,o3}, s_{2,o3}) = 0.4893$ ↓
o_3	$s_{1,o3}$: 29 April 2011 ● ◊	$s_{2,o3}$: 2 June 2012
	$\vartheta(s_{1,o3}, s_{1,o2}) = 0.0011$ ◊;	$\text{edge}(s_{1,o3}, s_{2,o3}) = 0.4893$ ↓
o_2	$s_{1,o2}$: Westminster Abbey ◊	$s_{2,o2}$: Buckingham Palace

The retrieval process shown in Table 8.8 and Table 8.9 is illustrated in Figure 8.14 and Figure 8.15, respectively. Giving the same query with different user profile spaces, the semantic feature selection algorithm selects features according to the UPS structure and ontology correlations. Moreover, it also considers feature correlation, concept similarity and concept entropy, which insures that the selected features have appropriate correlation to the original query. In addition, the customised retrieval result has story-like format, which helps people recall and re-discover events from their life.

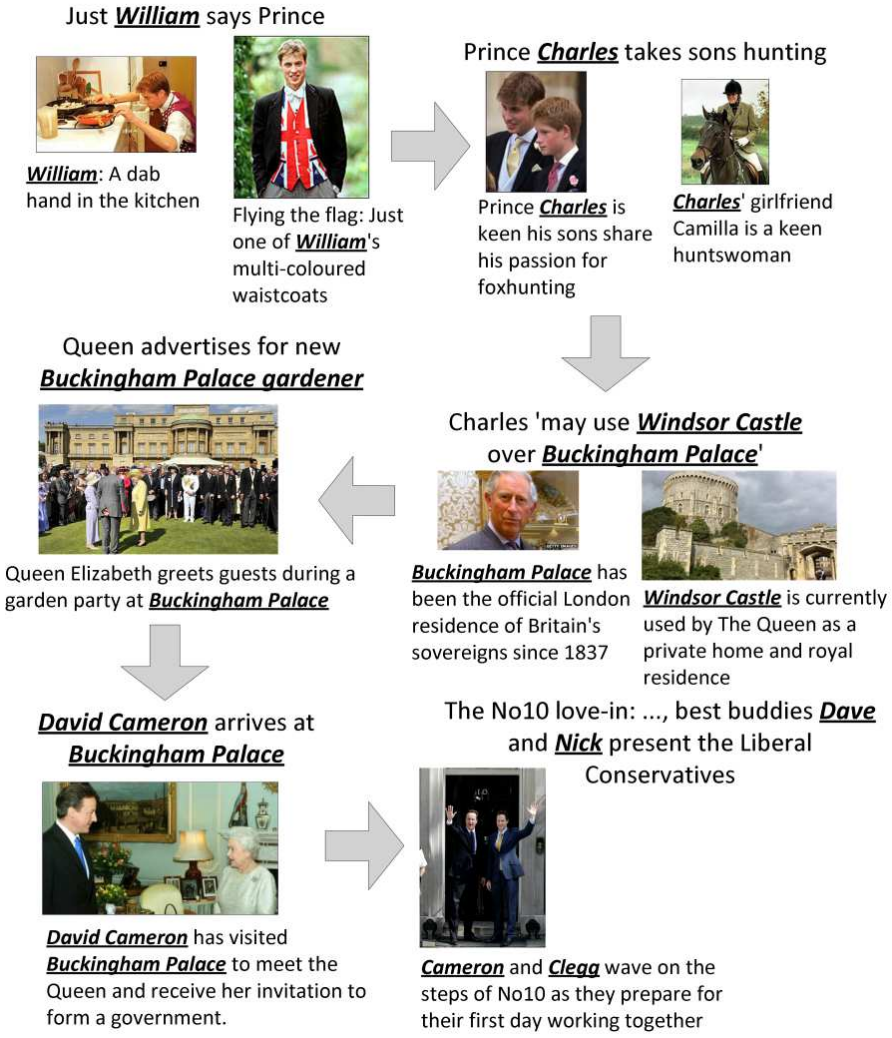


Figure 8.14 Retrieval results based on UPS_1 with the query

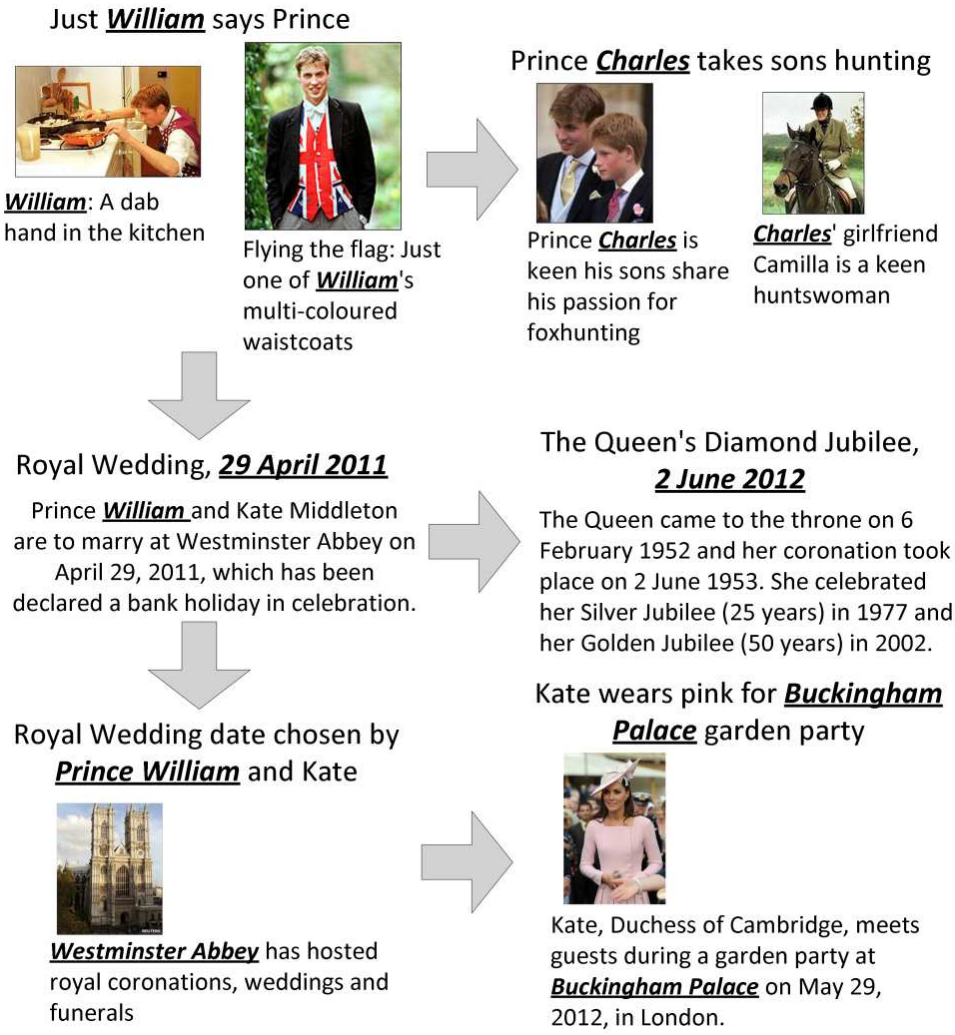


Figure 8.15 Retrieval results based on UPS_4 with the query

8.5 Summary

This chapter proposes a mechanism that integrates ontology-based personalised retrieval with reminiscence support. The aim is to assist people to recall, browse and re-discover events from their lives by considering their profiles and background knowledge and providing them with customised retrieval. To identify people's search intention and provide data from different knowledge domains, this chapter introduces a multi-ontology query expansion model which is based on ontologies, ontology graphs and semantic feature selection algorithms. Furthermore, the chapter defines a user

profile space and describes its construction method. The model has dynamic structure based on relevance feedback. The method developed employs semantic feature selection to achieve concept-based retrieval. Based on the search intention, the semantic feature selection generates knowledge spanning trees of the ontology graph or user profile space. For the query expansion, these trees provide various semantic features and relations that enhance the semantic representation of the original query, and further facilitate customised retrieval. The experiments show the positive effect of combining semantic feature selection with ontology/user profile space on identifying search intention and query expansion.

The evaluation shows that the ontology-based personalised retrieval outperforms term-based retrieval with precision, recall and f-score in all tests. It also shows that the proposed tree-based semantic feature selection has proven ability to predict the search intention of the query, and appropriately expand the query according to the predicted intention. In addition, the semantic relations between the information objects are considered, thus enabling the retrieval results to be displayed in a story-like format.

Chapter 9

System Evaluation

Crowdsourcing as a distributed problem-solving model is widely used in evaluation tasks recently (Alonso et al., 2008, Kittur et al., 2008). With crowdsourcing, a task is split to sub-tasks by its requester, and theses sub-tasks are distributed to experienced workers, who complete them according to the requester's requirements. This evaluation therefore applies crowdsourcing to test the performance of Sem-LSB with real users. This chapter shows the system evaluation of Sem-LSB. Section 9.1 introduces the evaluation protocol. Section 9.2 lists the evaluation results and analyses the results accordingly. Section 9.3 summarises the chapter.

9.1 Evaluation Protocol

Sem-LSB is designed for recording, storing, managing and reusing personal information. If a user's requirement is determined, Sem-LSB then generates a collection of information based on his/her requirement and displays the information to the user. To enable the user to explore specific events from various perspectives, the generated content of Sem-LSB needs to meet the following requirements: (i) dynamic structure, (ii) informative, (iii) readable and understandable.

The objective of this evaluation is to measure users' satisfaction on the generated content of Sem-LSB. The evaluation contains three individual experiments, and each of which contains three scenarios. A single scenario contains one presentation of the generated content of its related experiment, so that each experiment contains three different presentations, (i) unstructured, (ii) LSB-based, and (iii) semantic/knowledge-based. Each presentation contains 12 to 20 information objects, which are images with annotations under its related scenario. The unstructured presentation is shown in Figure 9.1. It is similar to a keyword-based search result, which is related to the user's

requirement, but does not have a clear structure. The LSB-based presentation is shown in Figure 9.2. Its structure is derived from the typical life story book (LSB). This presentation is a type of reconstruction of the unstructured presentation, which organises the content using its timestamps/theme words. The semantic/knowledge-based presentation is shown in Figure 9.3. It is generated using the algorithms proposed in this thesis. This presentation organises the content based on "associations", which are established in accordance with the semantic relations between the information objects.

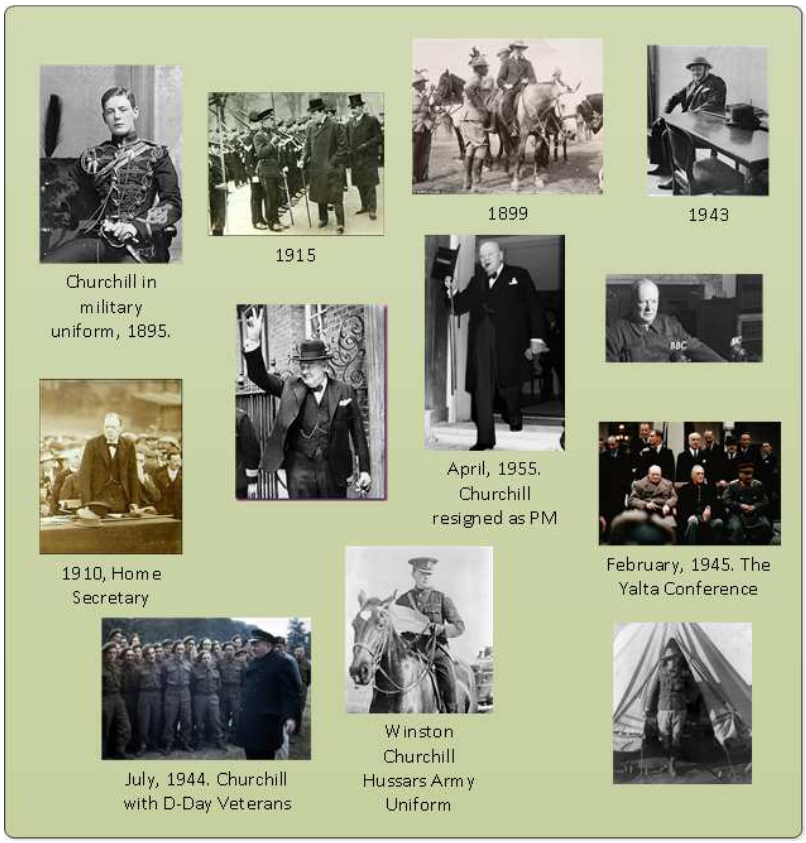


Figure 9.1 Experiment II: Unstructured presentation (Scenario I)

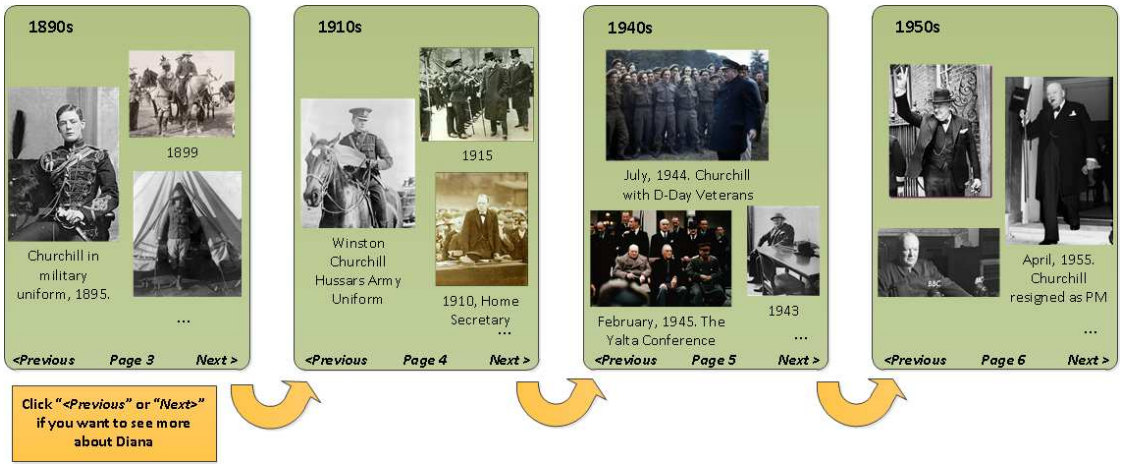


Figure 9.2 Experiment II: LSB-based presentation (Scenario II)

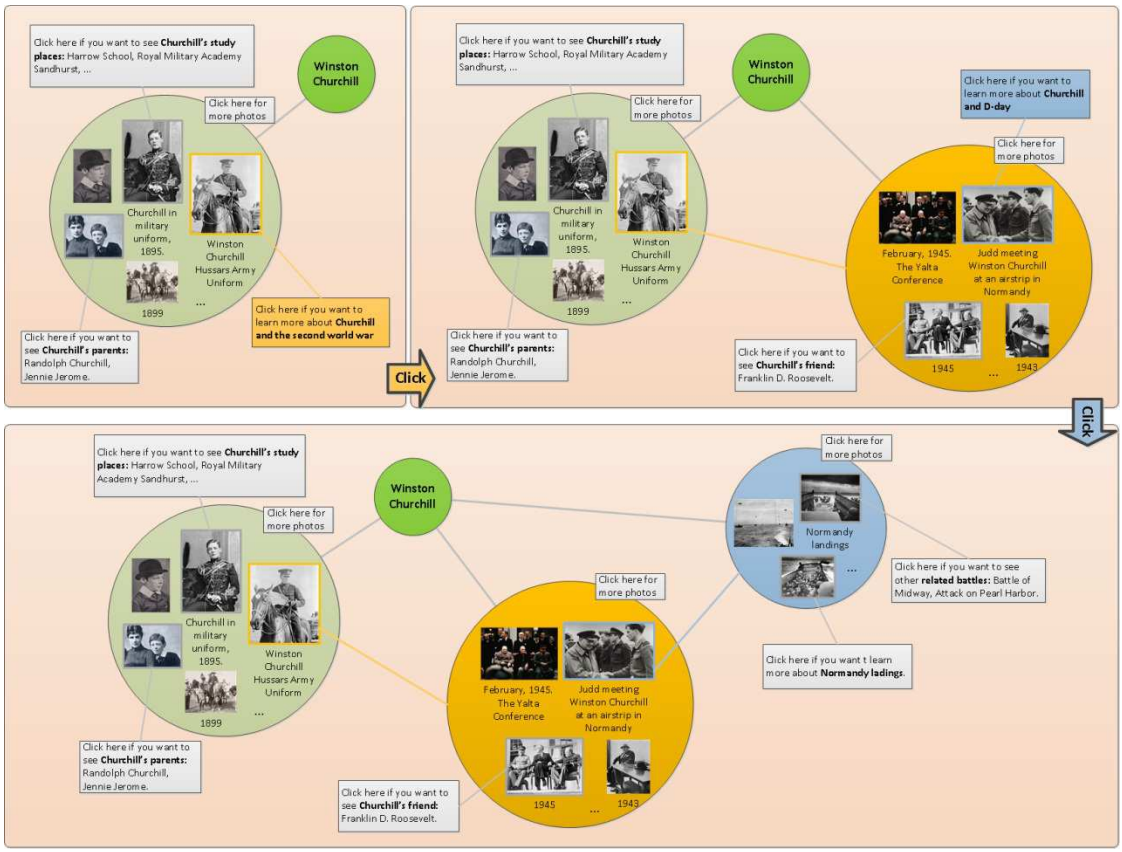


Figure 9.3 Experiment II: Semantic/knowledge-based presentation (Scenario III)

The question setting is based on the above mentioned requirements of generated content of Sem-LSB. In the questions, Q1 and Q2 aim to collect user feedback related to the first requirement, while Q3 aims to collect user feedback related to the second

requirement and Q4 to the last requirement. The list of the questions is shown in Table 9.1.

Table 9.1 List of questions in each scenario

	Question
Q1	To what extent do you agree this way of presenting information enables you to explore specific events based on their chronological order?
Q2	To what extent do you agree it enables you to explore specific events based on people, locations, dates or other concepts?
Q3	How informative do you find this way of presenting information?
Q4	With the information collection provided to you, do you think it is easy to read/understand?

For these questions, 4-points scales are used as their response options (see Table 9.2). The text-based options are converted to numerical scores to support quantitative analysis. These scores describe the positive/negative strength of the options, i.e. a higher score indicates the corresponding option has more positive strength.

Table 9.2 List of options with scores

Score	Options-1	Options-2	Options-3
3	Strongly agree	Very informative	Very easy
2	Agree	Informative	Easy
1	Partly agree	Fairly informative	Fairly easy
-1	Disagree	Not informative	Not easy

9.2 Evaluation Results

For this evaluation, user feedback was collected from 157 participants, who had completed 1,836 assignments regarding 9 different scenarios. The selection of the participants is based on 2 criteria: (i) the number of their previously completed assignments, and (ii) the acceptance rate of their previously completed assignments. Firstly, the minimum number of previously completed assignments for the participants is set as 500, which means any participant selected needs to complete at least 500 similar assignments before. Secondly, the acceptance rate of the previously completed assignments is sets as 95%, which means at least 95% of the previously completed

assignments had been approved and accepted by their assignment requestors. Base on such selection criteria, the experienced participants with serious attitude are selected to attend the evaluation ensuring that the collected feedback is of high quality.

Before commencing the evaluation, the participants are required to read the experiment induction and scenario descriptions. After that, the participants are required to do analysing, summarising and reminiscing based on the presentations, and then they need to answer the questions of each scenario. In the experiments, Scenario I, II and III are corresponding to the unstructured presentation, LSB-based presentation and semantic/knowledge-based presentation respectively. More details of the experiments are included in Appendix D.

The results of Experiment I, II and III are shown in Table 9.3, 9.4 and 9.5. The percentages in the tables represent the proportion of the relevant option being selected. A higher percentage means the option is selected by more participants.

Table 9.3 Evaluation results of Experiment I

	Scenario I	**Scenario II**	**Scenario III**
Q1			
3	9.62%	51.92%	34.62%
2	59.62%	38.46%	44.23%
1	21.15%	9.62%	21.15%
-1	9.62%	0.00%	0.00%
Q2			
3	17.31%	26.92%	55.77%
2	51.92%	44.23%	32.69%
1	28.85%	23.08%	11.54%
-1	1.92%	5.77%	0.00%
Q3			
3	21.15%	36.54%	51.92%
2	48.08%	48.08%	36.54%
1	28.85%	15.38%	9.62%
-1	1.92%	0.00%	1.92%
Q4			
3	38.46%	59.62%	40.38%
2	46.15%	30.77%	38.46%
1	15.38%	9.62%	15.38%
-1	0.00%	0.00%	5.77%

Table 9.4 Evaluation results of Experiment II

	Scenario I	Scenario II	Scenario III
Q1			
3	13.46%	38.46%	34.62%
2	44.23%	44.23%	44.23%
1	23.08%	17.31%	19.23%
-1	19.23%	0.00%	1.92%
Q2			
3	15.38%	23.08%	44.23%
2	46.15%	53.85%	40.38%
1	23.08%	19.23%	13.46%
-1	15.38%	3.85%	1.92%
Q3			
3	28.85%	32.69%	48.08%
2	21.15%	42.31%	34.62%
1	34.62%	25.00%	13.46%
-1	15.38%	0.00%	3.85%
Q4			
3	25.00%	42.31%	36.54%
2	38.46%	38.46%	34.62%
1	23.08%	17.31%	19.23%
-1	13.46%	1.92%	9.62%

Table 9.5 Evaluation results of Experiment III

	Scenario I	Scenario II	Scenario III
Q1			
3	9.43%	33.96%	33.96%
2	37.74%	62.26%	49.06%
1	35.85%	3.77%	15.09%
-1	16.98%	0.00%	1.89%
Q2			
3	7.55%	22.64%	45.28%
2	47.17%	52.83%	41.51%
1	41.51%	20.75%	11.32%
-1	3.77%	3.77%	1.89%
Q3			
3	16.98%	30.19%	45.28%
2	35.85%	54.72%	39.62%
1	43.40%	15.09%	15.09%
-1	3.77%	0.00%	0.00%
Q4			
3	18.87%	39.62%	30.19%
2	50.94%	52.83%	43.40%
1	26.42%	7.55%	20.75%
-1	3.77%	0.00%	5.66%

To facilitate data analysis, these percentage-based results are then converted to relevant average scores, which are shown in Table 9.6, 9.7 and 9.8.

Table 9.6 Average score of the evaluation results of Experiment I

	Scenario I	Scenario II	Scenario III
Q1	1.596	2.423	2.135
Q2	1.827	1.865	2.442
Q3	1.865	2.212	2.365
Q4	2.231	2.500	2.077

Table 9.7 Average score of the evaluation results of Experiment II

	Scenario I	Scenario II	Scenario III
Q1	1.327	2.212	2.096
Q2	1.462	1.923	2.250
Q3	1.481	2.077	2.231
Q4	1.615	2.192	1.885

Table 9.8 Average score of the evaluation results of Experiment III

	Scenario I	Scenario II	Scenario III
Q1	1.226	2.302	2.132
Q2	1.547	1.906	2.283
Q3	1.623	2.151	2.302
Q4	1.811	2.321	1.925

According to the average scores, the LSB-based presentation is shown as the strongest one that can support the participants exploring specific information based on chronological orders. The average score of the LSB-based presentation is higher than the unstructured presentation by 51.82% in Experiment I, 66.69% in Experiment II, and 87.76% in Experiment III. It is higher than the semantic/knowledge-based presentation by 13.49% in Experiment I, 5.53% in Experiment II, and 7.93% Experiment III.

The semantic/knowledge-based presentation is shown as the strongest one that can support the participants exploring specific information based on associations, e.g. the semantic relations of people, locations, dates, etc. The average score of the semantic/knowledge-based presentation is higher than the unstructured presentation by 33.66% in Experiment I, 53.90% in Experiment II, and 47.58% in Experiment III.

It is higher than LSB-based presentation by 30.94% in Experiment I, 17.00% in Experiment II, and 19.78% in Experiment III.

Meanwhile, the semantic/knowledge-based presentation is shown as the most informative one. The average score of the semantic/knowledge-based presentation is higher than the unstructured presentation by 26.81% in Experiment I, 50.64% in Experiment II, and 41.84% in Experiment III. It is higher than LSB-based presentation by 6.92% in Experiment I, 7.41% in Experiment II, and 7.02% in Experiment III.

The LSB-based presentation is shown as the most readable and understandable one. The average score of the LSB-based presentation is higher than the unstructured presentation by 12.06% in Experiment I, 35.73% in Experiment II, and 28.16% in Experiment III. It is higher than the semantic/knowledge-based presentation by 20.37% in Experiment I, 16.29% in Experiment II, and 20.57% in Experiment III.

9.3 Summary

Averaging the experiments on each scenario for each question, Figure 9.4 shows the normalised average scores generated from Table 9.6, 9.7 and 9.8. The scores reflect the overall satisfaction on each type of presentations, and indicate advantage and disadvantage of each.

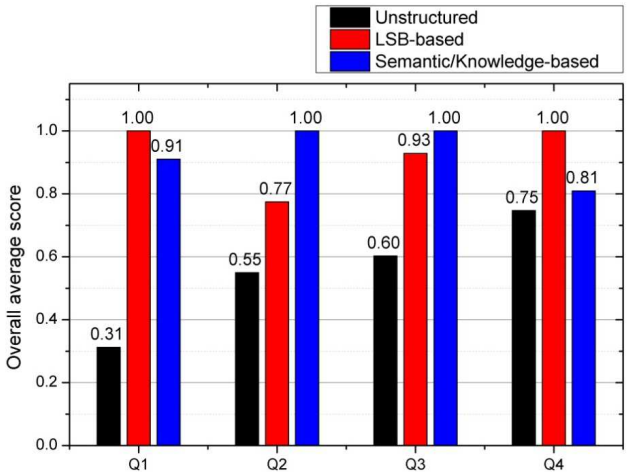


Figure 9.4 Overall average score of the presentations

For Q1 and Q4, the LSB-based presentation has the best performance. There are two reasons that can explain the results, (i) the organisation of information objects in this presentation is based on timestamps or theme words. The information objects are gathered if they are close to each other in terms of occurrence time, or have similar themes words. With this presentation, the participants therefore can easily explore the information objects by time sequence; (ii) this presentation has a page-by-page format, so that its structure is close to a paper-based book. This format is intuitive and clear for most participants, so that they can read and understand this presentation without any difficulty.

The unstructured presentation has poor performance on Q1 and Q4. The main reason is the lack of organisation, which reduces its readability and increases the difficulty to understand the content for the participants. These drawbacks of this presentation cause the participants spending more time on analysing and exploring.

The semantic/knowledge-based presentation has good performance on Q1, and average performance on Q4. For Q1, the performance of this presentation is slightly lower than the LSB-based presentation. The considered semantic relations between information objects in this presentation include date-related relationships, which are similar to the timestamps in the LSB-based presentation. Therefore, the participants can also explore specific information based on the chronological order in this presentation. However, this presentation is more difficult to read and understand than the LSB-based presentation, which is indicated by Q4. Unlike the page-by-page format, the structure of semantic/knowledge-based presentation is similar to a graph, which contains information objects and the associations between them. To explore more information, the participants can "click" on some information objects, and then more similar information objects are displayed. As such interaction is required, this presentation becomes less straightforward than the LSB-based one, and thus some participants may feel difficult to understand.

For Q2 and Q3, the semantic/knowledge-based presentation has the best performance. There are two reasons that can explain the results, (i) the organisation of information objects in this presentation is based on their content and semantic

relations, instead of only based on timestamps or theme words. The presentation therefore has a more complicated but well-organised structure that enables the participants to explore specific information from different perspectives; (ii) the structure of the presentation is flexible, and the interaction involved enables the participants to explore more associated information objects on their demands, e.g. a participant may choose to click an image of young Diana, then the associated information about her birth place, parents or study places are displayed. Each information object in this presentation is linked to more associated information objects, and hence this presentation is considered to be more informative than the others.

The unstructured presentation still has poor performance on Q2 and Q3. The reasons are similar to the previous, i.e. lack of organisation, poor readability and difficult to understand.

The LSB-based presentation has average performance on Q2 and Q3. As mentioned above, this presentation has relatively simpler structure that only considers timestamps and theme words. According to the result of Q2, most participants found it difficult to explore specific information from different perspectives. Furthermore, the structure of this presentation is fixed, which prevents the participants from viewing or exploring more associated information, thus most of them consider this presentation to be less informative than the semantic/knowledge-based presentation.

In summary, the semantic/knowledge-based presentation has better performance than the others. Firstly, most participants consider it to be able to help them explore specific information based on chronological order and associations. Secondly, they consider this presentation to be more informative than the others. Moreover, the evaluation results also indicate certain disadvantage of the semantic/knowledge-based presentation, e.g. some participants are not satisfied with its interaction process, and they consider it to be less readable and understandable than the LSB-based one.

Chapter 10 Conclusions and Further Work

This chapter concludes the thesis. Section 10.1 lists the main contributions of this research. Section 10.2 provides the conclusions. The directions of further work are discussed in Section 10.3.

10.1 Contributions

The main contributions made by this research are as follows:

- A conceptual model of a semantic reminiscence support system is proposed, which integrates essential features of personal information management and reminiscence therapy. It employs advanced technologies, such as natural language processing, knowledge modelling, topic identification, data clustering and personalised retrieval, to provide improved personal information management of stored memories.
- An interactive knowledge model for modelling user's background, named user-oriented ontology, is defined. The main features of the ontology include homogeneous semantic topics, simplified and extendable structure and effective interactivity for improved content management;
- A semantic feature matching algorithm is proposed, which includes concept similarity measure, semantic feature extraction and semantic feature selection. The algorithm enhances the semantic representation of the information objects and bridges the semantic knowledge gap in information retrieval using the user-oriented ontology;
- A topic identification algorithm is proposed. It uses a user-oriented ontology, dimensionality reduction, semantic feature matching and k-means clustering to achieve improved topic identification and clustering of information objects;

- A multiple ontology model, named a user profile space, is defined. It consists of multiple user-oriented ontologies, and therefore has a more comprehensive knowledge coverage than a single user-oriented ontology. The user profile space is adjustable based on relevance feedback from the user;
- An ontology-based personalised retrieval mechanism is proposed. It includes language models, knowledge spanning tree generation, tree-based query expansion, data clustering and implicit/explicit relevance feedback collection. The mechanism can be used to simplify the knowledge structure of the ontologies, identify the search intention and adjust the semantic representation of the queries. More importantly, it could provide retrieval results to people which are customised according to their background.

10.2 Conclusions

In this thesis, the research objectives have been achieved, together with the examination of the hypotheses. The developed semantic reminiscence support system integrates the essential features of the personal information management systems and reminiscence therapy. By employing advanced semantic and information retrieval technologies, it performs personal information analysis, management, knowledge modelling and personalised retrieval of stored memories. The evaluations conducted, which are described and discussed in the thesis, show that the proposed approaches have clear benefits.

In terms of management of personal memories, this research uses hierarchical relations between the personal information objects and people's background knowledge. The role of the user-oriented ontology in the system is similar to the semantic memory in human's memory system. The ontology contains categorised knowledge that represents the user's recalling, understanding and reflecting on his/her life experience. To simplify the building process of the user-oriented ontology, and improve its computational feasibility, the ontology utilises a simple and flexible

structure with homogeneous semantic topics. Different from the generic ontologies, this ontology has a simple structure and does not need to be maintained by knowledge experts. Its flexible structure means that it can be easily extended according to various information needs. These further reduce the computational complexity of using the ontology. Overall, as the knowledge base of the semantic system for reminiscence support, the user-oriented ontology is straightforward, extendable and covers users' background knowledge, which provides a reliable way to address the semantic gap.

In this thesis, a semantic feature matching algorithm is produced that automatically analyses the observed semantic features in the information objects and then selects the latent semantic features from the user-oriented ontology to enhance their semantic representation. The experiment shows that the similarity measure based on the enhanced representation of information objects can improve the retrieval performance on semantic knowledge basis.

An essential task in personal information management is understanding the content of the information objects. This includes identifying the semantic relations between the information objects, and then categorising them based on their similarities. The proposed algorithm uses the user-oriented ontology to automatically detect the semantic relations within the stored personal information. It combines semantic feature selection with dimensionality reduction and k-means clustering, to achieve topic identification based on semantic similarity. The experiments conducted explore the effect of semantic feature selection as a result of establishing semantic relations, with the help of the ontology, within the information content.

To provide a personalisation content generation mechanism to different users, an ontology-based personalised retrieval mechanism is proposed. The mechanism developed employs semantic feature selection to achieve concept-based retrieval. Based on user's search intention, the semantic feature selection algorithm generates knowledge spanning trees from the ontology graph or user profile space. Due to the query expansion, the knowledge spanning tree provides various semantic features and relations that can be applied to enhance the semantic representation of the original

query, which further facilitates customised retrieval. The experiments show the positive effects of combining semantic feature selection with ontology/user profile space on identifying the search intention and query expansion.

10.3 Further Work

The developed prototype is a complex system involving various technologies. It is an open system with a configurable and extendable structure. This research develops the essential features of the prototype, e.g. semantic content management, interactive knowledge model, personalised retrieval. However, there are several aspects worth to be studied further.

Intelligent interface: to improve the system usability, an intelligent interface can be proposed. For example, the generated content of people's life experience could be displayed in a story-like format with dynamic structure. The story-like format needs to reflect the semantic relations among the retrieved information objects, and it should be adjustable to match the different requirements of the reminiscing tasks.

Exchange of personal information: each single user of Sem-LSB has his/her own personal information repository and user profile spaces, which is isolated with other users'. Sharing or exchanging the personal information between them is sometimes necessary, e.g. a youth inherits his grandfather's repository, or a patient shares some specific information to his/her caregivers, etc. To find a reliable, secure and controllable way of sharing/exchanging personal information is an interesting research direction, which can enhance the functionality of the data management mechanism of Sem-LSB.

Collaborative building mechanism of user-oriented ontology: to facilitate knowledge sharing within a group of users, a collaborative ontology building mechanism can be developed. The users may have similar background knowledge in a specific domain, e.g. researchers from the same research community, members of a family, thus their collaboration could enable the building of comprehensive and high quality ontologies.

Potential applications: the system could be potentially applied in the medical domain, e.g. assisting the care of patients with cognitive impairment or elderly people, automatic reminiscence therapy, etc.

References

Abiteboul, S., R. Agrawal, P. Bernstein, M. Carey, S. Ceri, B. Croft, D. DeWitt, M. Franklin, H. G. Molina, D. Gawlick, J. Gray, L. Haas, A. Halevy, J. Hellerstein, Y. Ioannidis, M. Kersten, M. Pazzani, M. Lesk, D. Maier, J. Naughton, H. Schek, T. Sellis, A. Silberschatz, M. Stonebraker, R. Snodgrass, J. Ullman, G. Weikum, J. Widom and S. Zdonik (2005) The Lowell Database Research Self-Assessment. *Commun. ACM,* 48**,** 111-118.

Alonso, O., D. E. Rose and B. Stewart (2008) Crowdsourcing for Relevance Evaluation. *ACM SigIR Forum.* ACM.

Bashir, F., S. Khanvilkar, A. Khokhar and D. Schonfeld (2005) Multimedia Systems: Content-Based Indexing and Retrieval. *The Electrical Engineering Handbook.* Burlington, Academic Press.

Bayer, A. and J. Reban (Eds.) (2004) *Alzheimer's Disease and Related Conditions*, Medea Press–M. Bransovsky, Rudolfov, CZ.

Beaulieu, M. (1997) Experiments on Interfaces to Support Query Expansion. *Journal of Documentation,* 53**,** 8-19.

Berberich, K., A. C. Knig, D. Lymberopoulos and P. Zhao (2011) Improving Local Search Ranking through External Logs. *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval.* Beijing, China, ACM.

Berger, A. L., V. J. D. Pietra and S. A. D. Pietra (1996) A Maximum Entropy Approach to Natural Language Processing. *Computational Linguistics,* 22**,** 39-71.

Berners-Lee, T., J. Hendler and O. Lassila (2001) The Semantic Web. *Scientific American,* 284**,** 28-37.

Berntsen, D. and D. C. Rubin (2004) Cultural Life Scripts Structure Recall from Autobiographical Memory. *Memory and Cognition,* 32**,** 427-442.

Bhatia, S., D. Majumdar and P. Mitra (2011) Query Suggestions in the Absence of Query Logs. *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval.* Beijing, China, ACM.

Bhogal, J., A. Macfarlane and P. Smith (2007) A Review of Ontology Based Query Expansion. *Information Processing and Management,* 43**,** 866-886.

Blei, D. M. and J. D. Lafferty (2007) A Correlated Topic Model of Science. *The Annals of Applied Statistics,* 1**,** 17-35.

Blei, D. M., A. Y. Ng and M. I. Jordan (2003) Latent Dirichlet Allocation. *The Journal of Machine Learning Research,* 3**,** 993-1022.

Bluck, S. and L. J. Levine (1998) Reminiscence as Autobiographical Memory: A Catalyst for Reminiscence Theory Development. *Ageing and Society,* 18**,** 185-208.

Blunschi, L., J. Peter Dittrich, O. R. Girard, S. Kirakos, K. Marcos and A. V. Salles (2007) A Dataspace Odyssey: The Imemex Personal Dataspace Management System. *In CIDR.*

Bohn, A. and D. Berntsen (2011) The Reminiscence Bump Reconsidered Children's Prospective Life Stories Show a Bump in Young Adulthood. *Psychological Science,* 22**,** 197-202.

Bollegala, D., Y. Matsuo and M. Ishizuka (2009) A Relational Model of Semantic Similarity between Words Using Automatically Extracted Lexical Pattern Clusters from the Web. *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing.* Singapore, Association for Computational Linguistics.

Bryant, F. B., C. M. Smart and S. P. King (2005) Using the Past to Enhance the Present: Boosting Happiness through Positive Reminiscence. *Journal of Happiness Studies,* 6**,** 227-260.

Buitelaar, P., P. Cimiano, A. Frank, M. Hartung and S. Racioppa (2008) Ontology-Based Information Extraction and Integration from Heterogeneous Data Sources. *International Journal of Human-Computer Studies,* 66**,** 759-788.

Bush, V. (1945) As We May Think. *The Atlantic Monthly,* 176**,** 101-108.

Butler, R. (1980) The Life Review: An Unrecognized Bonanza. *International Journal of Aging and Human Development,* 12**,** 35.

Butler, R. N. (1963) The Life Review: An Interpretation of Reminiscence in the Aged. *Psychiatry,* 26**,** 65-76.

Cai, Y., X. L. Dong, A. Halevy, J. M. Liu and J. Madhavan (2005) Personal Information Management with Semex. *Proceedings of the 2005 ACM SIGMOD*

International Conference on Management of Data. Baltimore, Maryland, ACM.

Cao, G., J.-Y. Nie, J. Gao and S. Robertson (2008) Selecting Good Expansion Terms for Pseudo-Relevance Feedback. *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.* Singapore, Singapore, ACM.

Carpineto, C. and G. Romano (2012) A Survey of Automatic Query Expansion in Information Retrieval. *ACM Comput. Surv.,* 44**,** 1-50.

Casale, M. B. and F. G. Ashby (2008) A Role for the Perceptual Representation Memory System in Category Learning. *Attention, Perception and Psychophysics,* 70**,** 983-999.

Casey, M. A., R. Veltkamp, M. Goto, M. Leman, C. Rhodes and M. Slaney (2008) Content-Based Music Information Retrieval: Current Directions and Future Challenges. *Proceedings of the IEEE,* 96**,** 668-696.

Chen, M., H. Chu and Y. Chen (2010) Developing a Semantic-Enable Information Retrieval Mechanism. *Expert Systems with Applications,* 37**,** 322-340.

Cheng, E., F. Jing, L. Zhang and H. Jin (2006) Scalable Relevance Feedback Using Click-through Data for Web Image Retrieval. *Proceedings of the 14th Annual ACM International Conference on Multimedia.* Santa Barbara, CA, USA, ACM.

Chinchor, N. (1998) Muc-7 Named Entity Task Definition (Version 3.5). *MUC-7.* Fairfax, Virginia.

Collins-Thompson, K. (2009) Reducing the Risk of Query Expansion Via Robust Constrained Optimization. *Proceedings of the 18th ACM Conference on Information and Knowledge Management.* Hong Kong, China, ACM.

Collins, A. M. and M. R. Quillian (1969) Retrieval Time from Semantic Memory. *Computation Intelligence: Collected Readings.* American Association for Artificial Intelligence.

Conway, M. A. (2005) Memory and the Self. *Journal of Memory and Language,* 53**,** 594-628.

Conway, M. A. and C. W. Pleydell-Pearce (2000) The Construction of Autobiographical Memories in the Self-Memory System. *Psychological Review,* 107**,** 261.

Cover, T. M. and J. A. Thomas (2006) *Elements of Information Theory*, Wiley-Interscience.

Cucerzan, S. (2007) Large-Scale Named Entity Disambiguation Based on Wikipedia Data *In Proc. 2007 Joint Conference on EMNLP and CNLL.*

Davenport, T. H., D. W. De Long and M. C. Beers (1998) Successful Knowledge Management Projects. *Sloan Management Review,* 39**,** 43-57.

Deerwester, S., S. T. Dumais, G. W. Furnas, T. K. Landauer and R. Harshman (1990) Indexing by Latent Semantic Analysis. *Journal of the American Society for Information Science,* 41**,** 391-407.

Dittrich, J.-P., M. A. V. Salles, D. Kossmann and L. Blunschi (2005) Imemex: Escapes from the Personal Information Jungle. *Proceedings of the 31st International Conference on Very Large Data Bases.* Trondheim, Norway, VLDB Endowment.

Dittrich, J., M. A. V. Salles and L. Blunschi (2008) Managing Personal Information Using Itrails. *SIGCHI PIM Workshop.* Florence, Italy.

Downes, S. (2005) Semantic Networks and Social Networks. *The Learning Organization,* 12**,** 411-417.

Fernandez, M., A. Gomez-Perez and N. Juristo (1997) Methontology: From Ontological Art Towards Ontological Engineering.

Gaeta, M., F. Orciuoli, S. Paolozzi and S. Salerno (2011) Ontology Extraction for Knowledge Reuse: The E-Learning Perspective. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on,* 41**,** 798-809.

Gemmell, J., G. Bell and R. Lueder (2006) Mylifebits: A Personal Database for Everything. *Commun. ACM,* 49**,** 88-95.

Gemmell, J., G. Bell, R. Lueder, S. Drucker and C. Wong (2002) Mylifebits: Fulfilling the Memex Vision. *Proceedings of the 10th ACM International Conference on Multimedia.* Juan-les-Pins, France, ACM.

Gheyas, I. A. and L. S. Smith (2010) Feature Subset Selection in Large Dimensionality Domains. *Pattern Recognition,* 43**,** 5-13.

Grishman, R. and B. Sundheim (1996) Message Understanding Conference-6: A Brief History. *Proceedings of the 16th Conference on Computational Linguistics.* Copenhagen, Denmark, Association for Computational Linguistics.

Groza, T., S. Handschuh and K. Moeller (2007) The Nepomuk Project - the Nepomuk Project-on the Way to the Social Semantic Desktop. *Proceedings of International Conferences on new Media Technology (I-MEDIA-2007) and Semntic Systems (I-SEMANTICS-07).* Graz, Austria.

Gruber, T. R. (1993) A Translation Approach to Portable Ontology Specifications. *Knowl. Acquis.,* 5**,** 199-220.

Guarino, N. (1998) *Formal Ontology in Information Systems: Proceedings of the First International Conference (Fois'98), June 6-8, Trento, Italy*, Ios Pr Inc.

Gudivada, V. N., V. V. Raghavan, W. I. Grosky and R. Kasanagottu (1997) Information Retrieval on the World Wide Web. *Internet Computing, IEEE,* 1**,** 58-68.

Guo, J., G. Xu, X. Cheng and H. Li (2009) Named Entity Recognition in Query. *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval.* Boston, MA, USA, ACM.

Haas, K., P. Mika, P. Tarjan and R. Blanco (2011) Enhanced Results for Web Search. *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval.* Beijing, China, ACM.

Haber, D. (2006) Life Review: Implementation, Theory, Research, and Therapy. *The International Journal of Aging and Human Development,* 63**,** 153-171.

Haight, B. K., D. L. Bachman, S. Hendrix, M. T. Wagner, A. Meeks and J. Johnson (2003) Life Review: Treating the Dyadic Family Unit with Dementia. *Clinical Psychology and Psychotherapy,* 10**,** 165-174.

Haight, B. K. and I. Burnside (1993) Reminiscence and Life Review: Explaining the Differences. *Archives of Psychiatric Nursing,* 7**,** 91-98.

Haight, B. K., F. Gibson and Y. Michel (2006) The Northern Ireland Life Review/Life Storybook Project for People with Dementia. *Alzheimer's and Dementia,* 2**,** 56-58.

Havighurst, R. J. and R. Glasser (1972) An Exploratory Study of Reminiscence. *Journal of Gerontology,* 27**,** 245-253.

Hodges, S., E. Berry and K. Wood (2011) Sensecam: A Wearable Camera That Stimulates and Rehabilitates Autobiographical Memory. *Memory,* 19**,** 685-696.

Hodges, S., L. Williams, E. Berry, S. Izadi, J. Srinivasan, A. Butler, G. Smyth, N. Kapur and K. Wood (2006) Sensecam: A Retrospective Memory Aid. *UbiComp 2006: Ubiquitous Computing.* Springer Berlin / Heidelberg.

Hofmann, T. (1999) Probabilistic Latent Semantic Indexing. *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.* Berkeley, California, United States, ACM.

Hoven, E. v. d., C. Sas and S. Whittaker (2012) Introduction to This Special Issue on Designing for Personal Memories: Past, Present, and Future. *Human–Computer Interaction,* 27**,** 1-12.

Huang, J., J. Gao, J. Miao, X. Li, K. Wang, F. Behr and C. L. Giles (2010) Exploring Web Scale Language Models for Search Query Processing. *Proceedings of the 19th International Conference on World Wide Web.* Raleigh, North Carolina, USA, ACM.

Jiang, X. and A.-H. Tan (2009) Learning and Inferencing in User Ontology for Personalized Semantic Web Search. *Information Sciences,* 179**,** 2794-2808.

Jing, Y. and W. B. Croft (1994) An Association Thesaurus for Information Retrieval. *RIAO 94 Conference Proceedings.*

Joachims, T., L. Granka, B. Pan, H. Hembrooke and G. Gay (2005) Accurately Interpreting Clickthrough Data as Implicit Feedback. *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.* Salvador, Brazil, ACM.

Jones, K. S. (1972) A Statistical Interpretation of Term Specificity and Its Application in Retrieval. *Journal of Documentation,* 28**,** 11-21.

Jones, K. S., S. Walker and S. Robertson (2000) A Probabilistic Model of Information Retrieval: Development and Comparative Experiments. *Information Processing and Management,* 36 (6)**,** 779-808, 809-840.

Jones, W. and J. Teevan (2007) *Personal Information Management*, Seattle: University of Washington Press.

Kazama, J. i. and K. Torisawa (2007) Exploiting Wikipedia as External Knowledge for Named Entity Recognition. *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning.*

Keil, F. (1979) *Semantic and Conceptual Development: An Ontological Perspective*, Cambridge, MA :Harvard Univ Press.

Kersten, M., G. Weikum, M. Franklin, D. Keim, A. Buchmann and S. Chaudhuri (2003) A Database Striptease or How to Manage Your Personal Databases.

Proceedings of the 29th International Conference on Very Large Data Bases. Berlin, Germany, VLDB Endowment.

Kiryakov, A., B. Popov, I. Terziev, D. Manov and D. Ognyanoff (2004) Semantic Annotation, Indexing, and Retrieval. *Web Semantics: Science, Services and Agents on the World Wide Web,* 2**,** 49-79.

Kittur, A., E. H. Chi and B. Suh (2008) Crowdsourcing User Studies with Mechanical Turk. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems.* ACM.

Koller, D. and M. Sahami (1996) Toward Optimal Feature Selection. *Proceedings 13th International Conference on Machine Learning.* Bari, Italy, Morgan Kaufmann, San Mateo, CA.

Kumaran, G. and V. R. Carvalho (2009) Reducing Long Queries Using Query Quality Predictors. *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval.* Boston, MA, USA, ACM.

Lafferty, J. and C. Zhai (2001) Document Language Models, Query Models, and Risk Minimization for Information Retrieval. *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.* New Orleans, Louisiana, United States, ACM.

Lau, T. and E. Horvitz (1999) Patterns of Search: Analyzing and Modeling Web Query Refinement. *Proceedings of the 7th International Conference on User Modeling.* Banff, Canada, Springer-Verlag New York, Inc.

Lee, K. S., W. B. Croft and J. Allan (2008) A Cluster-Based Resampling Method for Pseudo-Relevance Feedback. *Proceedings of the 31st annual International ACM SIGIR Conference on Research and Development in Information Retrieval.* Singapore, Singapore, ACM.

Lew, M. S., N. Sebe, C. Djeraba and R. Jain (2006) Content-Based Multimedia Information Retrieval: State of the Art and Challenges. *ACM Trans. Multimedia Comput. Commun. Appl.,* 2**,** 1-19.

Li, Z. and K. Ramani (2007) Ontology-Based Design Information Extraction and Retrieval. *AI EDAM,* 21**,** 137-154.

Liu, C., R. W. White and S. Dumais (2010) Understanding Web Browsing Behaviors through Weibull Analysis of Dwell Time. *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval.* Geneva, Switzerland, ACM.

Liu, F., C. Yu and W. Meng (2004) Personalized Web Search for Improving Retrieval Effectiveness. *IEEE Trans. on Knowl. and Data Eng.,* 16**,** 28-40.

Lu, Y., F. Peng, X. Wei and B. Dumoulin (2010) Personalize Web Search Results with User's Location. *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval.* Geneva, Switzerland, ACM.

Lv, Y. and C. Zhai (2009) Adaptive Relevance Feedback in Information Retrieval. *Proceedings of the 18th ACM Conference on Information and Knowledge Management.* Hong Kong, China, ACM.

Mandala, R., T. Tokunaga and H. Tanaka (2000) Query Expansion Using Heterogeneous Thesauri. *Inf. Process. Manage.,* 36**,** 361-378.

Manning, C. D., P. Raghavan and H. Schutze (2008) *Introduction to Information Retrieval*, Cambridge University Press.

Maron, M. and J. Kuhns (1960) On Relevance, Probabilistic Indexing and Information Retrieval. *Journal of the ACM (JACM),* 7**,** 216-244.

Mather, M. and L. L. Carstensen (2005) Aging and Motivated Cognition: The Positivity Effect in Attention and Memory. *Trends in Cognitive Sciences,* 9**,** 496-502.

Maybury, M. (1997) Intelligent Multimedia Information Retrieval. *AI Magazine,* 18**,** 478.

Miller, G. A. (1956) The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information. *Psychological Review,* 63**,** 81-97.

Miller, G. A. (2003) The Cognitive Revolution: A Historical Perspective. *Trends in Cognitive Sciences,* 7**,** 141-144.

Nagypál, G. (2005) Improving Information Retrieval Effectiveness by Using Domain Knowledge Stored in Ontologies. *Proceedings of the 2005 OTM Confederated International Conference.* Agia Napa, Cyprus, Springer-Verlag.

Nguyen, T. and T. Phan (2008) The Effect of Semantic Index in Information Retrieval Development. *Proceedings of the 10th International Conference on Information Integration and Web-based Applications and Services.* Linz, Austria, ACM.

Pasupathi, M. and L. L. Carstensen (2003) Age and Emotional Experience During Mutual Reminiscing. *Psychology and aging,* 18**,** 430.

Peesapati, S. T., V. Schwanda, J. Schultz, M. Lepage, S.-y. Jeong and D. Cosley (2010) Pensieve: Supporting Everyday Reminiscence. *Proceedings of the 28th International Conference on Human Factors in Computing Systems.* Atlanta, Georgia, USA, ACM.

Petrakis, E. G. M., G. Varelas, A. Hliaoutakis and P. Raftopoulou (2006) X-Similarity: Computing Semantic Similarity between Concepts from Different Ontologies. *Journal of Digital Information Management,* 4**,** 233.

Popov, B., A. Kiryakov, D. Ognyanoff, D. Manov and A. Kirilov (2004) Kim–a Semantic Platform for Information Extraction and Retrieval. *Natural Language Engineering,* 10**,** 375-392.

Porter, M. (1980) An Algorithm for Suffix Stripping. *Program: Electronic Library and Information Systems,* 14**,** 130-137.

Rada, R., H. Mili, E. Bicknell and M. Blettner (1989) Development and Application of a Metric on Semantic Nets. *Systems, Man and Cybernetics, IEEE Transactions on,* 19**,** 17-30.

Razmerita, L. (2011) An Ontology-Based Framework for Modeling User Behavior: A Case Study in Knowledge Management. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on,* 41**,** 772-783.

Resnik, P. (1995) Using Information Content to Evaluate Semantic Similarity in a Taxonomy. *Proceedings of the 14th International Joint Conference on Artificial Intelligence.* Montreal, Quebec, Canada, Morgan Kaufmann Publishers Inc.

Rhodes, B. J. and P. Maes (2000) Just-in-Time Information Retrieval Agents. *IBM Syst. J.,* 39**,** 685-704.

Ritter, A., S. Soderl and O. Etzioni (2009) What Is This, Anyway: Automatic Hypernym Discovery. *In Proceedings of AAAI-09 Spring Symposium on Learning.*

Robertson, S. (1977) The Probability Ranking Principle in Ir. *Journal of Documentation,* 33**,** 294-304.

Rubin, D. C., T. A. Rahhal and L. W. Poon (1998) Things Learned in Early Adulthood Are Remembered Best. *Memory and Cognition,* 26**,** 3-19.

Ryan, T. and R. Walker (1985) *Making Life Story Books*, London: British Agencies for Adoption and Fosterings.

Salton, G. and C. Buckley (1997) Improving Retrieval Performance by Relevance Feedback. *Readings in Information Retrieval*, 355-364.

Salton, G., A. Wong and C. S. Yang (1975) A Vector Space Model for Automatic Indexing. *Commun. ACM,* 18, 613-620.

Sánchez, D., M. Batet, D. Isern and A. Valls (2012) Ontology-Based Semantic Similarity: A New Feature-Based Approach. *Expert Systems with Applications,* 39, 7718-7728.

Sang, E. F. T. K. and F. D. Meulder (2003) Introduction to the Conll-2003 Shared Task: Language-Independent Named Entity Recognition. *Proceedings of the 7th Conference on Natural Language Learning at HLT-NAACL 2003.* Edmonton, Canada, Association for Computational Linguistics.

Sauermann, L., A. Bernardi and A. Dengel (2005) Overview and Outlook on the Semantic Desktop. *The Semantic Desktop Workshop at the 4th Int. Semantic Web Conf, ISWC 2005.* Galway, Irenland.

Sauermann, L., G. Grimnes, M. Kiesel, C. Fluit, H. Maus, D. Heim, D. Nadeem, B. Horak and A. Dengel (2006) Semantic Desktop 2.0: The Gnowsis Experience. *The Semantic Web - ISWC 2006.* Springer Berlin / Heidelberg.

Sauermann, L., G. Grimnes and T. Roth-Berghofer (2008) The Semantic Desktop as a Foundation for Pim Research. *The Personal Information Management Workshop at ACM Conference on Human Factors in Computing Systems, CHI 2008.* Florence, Italy.

Schacter, D. L. (1990) Perceptual Representation Systems and Implicit Memory. *Annals of the New York Academy of Sciences,* 608, 543-571.

Sekine, S. (2008) Extended Named Entity Ontology with Attribute Information. *Proceedings of the 6th International Language Resources and Evaluation (LREC'08).* Marrakech, Morocco.

Sellen, A. J., A. Fogg, M. Aitken, S. Hodges, C. Rother and K. Wood (2007) Do Life-Logging Technologies Support Memory for the Past: An Experimental Study Using Sensecam. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems.* San Jose, California, USA, ACM.

Serrano, J. P., J. M. Latorre, M. Gatz and J. Montanes (2004) Life Review Therapy Using Autobiographical Retrieval Practice for Older Adults with Depressive Symptomatology. *Psychology and Aging,* 19, 272.

Shadbolt, N., T. Berners-Lee and W. Hall (2006) The Semantic Web Revisited. *IEEE Intelligent Systems,* 21**,** 96-101.

Shen, D., J.-T. Sun, Q. Yang and Z. Chen (2006) Building Bridges for Web Query Classification. *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.* Seattle, Washington, USA, ACM.

Shi, L. and R. Setchi (2010) An Ontology Based Approach to Measuring the Semantic Similarity between Information Objects in Personal Information Collections. *Lecture Notes in Computer Science,* 6276**,** 617-626.

Shi, L. and R. Setchi (2012) User-Oriented Ontology-Based Clustering of Stored Memories. *Expert Systems with Applications,* 39**,** 9730-9742.

Shiri, A. and C. Revie (2006) Query Expansion Behavior within a Thesaurus-Enhanced Search Environment: A User-Centered Evaluation. *Journal of the American Society for Information Science and Technology,* 57**,** 462-478.

Singhal, A. (2001) Modern Information Retrieval: A Brief Overview. *IEEE Data Engineering Bulletin,* 24**,** 35-43.

Sjöberg, M., J. Laaksonen, T. Honkela and M. Pöllä (2008) Inferring Semantics from Textual Information in Multimedia Retrieval. *Neurocomputing,* 71**,** 2576-2586.

Snow, R., D. Jurafsky and A. Y. Ng (2004) Learning Syntactic Patterns for Automatic Hypernym Discovery. *Advances in Neural Information Processing Systems, vol. 17MIT Press.*

Song, F. and W. B. Croft (1999) A General Language Model for Information Retrieval. *Proceedings of the 8th International Conference on Information and Knowledge Management.* Kansas City, Missouri, United States, ACM.

Song, Y. and L.-w. He (2010) Optimal Rare Query Suggestion with Implicit User Feedback. *Proceedings of the 19th International Conference on World Wide Web.* Raleigh, North Carolina, USA, ACM.

Sowa, J. F. (1991) *Principles of Semantic Networks: Principles of Semantic Networks: Explorations in the Representation of Knowledge*, Morgan Kaufmann.

Sowa, J. F. (2006) Semantic Networks. *Encyclopedia of Cognitive Science.* John Wiley and Sons, Ltd.

Spink, A., B. J. Jansen and H. C. Ozmultu (2000) Use of Query Reformulation and Relevance Feedback by Excite Users. *Internet Research,* 10**,** 317-328.

Steyvers, M. and J. B. Tenenbaum (2005) The Large-Scale Structure of Semantic Networks: Statistical Analyses and a Model of Semantic Growth. *Cognitive Science: A Multidisciplinary Journal,* 29**,** 41 - 78.

Suchanek, F., G. Kasneci and G. Weikum (2008) Yago: A Large Ontology from Wikipedia and Wordnet. *Web Semantics: Science, Services and Agents on the World Wide Web,* 6**,** 203-217.

Sunassee, N. N. and D. A. Sewry (2002) A Theoretical Framework for Knowledge Management Implementation. *Proceedings of the 2002 Annual Research Conference of the South African Institute of Computer Scientists and Information Technologists on Enablement through Technology.* Port Elizabeth, South Africa, South African Institute for Computer Scientists and Information Technologists.

Teevan, J., S. T. Dumais and E. Horvitz (2005) Personalizing Search Via Automated Analysis of Interests and Activities. *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.* Salvador, Brazil, ACM.

Thomsen, D. K. and D. Berntsen (2008) The Cultural Life Script and Life Story Chapters Contribute to the Reminiscence Bump. *Memory,* 16**,** 420-435.

Thomsen, D. K., D. B. Pillemer and Z. Ivcevic (2011) Life Story Chapters, Specific Memories and the Reminiscence Bump. *Memory,* 19**,** 267-279.

Thorgrimsen, L., P. Schweitzer and M. Orrell (2002) Evaluating Reminiscence for People with Dementia: A Pilot Study. *The Arts in Psychotherapy,* 29**,** 93-97.

Tulving, E. (1993) What Is Episodic Memory? *Current Directions in Psychological Science,* 2**,** 67-70.

Tulving, E. (2002) Episodic Memory: From Mind to Brain. *Annual Review of Psychology,* 53**,** 1-25.

Uren, V., P. Cimiano, J. Iria, S. Handschuh, M. Vargas-Vera, E. Motta and F. Ciravegna (2006) Semantic Annotation for Knowledge Management: Requirements and a Survey of the State of the Art. *Web Semantics: Science, Services and Agents on the World Wide Web,* 4**,** 14-28.

Vallet, D., M. Fernández and P. Castells (2005) An Ontology-Based Information Retrieval Model. *Proceedings of the Second European conference on The Semantic Web: Research and Applications.* Heraklion, Greece, Springer-Verlag.

Voorhees, E. M. (1994) Query Expansion Using Lexical-Semantic Relations. *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.* Dublin, Ireland, Springer-Verlag New York, Inc.

Wei, X. and W. B. Croft (2006) Lda-Based Document Models for Ad-Hoc Retrieval. *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.* Seattle, Washington, USA, ACM.

White, R. W., C. L. A. Clarke and S. Cucerzan (2007) Comparing Query Logs and Pseudo-Relevance Feedback for Web-Search Query Refinement. *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.* Amsterdam, The Netherlands, ACM.

Whitelaw, C., A. Kehlenbeck, N. Petrovic and L. Ungar (2008) Web-Scale Named Entity Recognition. *Proceedings of the 17th ACM Conference on Information and Knowledge Management.* Napa Valley, California, USA, ACM.

Woods B, Spector AE, Jones CA, OrrellM and D. SP (2005) Reminiscence Therapy for Dementia. *Cochrane Database of Systematic Reviews.*

Xu, J. and W. B. Croft (1996) Query Expansion Using Local and Global Document Analysis. *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.* Zurich, Switzerland, ACM.

Xu, J. and W. B. Croft (2000) Improving the Effectiveness of Information Retrieval with Local Context Analysis. *ACM Trans. Inf. Syst.,* 18**,** 79-112.

Yates, R. and B. Neto (1999) Modern Information Retrieval. *New York, Addision Wesley.*

Appendix A

Experimental Data

This section introduces the experimental data used in this thesis. The data is automatically collected from public resources on the internet, e.g. English News websites, using a crawler developed by the author of the thesis. Figure A – 1 shows the modules and processes of the crawler.

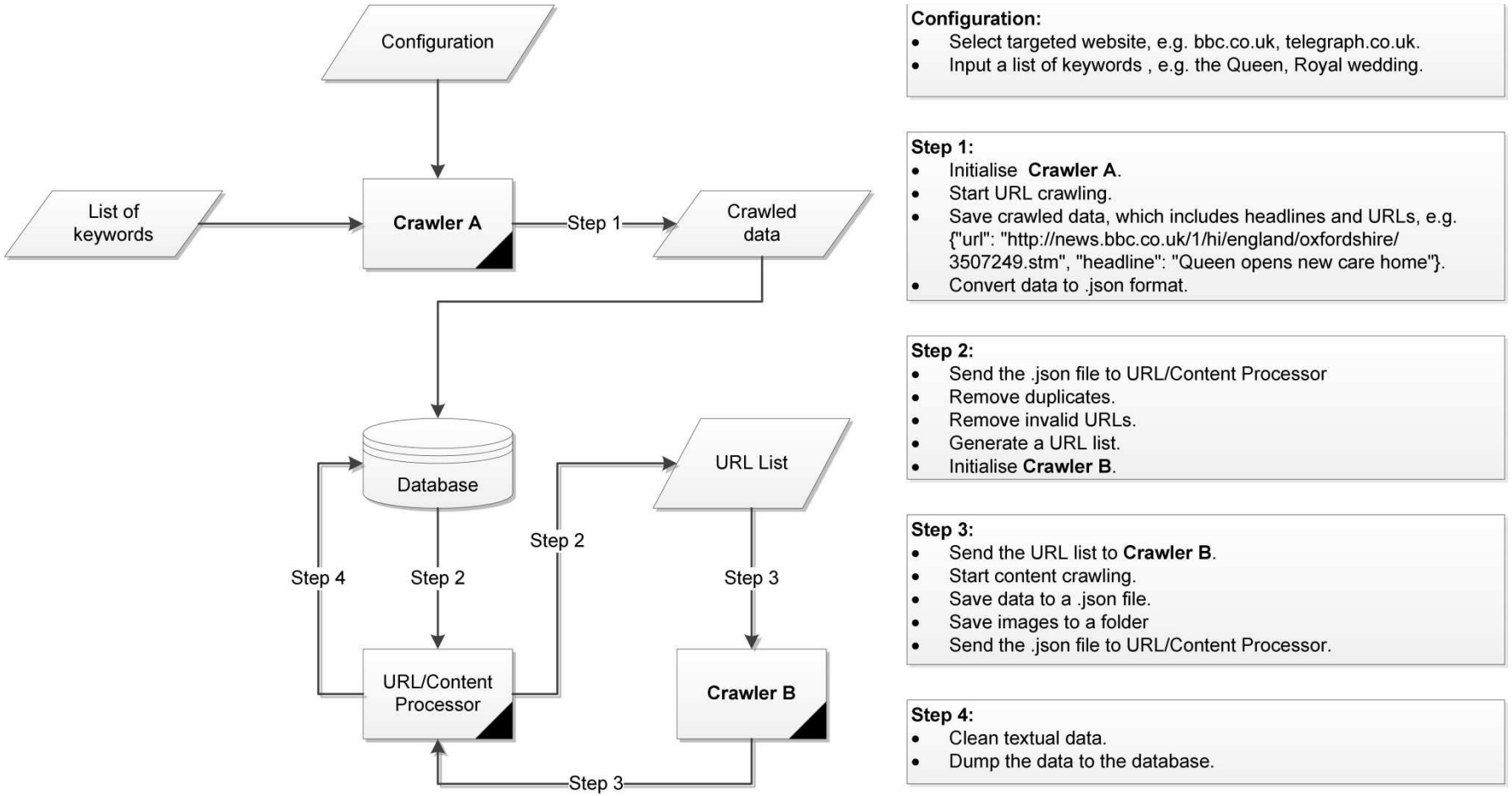


Figure A - 1 Modules of the crawler

The next table shows a URL list that contains headlines and URLs collected in **Step 2**.


```
{"url": "http://www.telegraph.co.uk/news/uknews/1317794/Princess-Royal-accepts-bouquets-after-brickbats.html", "headline":
"Princess Royal accepts bouquets after brickbats"}
{"url": "http://www.telegraph.co.uk/health/1313100/Princess-may-have-suffered-second-stroke.html", "headline": "Princess may have
suffered second stroke"}
{"url": "http://www.telegraph.co.uk/news/uknews/1319368/Hollywoods-glitz-may-be-too-hard-for-him-to-ignore.html", "headline":
"Hollywood's glitz may be too hard for him to ignore"}
{"url": "http://www.telegraph.co.uk/news/uknews/1350938/Why-we-must-learn-to-lose-by-Prince-Philip.html", "headline": "Why we must
learn to lose"}
{"url": "http://www.telegraph.co.uk/news/uknews/1322242/Gordonstoun-plans-mortgage-type-scheme-for-payment-of-fees.html",
"headline": "Gordonstoun plans mortgage-type scheme for payment of fees "}
{"url": "http://www.telegraph.co.uk/news/uknews/1370446/Queen-in-bad-odour-with-perfumer-over-new-scents.html", "headline": "Queen
in bad odour with perfumer over new scents"}
{"url": "http://www.telegraph.co.uk/news/uknews/1346546/Queen-is-described-as-a-bad-mother-in-Morton-programme.html", "headline":
"Queen is described as a bad mother in Morton programme"}
{"url": "http://www.telegraph.co.uk/news/uknews/1542110/Let-us-show-the-world-we-are-united-in-grief.html", "headline": "Let us
show the world we are united in grief"}
{"url": "http://www.telegraph.co.uk/news/obituaries/1352792/Captain-Jim-Pertwee.html", "headline": "Captain Jim Pertwee"}
{"url": "http://www.telegraph.co.uk/news/uknews/1374973/Pheasant-feathers-in-Queens-hat-are-message-to-critics.html", "headline":
"Pheasant feathers in Queen's hat are message to critics "}
{"url": "http://www.telegraph.co.uk/news/uknews/1348771/Royal-pageant-given-wings-by-boy-soprano.html", "headline": "Royal pageant
given wings by boy soprano"}
{"url": "http://www.telegraph.co.uk/news/uknews/1349988/Messages-from-Machiavelli.html", "headline": "Messages from Machiavelli"}
{"url": "http://www.telegraph.co.uk/news/uknews/1379544/Harrods-renounces-claim-to-the-throne.html", "headline": "Harrods renounces
claim to the throne"}
```

{"url": "http://www.telegraph.co.uk/news/uknews/1354362/Princess-calls-for-open-mind-on-GM.html", "headline": "Princess calls for open mind on GM"}
{"url": "http://www.telegraph.co.uk/news/uknews/1317878/Heath-decided-conference-was-not-fit-for-the-Queen.html", "headline": "Heath decided conference was not fit for the Queen"}
{"url": "http://www.telegraph.co.uk/news/uknews/1322131/Revealed-the-Queens-plans-for-Prince-Philips-80th.html", "headline": "Revealed: the Queen's plans for "}
{"url": "http://www.telegraph.co.uk/health/1314278/Margaret-in-hospital-as-fears-grow.html", "headline": "Queen loses role as final arbiter for all students"}
{"url": "http://www.telegraph.co.uk/news/obituaries/1328746/Lord-Morris-of-Castle-Morris.html", "headline": "Lord Morris of Castle Morris"}
{"url": "http://www.telegraph.co.uk/news/obituaries/1328354/Daniel-Counihan.html", "headline": "Daniel Counihan"}
{"url": "http://www.telegraph.co.uk/news/uknews/1327860/Queen-is-understanding-about-horse-shows-cancellation.html", "headline": "Queen is 'understanding' about horse show's cancellation "}
{"url": "http://www.telegraph.co.uk/news/uknews/1350923/Record-sales-for-Queen-Mother-centenary-memorabilia.html", "headline": "Record sales for Queen Mother centenary memorabilia"}
{"url": "http://www.telegraph.co.uk/news/uknews/1342729/Queen-ends-Duchesss-exile-with-invitation-to-the-ball.html", "headline": "Queen ends Duchess's exile with invitation to the ball"}
{"url": "http://www.telegraph.co.uk/news/uknews/1310837/Chris-Smith-criticises-countesss-behaviour.html", "headline": "Chris Smith criticises countess's behaviour"}
{"url": "http://www.telegraph.co.uk/news/1335499/What-the-tabloids-say.html", "headline": "What the tabloids say"}
{"url": "http://www.telegraph.co.uk/news/1334197/Duchess-had-nervous-breakdown.html", "headline": "Duchess had nervous breakdown"}
{"url": "http://www.telegraph.co.uk/news/uknews/1358086/Scottish-hoteliers-get-charm-lessons.html", "headline": "Scottish hoteliers get charm lessons"}
{"url": "http://www.telegraph.co.uk/news/obituaries/1338438/Sir-Roderick-Sarell.html", "headline": "Sir Roderick Sarell"}
{"url": "http://www.telegraph.co.uk/news/obituaries/1336676/Larry-Adler.html", "headline": "Larry Adler"}

```
{"url": "http://www.telegraph.co.uk/news/uknews/1308944/Dusty-welcome-stirs-memories-for-the-Queen.html", "headline": "Dusty welcome stirs memories for the Queen"}
{"url": "http://www.telegraph.co.uk/news/uknews/1331891/Old-soldiers-final-march-for-the-Forgotten-Army.html", "headline": "Old soldiers' final march for the Forgotten Army"}
{"url": "http://www.telegraph.co.uk/news/obituaries/1312242/The-Very-Reverend-Ivan-Neill.html", "headline": "The Very Reverend Ivan Neill"}
{"url": "http://www.telegraph.co.uk/news/uknews/1384463/Town-gives-its-heart-to-a-grieving-sister.html", "headline": "Town gives its heart to a grieving sister"}
{"url": "http://www.telegraph.co.uk/health/3294556/The-real-Elizabeth-II-part-two.html", "headline": "The real Elizabeth II: part two"}
{"url": "http://www.telegraph.co.uk/news/uknews/1340091/Queen-Mother-at-church-service.html", "headline": "Queen Mother at church service"}
{"url": "http://www.telegraph.co.uk/news/uknews/1340756/Queen-Mother-joins-in-prayers.html", "headline": "Queen Mother joins in prayers"}
{"url": "http://www.telegraph.co.uk/news/uknews/1333077/Will-I-marry-again-Who-knows-what-the-good-Lord-has-planned.html",
"headline": "Will I marry again? Who knows what the good Lord has planned "}
{"url": "http://www.telegraph.co.uk/news/uknews/1309977/Queens-birthday-tribute-to-her-loyal-husband.html", "headline": "Queen's birthday tribute to her loyal husband"}
{"url": "http://www.telegraph.co.uk/health/3294423/1952-the-way-we-were.html", "headline": "1952: the way we were"}
{"url": "http://www.telegraph.co.uk/news/uknews/1364342/Queen-invites-cardinal-to-stay-at-Sandringham.html", "headline": "Queen invites cardinal to stay at Sandringham"}
```


The content of articles are analysed and collected by **Crawler B** based on the URLs list above. The analysed content fields of the article include `"originalPublicationDate"`, `"subtitle"`, `"description"`, `"url"`, `"section"`, `"image_urls"`, `"content"`,

“headline” and “images” (see the next table). Textual data and images are stored in a database and directories, respectively (as shown in Figure A – 2).

{"originalPublicationDate": ["2011-02-09"], "subtitle": [], "description": ["Telegraph View: A royal visit looms. Or does it?"], "url": "http://www.telegraph.co.uk/news/uknews/queen-elizabeth-II/8314653/Ireland-awaits-the-Queen.html%0A", "section": ["news-uk_news-queen_elizabeth_11"], "image_urls": ["http://i.telegraph.co.uk/multimedia/archive/01661/queenLiz_1661712c.jpg"], "content": ["\nWill she or won\u2019t she? And if not, why not? There is growing speculation in \n Ireland that the Queen might be about to make the first visit by a reigning \n monarch for 100 years. Her Majesty\u2019s grandfather, George V, included Ireland \n on his accession tour of the empire in 1911. But, for obvious reasons, a \n follow-up visit has been considered unwise. For years now, there has been \n talk of the Queen visiting Dublin, but nothing has come of it. Mary \n McAleese, Ireland\u2019s president, has expressed the hope that she might preside \n over a state visit before her term of office ends in November. It looks like \n she may get her wish, with a spring visit in May being talked of. Enda \n Kenny, the Fine Gael leader expected to be the next Taoiseach, said a royal \n visit would be \u201cvery warmly received\u201d. Gerry Adams of Sinn Fein is less \n keen. That seems a good reason why one should be arranged.\n"], "headline": ["Ireland awaits the Queen"], "imageCaption": ["Mr Cowen said he wanted the Queen's visit to take place before November 2011."], "images": [{"url": "http://i.telegraph.co.uk/multimedia/archive/01661/queenLiz_1661712c.jpg", "path": "full/408fcd3a07904e6730b85df133beda09e9c0888f.jpg", "checksum": "d3e5aff3b3cf9dab8a7270769a70a6b4"}]}

{"originalPublicationDate": ["2010-12-31"], "subtitle": [], "description": ["The Queen expressed her delight at becoming a great grandmother yesterday \n after Autumn Phillips, wife of her grandson Peter, gave birth to a baby girl."], "url": "http://www.telegraph.co.uk/news/newstopics/theroyalfamily/8232330/Queen-delighted-at-birth-of-great-granddaughter.html%0A", "section": ["news-news_topics-the_royal_family"], "image_urls": ["http://i.telegraph.co.uk/multimedia/archive/01794/Peter-Phillips-aut_1794538c.jpg"], "content": ["\nThe newborn, who weighed 8lb 8oz, is 12th in line to the throne but her name \n has not yet been confirmed by Buckingham Palace. \n", "\nShe is Mr and Mrs Phillips\u2019 first child, the first grandchild for the Princess \n Royal and the first great-grandchild for the Queen and the Duke of \n Edinburgh. \n", "\nA statement from Buckingham Palace read: \u201cThe Queen, the Duke of Edinburgh, \n the Princess Royal, Captain Mark Phillips and Autumn's family have been \n informed and are delighted with the news.\u201d\n", "\nMr Phillips, the Queen\u2019s eldest grandson, was by his Canadian-born wife\u2019s side \n at the birth at Gloucestershire Royal Hospital on Wednesday. \n", "\nThe couple met at the Montreal Grand Prix in 2003 and were married five years \n later at a glittering ceremony at Windsor castle. \n"], "headline": ["Queen 'delighted' at birth of great-granddaughter"], "imageCaption": ["Mr Phillips, the Queen's eldest grandson, was by his Canadian-born wife's side at the birth"], "images": [{"url": "http://i.telegraph.co.uk/multimedia/archive/01794/Peter-Phillips-aut_1794538c.jpg", "path": "full/850a23f4258ed6491bd0d5d540eeed29b088413c.jpg", "checksum": "394094a180fcf41f889a9fc5e8a77bf3"}]}

{"originalPublicationDate": ["2010-11-21"], "subtitle": [], "description": ["Sophie Winkleman appeared topless in an 'art-house' film before she married Lord Frederick Windsor."], "url": "http://www.telegraph.co.uk/news/newstopics/celebritynews/8149113/Royal-bride-Sophie-Winkleman-is-alarmed-by-the-release-of-her-risque-film.html%0A", "section": ["news-news_topics-celebrity_news"], "image_urls": ["http://i.telegraph.co.uk/multimedia/archive/01767/Sophie-Winkleman_1767213b.jpg"], "content": ["Courtiers are confident that Kate Middleton has no skeletons in her closet, but the last royal bride, Sophie Winkleman, has received an unsettling blast from the past. ", "The actress, who married Lord Frederick Windsor at Hampton Court Palace last year, has learnt that the film ", ", in which she appears topless, has been released as a DVD around the world. Happily, it will not be sold in Britain, to spare her blushes.", "The

film begins with a scene in which Sophie engages in an act that does not regularly feature on television screens at royal palaces.", "Before her wedding, Patrick Fischer, the producer, told me: \"The sex scenes are all done very tastefully. It is an art film and should be treated in that way.", "\"If she did not want the film to appear, then I would definitely take that on board as the producer, but also as a friend.\""], "headline": ["Royal bride Sophie Winkleman is alarmed by the release of her risqu\u00e9 film "], "imageCaption": ["Lord Freddie Windsor and Sophie Winkleman wedding"], "images": [{"url": "http://i.telegraph.co.uk/multimedia/archive/01767/Sophie-Winkleman_1767213b.jpg", "path": "full/c6cac6cecbb331e08d0bf84fc0be0ddc9ebc7aff.jpg", "checksum": "d0e9e9495f8bca54e4833db02fa190f2"}]}

{"originalPublicationDate": ["2011-01-05"], "subtitle": [], "description": ["Singer Charlotte Church has claimed the Queen 'has no idea what is going on'."], "url": "http://www.telegraph.co.uk/news/newstopics/celebritynews/8239165/Charlotte-Church-Queen-has-no-idea-what-is-going-on.html%0A", "section": ["news-news_topics-celebrity_news"], "image_urls": ["http://i.telegraph.co.uk/multimedia/archive/01796/charlotte-church_1796798c.jpg"], "content": ["\nThe Welsh diva has risked upsetting the royal family with her comments in the \n new edition of Esquire magazine. \n", "\nShe said: \"I've met her about seven times and she never remembers me. When you \n get close to her you realise she's an old woman and has no idea what's going \n on. \n", "\n\"I feel really sorry for her. She probably doesn't want to be wheeled out at \n every Royal Variety Performance to watch scantily clad dancers and s***** \n comedians.\" \n", "\nThe mother of two also said she has never been chatted up and always does the \n chasing herself. \n", "\nChurch, 24, told the February edition of Esquire, which is on sale from \n Thursday, she has a 100 per cent success rate with all the men she has \n approached. \n"], "headline": ["Charlotte Church: 'Queen has no idea what is going on'"], "imageCaption": ["Charlotte Church, right, meeting the Queen at the 2005 Royal Variety Performance"], "images": [{"url": "http://i.telegraph.co.uk/multimedia/archive/01796/charlotte-church_1796798c.jpg", "path": "full/2b7d9667b476e009f6b587733f6265628a7b0486.jpg", "checksum": "c25ae0d4de3a0119e3f9561db6e5b0ad"}]}

{"originalPublicationDate": ["2010-12-05"], "subtitle": [], "description": ["Campaigners in Malaysia have petitioned the Queen to use her influence in gaining an apology and compensation from the British government over an alleged massacre of 24 unarmed villagers by British soldiers in 1948."], "url": "http://www.telegraph.co.uk/news/worldnews/asia/malaysia/8182282/Malaysian-campaigners-ask-Queen-to-press-for-action-over-1948-deaths.html%0A", "section": ["news-world_news-asia-malaysia"], "image_urls": ["http://i.telegraph.co.uk/multimedia/archive/01676/queen_1676535c.jpg"], "content": ["The move comes after a request was rejected for an investigation into the killing of the 24 ethnic Chinese in the remote village of Batang Kali, Selangor province, despite a decades-long campaign. A lawyer acting for the victims' families, Quek Ngee Meng, criticised the British government's decision as \"legally and morally hollow\", adding that the failure to hold an inquiry amounted to a \"very British cover-up\".", "The families along with a delegation from the Chinese Associations' Federation have presented their petition in Kuala Lumpur. It asks that the Queen use her \"vast influence\" over the government to ensure it issues an \"official apology\" and \"reasonable compensation\" for the victims' families and the wider community, which it sets at \u00a330 million and \u00a350 million respectively.", " The families contend that a contingent of 14 Scots Guards entered the village on 12 December, 1948, and detained 25 unarmed men at the beginning of a 12-year insurgency in what was then the colony of Malaya.", " Twenty-four of the men were killed while one, who fainted and was presumed dead, survived and is still alive. The men's wives and children who had been separated from them witnessed the killings.", " At the time a British investigation into the massacre found they were killed on a river bank to prevent them from escaping."], "headline": ["Malaysian campaigners ask Queen to press for action over 1948 deaths "], "imageCaption": ["The petition asks that the Queen use her 'vast influence' over the government to ensure it issues an 'official apology'"], "images": [{"url":

"http://i.telegraph.co.uk/multimedia/archive/01676/queen_1676535c.jpg", "path": "full/19bb14cb6e8c4b7968b0d66c5753e88b8b446d4e.jpg", "checksum": "740750428a576deb2a94a056c2a661f8"}]}

{"originalPublicationDate": ["2010-11-14"], "subtitle": [], "description": ["The Queen`s composer has said he will never to wear a poppy again because he \n feels the cause has been \"hijacked\" to support the wars in \n Afghanistan and Iraq."], "url": "http://www.telegraph.co.uk/news/uknews/8132185/Remembrance-Sunday-Queens-composer-says-he-will-boycott-poppies.html%0A", "section": ["news-uk_news"], "image_urls": ["http://i.telegraph.co.uk/multimedia/archive/01475/Peter-Maxwell-Davi_1475840c.jpg"], "content": ["\nSir Peter Maxwell Davies - who lives on Orkney - composed one of the sentinel \n pieces for last year`s national commemoration of the war dead - symbolically \n marking the death of Harry Patch, the last British survivor of the World War \n I trenches.\n", "\nBut Sir Peter decided for the first time this year not to buy a poppy or \n attend any Remembrance Day ceremonies in protest over the wars in \n Afghanistan and Iraq.\n", "\nHe blamed politicians for \"spinning\" Remembrance Day into an event \n that \"legitimised\" those recent conflicts together with those \"entirely \n justified\" two great world wars.\n", "\nBut Sir Peter recognised his decision would cause anger among many veterans \n but felt he must be at peace with his conscience.\n", "\n\"This is the first year I have decided not to buy a poppy. I am not a \n pacifist and I back the need for the first two world wars and pay tribute to \n the huge sacrifice made by those who took part in those wars,\" said Sir \n Peter, 75, from his home on Sanday.\n"], "headline": ["Remembrance Sunday: Queen's composer says he will boycott poppies"], "imageCaption": ["Composer Sir Peter Maxwell Davies"], "images": [{"url": "http://i.telegraph.co.uk/multimedia/archive/01475/Peter-Maxwell-Davi_1475840c.jpg", "path": "full/47bcca24baaa724ba1773ea45133171d3f576b8d.jpg", "checksum": "d0586b07fdf88b253784334ec713662f"}]}

{"originalPublicationDate": ["2010-07-03"], "subtitle": [], "description": ["Madonna has cast Natalie Dormer as Queen Elizabeth, the Queen Mother in a film \n about the Edward VIII abdication crisis."], "url": "http://www.telegraph.co.uk/news/newstopics/celebritynews/madonna/7870311/Madonna-casts-Natalie-Dormer-to-portray-the-game-playing-Queen-Mother.html%0A", "section": ["news-news_topics-celebrity_news-madonna"], "image_urls": ["http://i.telegraph.co.uk/multimedia/archive/01547/Madonna1_1547006c.jpg", "http://i.telegraph.co.uk/multimedia/archive/01637/queen-mother_1637852c.jpg"], "content": ["Much missed since her death in 2002, Queen Elizabeth, the Queen Mother is to be subjected to a radical new portrayal by Madonna. Mandrake can disclose that the American singer has cast an actress who appeared naked as Anne Boleyn in the television series ", " to perform the role of Queen Elizabeth in a film that she is making about the Edward VIII abdication crisis.", "\"This country tends to remember the Queen Mother as a rather wrinkly 97 year-old, but I am playing her when she was quite an enchanting, engaging twenty and thirtysomething,\" Natalie Dormer told me. \"She was quite a savage and savvy game player.\"", "The \"Queen of Pop\" is understood to be portraying Queen Elizabeth as a highly negative influence in ", ", which is sympathetic to Wallis Simpson, the twice-divorced American socialite, who caused a constitutional dilemma when the king declared his desire to marry her.", "Dormer, who was speaking at a party to mark the 100th anniversary of Elizabeth Arden, the cosmetics firm patronised by Queen Elizabeth, said: \"She never really spoke in public \u2013 she left it up to her husband, the king, and, therefore, even if she was a dominant force in the marriage, and a rock and forward thinker, she always let the king take the lead. She never took any credit for it.\"", "Madonna has long been fascinated by the 1936 crisis, which was resolved by Edward's abdication and his succession by his brother, George VI, who was the present Queen's father. Edward was given the title the Duke of Windsor and he married Simpson."], "headline": ["Madonna casts Natalie Dormer to portray the 'game-playing' Queen Mother "], "imageCaption": ["Madonna is making a film about the Edward VIII abdication crisis.", "The Queen Mother"], "images": [{"url": "http://i.telegraph.co.uk/multimedia/archive/01547/Madonna1_1547006c.jpg", "path": "full/e83970088593284f712f6a18acfc4c093725d093.jpg", "checksum": "fe3c7040951c33ec0600854a3dfdc323"}, {"url":

"http://i.telegraph.co.uk/multimedia/archive/01637/queen-mother_1637852c.jpg", "path":
"full/43f62d1384860caa52477f7f336a68040e690ef2.jpg", "checksum": "b3a5c82508e8f2e06037b70e05bddaf5"}]}

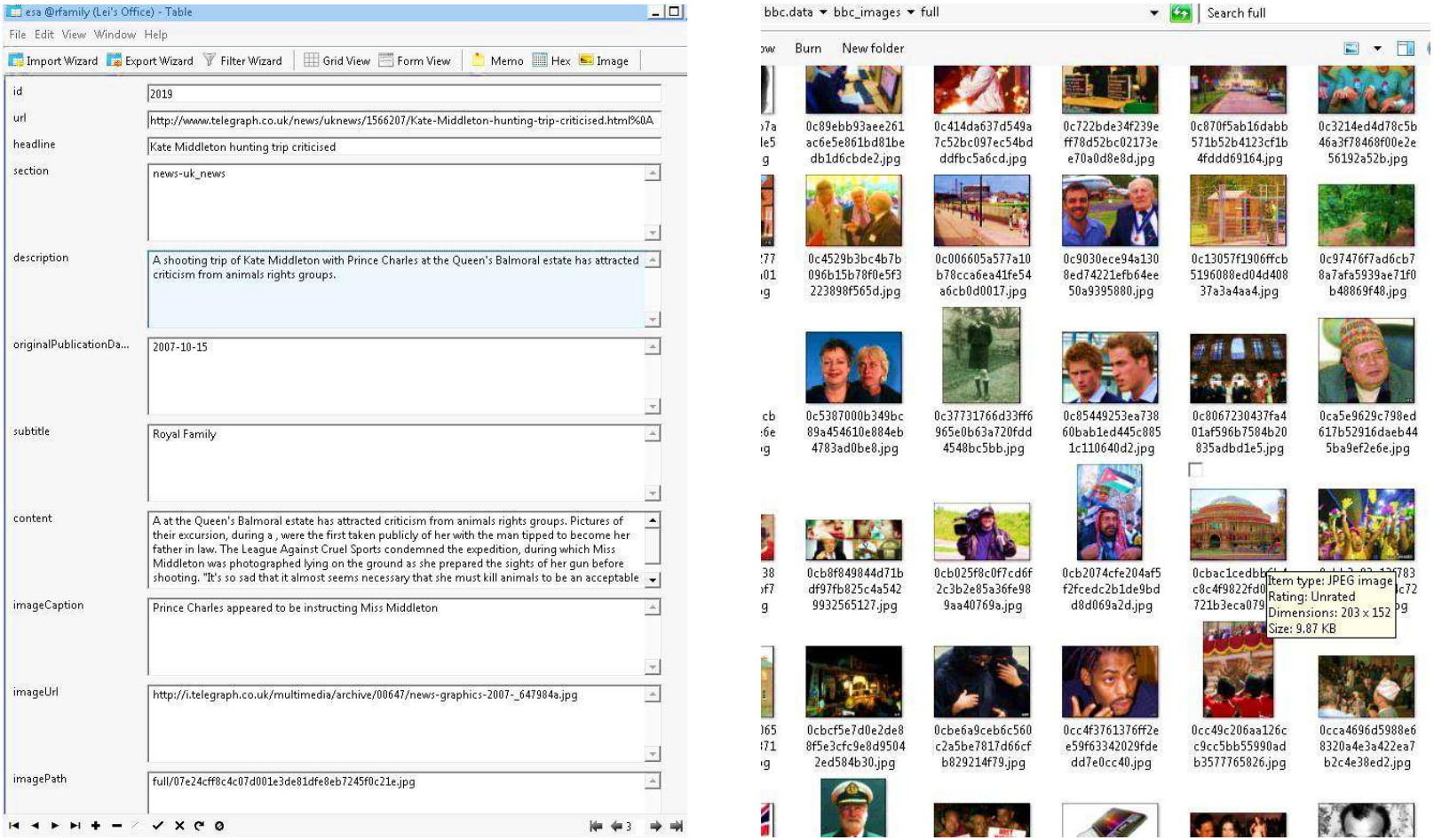


Figure A - 2 Screenshots of stored textual data in a database (left) and images in directory (right)

Appendix B

Result (Samples)

The following table shows part of a vocabulary generated from the raw data, which contains terms (meaningful tokens, without duplications and stop words) with their unique term id.


```
{"term_id": 0, "term": "rail"}
{"term_id": 1, "term": "maritime"}
{"term_id": 2, "term": "transport"}
{"term_id": 3, "term": "union"}
{"term_id": 4, "term": "rmt"}
{"term_id": 5, "term": "cities"}
{"term_id": 6, "term": "railway"}
{"term_id": 7, "term": "cost"}
{"term_id": 8, "term": "travel"}
{"term_id": 9, "term": "cheaper"}
{"term_id": 10, "term": "study"}
{"term_id": 11, "term": "found"}
{"term_id": 12, "term": "card"}
{"term_id": 13, "term": "cover"}
{"term_id": 14, "term": "radius"}
{"term_id": 15, "term": "central"}
{"term_id": 16, "term": "london"}
{"term_id": 17, "term": "compare"}
{"term_id": 18, "term": "tube"}
{"term_id": 19, "term": "spokesman"}
{"term_id": 20, "term": "low"}
{"term_id": 21, "term": "add"}
{"term_id": 22, "term": "older"}
{"term_id": 23, "term": "larger"}
{"term_id": 24, "term": "network"}
```

```
{"term_id": 25, "term": "european"}
{"term_id": 26, "term": "revenue"}
{"term_id": 27, "term": "end"}
{"term_id": 28, "term": "invested"}
{"term_id": 29, "term": "back"}
{"term_id": 30, "term": "system"}
{"term_id": 31, "term": "investing"}
...
```


The raw textual data is converted into normalised vectors, e.g. TF-IDF-based or semantic. The following sample shows the vectors of some information objects. Each entry of a vector is a tuple that indicates the id of a term or named entity with its normalised weight. For example, **`[0, 0.024218724637266462]`** means term **`rail (term_id: 0)`** has weight **`0.024218724637266462`** in a document vector with **`docId: 0`**.


```
{"vector": [[0, 0.024218724637266462], [1, 0.033672137904695733], [2, 0.030192091843680807], [3, 0.014948702394725203], [4,
0.1915616452095085], [5, 0.027955128359064628], [6, 0.02857211025230617], [7, 0.02014648160390211], [8, 0.012864253832048323], [9,
0.03727112570179425], [10, 0.017971694300743375], [11, 0.026582119551978433], [12, 0.047307867001641662], [13, 0.1571213924586968],
[14, 0.038837256020428063], [15, 0.031328290030530496], [16, 0.059973058358046748], [17, 0.010337294565009795], [18,
0.03279117784791679], [19, 0.043474555312090199], [20, 0.011928839539786828], [21, 0.012150747491088789], [22, 0.017598938557186455],
[23, 0.02382249479600471], [24, 0.1019180081570768], [25, 0.033056518183666728], [26, 0.012591141979089721], [27,
0.042238310906726181], [28, 0.026773248280368114], [29, 0.029644215192289984], [30, 0.034028939743548793], [31, 0.026868893493159584],
[32, 0.030352469155313498], [33, 0.024850903618109457], [34, 0.031631598947299923], [35, 0.023495592611512427], [36,
0.010208341068731318], [37, 0.013897566835584095], [38, 0.025883784647409105], [39, 0.010182922365679197], [40, 0.011494811004780534],
[41, 0.035559074384624252], [42, 0.054475425225812744], [43, 0.030192091843680807], [44, 0.0076873701601822943], [45,
0.027493617763168086], [46, 0.021703511748251789], [47, 0.011882056207723623], [48, 0.006896597276111689], [49, 0.012274678967584288],
[50, 0.0053034870962039198], [51, 0.010049375713537406], [52, 0.024718296064037689], [53, 0.016116714826461136], [54,
0.0077191467458047908], [55, 0.05624737930849702], [56, 0.016314403269258863], [57, 0.025885509997332433], [58, 0.020395318728010415],
[59, 0.0091320242453046385], [60, 0.10453967681891603], [61, 0.017032257281754944], [62, 0.010429464483507685], [63,
0.055475774236383377], [64, 0.011517446727314961], [65, 0.011893713412164591], [66, 0.0091842810806711857], [67,
0.010049375713537406], [68, 0.020004318797229775], [69, 0.031878342314426898], [70, 0.017862703632550266], [71,
0.0086013440731356351], [72, 0.0088027262795464297], [73, 0.0079916270915125001], [74, 0.042831463565356859], [75,
0.080134613214305386], [76, 0.011876237199037258], [77, 0.0340031961744457], [78, 0.033320685780525279], [79, 0.038504985625087945],
[80, 0.019819122023277135], [81, 0.02237480370883440 7], [82, 0.011887881607414769], [83, 0.030166343886673913], [84,
```

0.011473490648795085], [85, 0.012163011140218409], [86, 0.056659371527094056], [87, 0.01016183277174852], [88, 0.012927691442703088], [89, 0.010187150365811901], [90, 0.0090052833376113809], [91, 0.014406501064231593], [92, 0.012206158596288147], [93, 0.036997156368760574], [94, 0.0037431966189997615], [95, 0.013441024882566707], [96, 0.012206158596288147], [97, 0.013144803303234649], [98, 0.013599373045567671], [99, 0.034846558939638672], [100, 0.018532568820861843], [101, 0.024986911733852847], [102, 0.020613847563324225], [103, 0.010000150061966893], [104, 0.0090757110636131583], [105, 0.014434689044936199], [106, 0.048584724811225166], [107, 0.0223328744320982], [108, 0.013394361597260981], [109, 0.10595949319375447], [110, 0.013241717224326176], [111, 0.0037231530396041202], [112, 0.03279169977174913], [113, 0.011806902540025067], [114, 0.019950742322644217], [115, 0.018869445568897022], [116, 0.036017055923729328], [117, 0.013821491853403575], [118, 0.010456095631767331], [119, 0.034028939743548793], [120, 0.017204125881723615], [121, 0.012243423568444533], [122, 0.013931731371649308], [123, 0.011976039191732563], [124, 0.01770330521639445], [125, 0.0073005263803065015], [126, 0.01770330521639445], [127, 0.01228723288086651], [128, 0.0091668048675438533], [129, 0.0093046713493162795], [130, 0.013488102344457938], [131, 0.013923169584664111], [132, 0.016073105873380004], [133, 0.010191381738900792], [134, 0.0055411747778610241], [135, 0.018847752181251121], [136, 0.010438326684208717], [137, 0.011609704144429208], [138, 0.0076873701601822943], [139, 0.00994733282223705602], [140, 0.031777643719865654], [141, 0.047971086185256209], [142, 0.05011249278573008], [143, 0.062791651295925469], [144, 0.011319147329050505], [145, 0.0097684614958971384], [146, 0.030807223674908082], [147, 0.0071504937473378157], [148, 0.025967744604054689], [149, 0.010882127661363435], [150, 0.0087444040877987135], [151, 0.081347253913991738], [152, 0.0223328744320982], [153, 0.02580453092994273], [154, 0.032999131217104911], [155, 0.023241763036403417], [156, 0.022417067086914669], [157, 0.036372432098752072], [158, 0.017844754289242377], [159, 0.035308069535535211], [160, 0.027119022801146701], [161, 0.014626537295950421], [162, 0.0098950363528123239], [163, 0.0077671714531447203], [164, 0.013472363383693686], [165, 0.016381971391739945], [166, 0.01847171890812711], [167, 0.035308069535535211], [168, 0.025197695465712964], [169, 0.027277096847529967], [170, 0.010901450181405694], [171, 0.014685471523225968], [172, 0.013056889259321677], [173, 0.044350097545231157], [174, 0.014105935071742978], [175, 0.012408000820476829], [176, 0.013855168321364885], [177, 0.012059662953611995], [178, 0.012433779854871586], [179, 0.013906087360927475], [180, 0.011970116214137027]], **"docId": 0**}

{"vector": [[4, 0.14034954207942441], [21, 0.024481506056119637], [22, 0.053187903195052399], [25, 0.066602762562647036], [39, 0.020516702840479569], [68, 0.04030499787293703], [76, 0.071785255958625213], [94, 0.007541848002725446], [132, 0.048576497750659572], [138, 0.015488627285700624], [139, 0.040084067076811739], [141, 0.04832642756440626 3], [147, 0.028813841470753866], [150, 0.017618354902824075], [168, 0.050768690123510568], [181, 0.19992187463398886], [182, 0.059398758614403135], [183, 0.084626621388536591], [184, 0.024444549373988711], [185, 0.28083745278760652], [186, 0.039035057179238289], [187, 0.084013468289246163], [188, 0.091564199747274602], [189, 0.049336462379509824], [190, 0.036396143798505136], [191, 0.026529089634349069], [192, 0.01742533406372981], [193, 0.033228131142719462], [194, 0.089357233572613895], [195, 0.063731814175300591], [196, 0.024668231189754912], [197, 0.089357233572613895], [198, 0.069349156154747474], [199, 0.13575622715368893], [200, 0.042936592530429446], [201, 0.022201017347000768], [202, 0.01715461261351715], [203, 0.034378976045407877], [204, 0.034889479309665142], [205, 0.027271730230505858], [206, 0.020381320128132721], [207, 0.031717531990949753], [208, 0.017637850128539273], [209, 0.012373129573555576], [210, 0.0664538666168593], [211, 0.035662664868140959], [212, 0.05538715765729732], [213, 0.068562011927742764], [214, 0.0093783016609063682], [215, 0.02446917295615586], [216, 0.095533580565743237], [217, 0.046532346805764568], [218, 0.027065448860482774], [219, 0.020933456085539662], [220, 0.029197726036761809], [221, 0.10088143502768585], [222, 0.016526871750101792], [223, 0.057342951388263071], [224, 0.060066890532711123], [225, 0.06782471064218043], [226, 0.051991351206995724], [227,

0.060417303234773026], [228, 0.025989908745372138], [229, 0.032816528408591984], [230, 0.032843480831283453], [231,
0.059398758614403135], [232, 0.012464929386769652], [233, 0.011643514509181049], [234, 0.046827848488160961], [235,
0.018157626596169596], [236, 0.025517340963290887], [237, 0.057891103843402153], [238, 0.032843480831283453], [239,
0.019509776228065896], [240, 0.01556335926809754], [241, 0.02446917295615586], [242, 0.037589198855783032], [243,
0.039931860669121341], [244, 0.021049180822504229], [245, 0.14908483751559817], [246, 0.033976128721087941], [247,
0.070209363196901631], [248, 0.048676840266468159], [249, 0.020347740744864114], [250, 0.035810428457829599], [251,
0.033144532568300834], [252, 0.040037148515193195]], "docId": 1}

{"vector": [[4, 0.26981755320648831], [5, 0.0481252842637062], [6, 0.02459371515388379], [13, 0.067621865108806217], [14,
0.066859073655420459], [24, 0.058484595398153767], [25, 0.028453711854295412], [26, 0.043351779978638025], [27, 0.048476036146116119],
[31, 0.046255310317338018], [32, 0.05225235196357671], [33, 0.04278130243117576], [34, 0.054454398187756832], [35,
0.040448108799565702], [47, 0.020455185370258387], [48, 0.011872623158875819], [62, 0.017954521136165125], [63, 0.063668399123612981],
[64, 0.0099137516133850304], [66, 0.015810914265459254], [68, 0.034437814638268983], [69, 0.16463751473779967], [74,
0.049156785188932767], [75, 0.11036260655337247], [81, 0.038518649422803537], [95, 0.023138979544671798], [98, 0.011705789456944326],
[104, 0.0078120044598189208], [106, 0.027879842085766337], [109, 0.045602819855539899], [111, 0.019228435951373177], [114,
0.034345581719995105], [115, 0.032484108827468293], [116, 0.031002022820425242], [117, 0.023793960659023877], [118,
0.090001835817744125], [119, 0.058581465887628303], [120, 0.029617229366005207], [123, 0.1030848943085841], [131,
0.023969000803978721], [133, 0.01754465717076592], [150, 0.015053657670134495], [209, 0.01057197780018989], [211,
0.010157088095356601], [235, 0.015514427787866427], [242, 0.032117353452726011], [249, 0.0173857278516244], [253,
0.014181365198434862], [254, 0.018969223682563947], [255, 0.052252351963577671], [256, 0.22904860504372548], [257,
0.025110655044000767], [258, 0.024360173490519683], [259, 0.065971744874442018], [260, 0.045546441413067809], [261,
0.031904169109331783], [262, 0.019423247439519099], [263, 0.027590651638343119], [264, 0.031593444112318049], [265,
0.038317125566447986], [266, 0.02312554174788085], [267, 0.022782925057162631], [268, 0.048770091947372099], [269,
0.066206791368837142], [270, 0.032984034752448217], [271, 0.049957604625699772], [272, 0.015341049516776227], [273,
0.019504172470885199], [274, 0.027546962649389398], [275, 0.067982901502629839], [276, 0.048770091947372099], [277,
0.012903667140900183], [278, 0.034208956009816965], [279, 0.033897894938113883], [280, 0.02467290889637461], [281,
0.047920085037805066], [282, 0.060783511211180745], [283, 0.020159339364394555], [284, 0.030597517986120225], [285,
0.019769919734102422], [286, 0.010314184121217604], [287, 0.013224868139224836], [288, 0.018272489767059405], [289,
0.018186088439385181], [290, 0.03713852551085961], [291, 0.021174367779977717], [292, 0.029853902442281994], [293,
0.057362193242423261], [294, 0.010314184121217604], [295, 0.038591406630637912], [296, 0.030094224697237146], [297,
0.033682022442416006], [298, 0.042966348709324413], [299, 0.009376931479270456 5], [300, 0.018364794166635515], [301,
0.047920085037805066], [302, 0.02467290889637461], [303, 0.023679104034275401], [304, 0.034437814638268983], [305,
0.038591406630637912], [306, 0.022744207057363033], [307, 0.019011911844800323], [308, 0.021163529818739202], [309,
0.031162501138633 43], [310, 0.019219639808753859], [311, 0.018209571473172363], [312, 0.021630177847431431], [313,
0.035783769595982111], [314, 0.019986326122055347], [315, 0.017215448207943004]], "docId": 2}

{"vector": [[4, 0.28707860879882263], [6, 0.19625287850068884], [15, 0.028691228580149143], [25, 0.045410974474532073], [49,
0.033724370496797239], [84, 0.03152312582295215], [120, 0.047267901412412355], [150, 0.024025029412941924], [186,
0.053229623426234031], [196, 0.033638497076938521], [214, 0.0063942965869816151], [260, 0.024230093411665703], [286,
0.016461021122751329], [292, 0.023822811039800783], [299, 0.014965203774997295], [316, 0.099932338695561257], [317,

0.061359008540714251], [318, 0.032297911731829471], [319, 0.13904334613368854], [320, 0.033183327300681999], [321,
0.013613065341174678], [322, 0.06670499912793515], [323, 0.036950354234005375], [324, 0.020241233045236881], [325,
0.036135325490918906], [326, 0.020461409882229269], [327, 0.058475846750922142], [328, 0.11594206092506788], [329,
0.028569788900864167], [330, 0.1534311070394811], [331, 0.10185011657385845], [332, 0.038206623860325993], [333,
0.083897144951856045], [334, 0.030410691732310331], [335, 0.03500175790041693], [336, 0.050660176982945691], [337,
0.025953243263253063], [338, 0.051665245461799986], [339, 0.05064018164497288], [340, 0.062185211481761191], [341,
0.013630408606751288], [342, 0.028545621934826817], [343, 0.030085031762611257], [344, 0.030355907167108082], [345,
0.034593844629418224], [346, 0.042105139080756901], [347, 0.013734903901322513], [348, 0.027599087416043021], [349,
0.025983592480430084], [350, 0.043417726682492781], [351, 0.046502978925914935], [352, 0.070039833488463751], [353,
0.035058277727624203], [354, 0.057195425212330231], [355, 0.1602816871789477], [356, 0.036237304053903824], [357,
0.045619515882982221], [358, 0.04486035063481314], [359, 0.091547742750534108], [360, 0.020656831853430816], [361,
0.086907019329955357], [362, 0.10164875285154421], [363, 0.016678646643951765], [364, 0.098396953913288024], [365,
0.036568515519738866], [366, 0.043859872157532238], [367, 0.028509446466069895], [368, 0.043152158017655123], [369,
0.029074254438329214], [370, 0.07253114629233369]], "docId": 3}

{"vector": [[2, 0.0220759381222612361], [4, 0.43293306326919218], [22, 0.045038143834520178], [24, 0.099360925515142973], [25,
0.0060425893454014454], [27, 0.0102946420847934791], [37, 0.010161661772255038], [39, 0.0074455776437224244], [48,
0.0050426732771569347], [61, 0.012453693496336949], [64, 0.012632038346087377], [73, 0.005843340238955377], [82,
0.0086922145086473588], [84, 0.00419460948450573], [87, 0.014860314590944073], [98, 0.0049718138016053856], [100,
0.013550695481920489], [102, 0.015072490691462876], [104, 0.0033180018942241657], [107, 0.016329413563254 6], [108,
0.019587453518575198], [109, 0.019368939616062647], [112, 0.055945695667930416], [116, 0.065837629107892315], [117,
0.010106037054101541], [118, 0.0076453172361309529], [119, 0.17416962707450781], [120, 0.025158721719509804], [121,
0.00895218067370138], [122, 0.010186642293248957], [123, 0.017513347635221815], [127, 0.0089842132892357287], [137,
0.0084888159335611418], [139, 0.014546637245617165], [147, 0.01045663601761229], [152, 0.016329413563254 6], [177,
0.017635636147217543], [178, 0.0090913659153899779], [186, 0.0070829740849424321], [205, 0.0098969988739739008], [208,
0.019202498123812917], [211, 0.0043140320405009227], [214, 0.001701707962664462], [219, 0.022790456222160117], [221,
0.024406798797020769], [239, 0.007080160727927139], [246, 0.018495070069947061], [250, 0.025991440009715032], [259,
0.009340085743872616], [272, 0.0065158221065877535], [280, 0.020958707557135421], [288, 0.0077608961913854463], [290,
0.007886944933757821], [291, 0.0089934142721410752], [292, 0.019019824943066754], [298, 0.0091245740538619061], [299,
0.0079653503963695287], [326, 0.0054453752105932725], [333, 0.0074424886650840041], [336, 0.020223215811740414], [337,
0.04834838059525369], [341, 0.0036274474517967135], [344, 0.0080785881976981177], [371, 0.0059427339325233341], [372,
0.020677804614097201], [373, 0.029996832016336666], [374, 0.012375792778670911], [375, 0.013691083779533689], [376,
0.01365654020862518], [377, 0.0064579012419363176], [378, 0.014128687124315248], [379, 0.0074890836886358043], [380,
0.0093755034555600691], [381, 0.043304351409706052], [382, 0.010616893457671205], [383, 0.011363376820550556], [384,
0.015108287732403624], [385, 0.0097937267592875989], [386, 0.0067903444402265586], [387, 0.10208123409933001], [388,
0.09728408493792641], [389, 0.017886654828401006], [390, 0.01930264377134687], [391, 0.015923858010494847], [392,
0.0095310675918523231], [393, 0.013318952057054327], [394, 0.0092553943731058866], [395, 0.042863174860592067], [396,
0.011672385332246482], [397, 0.038575572025661149], [398, 0.049293536538853343], [399, 0.027049538062520421], [400,
0.077449958981174016], [401, 0.041569998270890578], [402, 0.029214354366791758], [403, 0.040045788646170262], [404,

```
0.030770841556012108], [405, 0.0094287644003320965], [406, 0.027051684226620635], [407, 0.028167376428983121], [408,
0.012119993765121611], [409, 0.015153123058066608], [410, 0.010021064993572448], [411, 0.011115373018414377], [412,
0.020102860299950859], [413, 0.0073933967124153784], [414, 0.0084606106275997399], [415, 0.014726516213929355], [416,
0.012995720004857516], [417, 0.0057925642783041003], [418, 0.012476241306801245], [419, 0.024884937121861031], [420,
0.033844312943687907], [421, 0.017179573092288659], [422, 0.012078954096698732], [423, 0.013074070396415354], [424,
0.0096437986595066631], [425, 0.017950068167650143], [426, 0.017295349226406631], [427, 0.01266150458818493], [428,
0.028874458165095471], [429, 0.014587639547739858], [430, 0.020714178837862345], [431, 0.011061449733113436], [432,
0.025168781758499424], [433, 0.022492705820914477], [434, 0.01368731983427985], [435, 0.014568199845505784], [436,
0.01008156875542613], [437, 0.010513284112893265], [438, 0.012818832751105508], [439, 0.025991440009715032], [440,
0.008862090386301107], [441, 0.051312796888401485], [442, 0.042437105004626691], [443, 0.010009221439799624], [444,
0.0057925642783041003], [445, 0.022476255284629224], [446, 0.0061428747386223599], [447, 0.0090026371361203068], [448,
0.070232400711305792], [449, 0.0090397495004349033], [450, 0.012277880390902435], [451, 0.020620723412699236], [452,
0.027985191060082282], [453, 0.0093300255242870864], [454, 0.017708314788538921], [455, 0.010372729537778869], [456,
0.019718564725106666], [457, 0.011053607035849204], [458, 0.011124768894347029], [459, 0.015531785788818891], [460,
0.0093602848128438686], [461, 0.015923858010494847], [462, 0.017179573092288659], [463, 0.017978997583214924], [464,
0.020798224074627034], [465, 0.025168781758499424], [466, 0.025479204385972367], [467, 0.016886738760156499], [468,
0.0089068739379194409], [469, 0.011329579548047147], [470, 0.0080713188188583918], [471, 0.011812746096872658], [472,
0.0077408364269614446], [473, 0.0087264594329233731], [474, 0.02111229345880088], [475, 0.0099674070856937205], [476,
0.0051814979736518191], [477, 0.0091293413715802362], [478, 0.0052091748973644124], [479, 0.0090913659153899779], [480,
0.012442468560930516], [481, 0.029739856269631389], [482, 0.040706308795554848], [483, 0.018219957831829645], [484,
0.0088354663974681263], [485, 0.016499317717440957], [486, 0.012277880390902435], [487, 0.009100824974897493], [488,
0.041058934288176231], [489, 0.011371872360586384], [490, 0.020022894323085131], [491, 0.012119993765121611], [492,
0.013476822534549234], [493, 0.0065993863180778692], [494, 0.021925634238425695], [495, 0.011968284233858935], [496,
0.03370328315028371], [497, 0.01014303141228564], [498, 0.01930264377134687], [499, 0.014491401049277906], [500,
0.019610133139853686], [501, 0.0085581808674464882], [502, 0.024881375296358262], [503, 0.014163469532131103], [504,
0.0058535756961383906], [505, 0.024881375296358262], [506, 0.025168781758499424], [507, 0.0089804972711290485], [508,
0.025252173752994947], [509, 0.016647477991181377], [510, 0.0066343254055210767], [511, 0.015198482785390254], [512,
0.011423240164209246], [513, 0.024363512183609885], [514, 0.011092980089516952], [515, 0.006950183510373624], [516,
0.0061561487246859807], [517, 0.016816859167326], [518, 0.011617633187907875]], "docId": 4}
```


To detect the semantic relations between the information objects, dimensionality reduction and ontology-based semantic feature matching are applied. The sample below shows the projections of 2065 vectors in a semantic space with low dimensionality equal to 80. In the next table, each **row** indicates an information object vector in that semantic space, and each entry of the vector indicates the coordinate on its related dimension.

	d1	d2	d3	d4	d5	d6	d7	d8	d9	d10	d11	d12	d13	d14	d15	d16	d17	d18	d19	d20	...	d80
1	0.000853	-0.05569	-0.00076	-0.00518	0.047682	-0.00248	0.012289	-0.00093	-0.00896	-0.00807	-0.0057	0.006143	0.003445	0.027113	-0.00879	-0.01576	0.035699	-0.01112	-0.01984	-0.01186	...	0.087057
2	0.020445	-0.06208	-0.01532	0.009284	0.037258	-0.07587	0.012509	-0.09324	-0.00249	-0.00792	0.017492	0.008785	0.057816	0.027008	-0.0061	0.007123	-0.00176	-0.02291	-0.00313	-0.00964	...	-0.04868
3	0.000852	-0.04654	-0.0004	-0.00661	0.048932	-0.00141	0.007187	-5.49E-05	-0.00313	-0.00985	-0.00559	0.010282	0.000432	0.008181	-0.00315	-0.02343	0.040498	-0.03284	0.015066	0.001747	...	-0.01231
4	0.000866	-0.05278	-0.00031	-0.00471	0.03943	-0.00197	0.009094	0.000316	-0.02544	-0.00112	-0.00127	0.000873	0.002215	0.007931	-0.00127	-0.00498	0.006602	0.007519	-0.00137	-0.00245	...	0.008448
5	0.001086	-0.05205	-3.93E-05	-0.0101	0.060459	-0.00086	0.013581	-0.00071	0.010084	-0.01941	0.015142	-0.02076	-0.00603	-0.00142	-0.00518	-0.01801	0.016572	-0.00926	-0.03215	-0.01279	...	-0.03502
6	0.001116	-0.06843	-0.00033	-0.01298	0.085573	-0.00122	-0.0388	-0.00071	0.017048	-0.01506	-0.00933	0.007976	1.47E-05	0.022003	-0.00474	-0.01854	0.026203	0.027156	0.020279	0.006303	...	0.13423
7	0.00068	-0.04869	8.71E-05	-0.00249	0.022673	-0.00037	-0.02591	-0.00042	-0.00546	-0.00224	0.002152	-0.01294	0.0057	-0.01452	0.000716	-0.01232	0.008336	0.002858	0.003721	2.35E-05	...	0.002046
8	0.000679	-0.05209	-0.00013	-0.00302	0.026474	-0.00146	0.006197	-0.00068	-0.00978	-0.00941	-0.00369	0.004281	0.001533	0.020989	-0.007	-0.00395	0.015766	-0.00521	-0.00438	-0.00113	...	-0.02151
9	0.003156	-0.04014	4.08E-05	-0.01715	0.055538	0.000139	0.012741	0.000154	0.015439	-0.01229	0.006007	-0.02022	0.009243	-0.01974	0.002098	-0.00227	-0.02788	-0.01948	-0.02334	-0.0033	...	-0.0167
10	0.00296	-0.06564	0.000178	-0.01673	0.068031	-0.00103	0.014911	-0.00088	0.006828	-0.02088	0.005719	-0.03269	0.015241	0.001561	-0.00318	0.010227	-0.03173	-0.02335	-0.01085	0.00205	...	-0.00647
11	0.001284	-0.05404	-0.00068	-0.00915	0.06266	-0.00361	0.008855	-0.00469	0.00523	-0.01336	-0.00063	-0.0141	0.011842	-0.00024	-0.00289	-0.00877	0.002947	0.017947	0.014735	0.000512	...	0.003376
12	0.000517	-0.0361	7.55E-05	-0.00333	0.022925	-0.00027	0.00514	0.000376	-0.00018	-0.0068	0.000609	-0.01041	0.004769	-0.00847	-0.00075	-0.00078	-0.00368	0.001268	0.000571	-0.00216	...	0.005366
13	0.000843	-0.06966	0.000282	0.000834	0.007497	-0.00052	0.005574	0.002653	-0.05137	-0.01104	0.006782	-0.03394	0.012968	0.008581	-0.0099	-0.00852	0.021061	0.001006	-7.34E-05	-0.00178	...	-0.00401
14	0.001135	-0.08583	0.000175	-0.00045	0.014232	-0.00079	0.004775	-0.00023	-0.00659	-0.01394	0.000283	-0.01656	0.008336	0.006465	-0.00687	-0.0057	0.010622	0.009178	0.002541	0.00092	...	0.024859
15	0.000841	-0.05297	-0.00051	-0.00463	0.037239	-0.00083	0.005513	0.000235	0.001524	-0.01495	-0.00178	-0.01456	0.007595	0.016543	-0.00792	-0.00973	0.019607	-0.0088	0.001778	-0.00319	...	0.020336
16	0.001083	-0.07041	0.000159	-0.00753	0.040858	-0.00039	0.008751	0.001188	-0.00981	-0.01039	-0.00285	-0.01044	0.00449	0.010495	-0.0069	-0.00382	0.007922	0.0032	-0.00161	-0.00087	...	-0.00783
17	0.000896	-0.05764	-0.00025	-0.00446	0.035227	-0.00075	0.005793	0.000776	-0.00879	-0.01201	0.001279	-0.02084	0.009226	0.007472	-0.00609	0.001383	0.002198	0.013951	0.012514	0.001585	...	0.014542
18	0.017941	-0.08217	-0.01269	0.007251	0.040981	-0.05989	0.011563	-0.06807	-0.03107	-0.00613	0.011577	0.010623	0.040485	0.009985	-0.00777	-0.00668	0.015986	0.01074	0.002571	-0.00468	...	0.021683
19	0.000707	-0.04381	-0.00019	-0.00193	0.018664	-0.00094	0.003979	-0.00085	0.002402	-0.00651	0.002241	-0.01957	0.009664	-0.00748	-0.00147	-0.00677	0.006692	0.005624	0.001287	-0.0007	...	0.010522
20	0.000723	-0.04114	-5.11E-05	-0.00644	0.039638	-0.0002	0.009672	-0.00022	0.024557	-0.01542	0.003368	-0.04415	0.01827	-0.01814	0.005694	0.047727	0.008843	-0.01699	-0.01455	-0.00508	...	0.025868
21	0.000744	-0.04633	-0.00016	-0.00585	0.040636	-0.00015	0.007135	-8.52E-05	0.01005	-0.00645	1.49E-05	-0.01262	0.005577	-0.00793	-0.00111	-0.00054	-0.01222	0.00412	0.00265	-0.00303	...	0.013521
22	0.001003	-0.07963	-0.00021	0.001756	0.005969	-0.00095	0.007621	0.000799	-0.01764	0.036198	0.00127	0.004004	-0.00073	-0.02113	0.000837	-0.01617	0.006244	0.000424	-0.00979	-0.00478	...	0.035625
23	0.001258	-0.05783	-0.00078	-0.00916	0.064066	-0.00108	-0.01449	-0.00035	-0.00022	-0.01146	-0.00036	-0.01416	0.008005	-0.00235	0.002407	0.000589	0.02811	-0.02912	-0.00179	-0.0043	...	0.004535
24	0.00135	-0.05664	-0.00029	-0.00761	0.051847	-0.00063	0.009629	-0.00165	-0.00119	-0.01811	0.000553	-0.02486	0.012617	0.013164	-0.00691	-0.00432	0.007535	-0.02306	-0.01436	-0.00942	...	-0.00733
25	0.000794	-0.04388	1.24E-05	-0.00667	0.045917	-8.11E-05	0.011131	-0.00079	0.013561	-0.00814	0.001994	-0.02618	0.010545	-0.00263	-0.00186	-0.00239	-0.00601	-0.02101	-0.02297	-0.01229	...	-0.00298
26	0.000611	-0.04283	-5.53E-06	-0.00215	0.019261	-0.00025	0.004061	0.000375	-0.00198	-0.00678	-0.00309	0.007722	-0.00107	-0.00747	0.003383	-0.01135	0.006343	-0.03687	0.012298	0.003574	...	-0.04668
27	0.000819	-0.06333	0.000172	-0.00076	0.016633	-0.00123	-0.02298	5.98E-05	-0.03629	0.000367	0.006845	-0.0337	0.014042	-0.01168	-0.00623	-0.01157	0.00997	-0.00069	-0.00985	-0.00521	...	0.03458
28	0.000673	-0.0377	-1.80E-06	-0.00502	0.034549	-1.08E-05	0.007597	-0.00166	0.006396	-0.00758	0.000396	-0.01245	0.007061	-0.00265	-0.00144	-0.00602	-0.00166	-0.01774	-0.00431	-0.00644	...	-0.01589
29	0.000785	-0.04835	-0.0003	-0.00431	0.033292	-0.00041	0.007977	0.000197	0.001707	-0.00492	0.002305	-0.02118	0.008673	-0.00236	-0.00134	0.004468	-0.0149	-0.00607	-0.00156	-0.0053	...	0.003851
30	0.001313	-0.03504	9.87E-06	-0.00867	0.041779	-0.00013	0.008085	0.000104	0.010786	0.00418	0.003131	-0.01856	0.00808	-0.01197	0.002233	0.010451	-0.00672	-0.00319	0.000129	-0.00076	...	0.006009
31	0.000806	-0.05372	0.000191	-0.00112	0.01694	-0.00123	0.005741	0.000116	-0.02487	-0.00747	0.007324	-0.0347	0.014402	-0.0194	-0.00474	-0.01967	0.021589	0.026406	0.012733	0.00077	...	0.004893
32	0.000739	-0.04572	-4.14E-05	-0.00575	0.038904	-0.00044	0.000995	-0.00015	0.006047	-0.00747	0.002446	-0.02359	0.010074	-0.00646	-0.00153	0.001064	-0.01208	-0.00413	-0.00417	-0.0044	...	0.010884
33	0.002696	-0.0564	0.00024	-0.00806	0.017552	-0.0004	0.006914	0.00047	-0.01967	-0.00543	0.004067	-0.01132	0.00843	0.004803	-0.00379	-0.00504	0.006671	-0.00625	-0.00727	0.000813	...	-0.02221
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
2065	0.000817	-0.03415	-0.00062	-0.00412	0.030049	-0.00287	0.006209	-0.00358	0.004493	-0.00383	0.002457	-0.01529	0.008519	-0.00966	0.000512	0.004925	-0.01731	-0.0016	0.00107	-0.00217	...	0.009242

Using the projected vectors, the performance of topic identification and clustering is improved. Two results of topic identification and clustering are visualised in Figures A – 3 and A – 4, respectively.

Figure A – 3 (without ontology) clearly shows the "Royal family" cluster (with pink colour) has plenty of overlaps with the "Celebrity" and "Politician" clusters (see data points inside the polygon). It means that information objects under those topics are difficult to be distinguished on word level, as the topics are correlated and the information objects have lots of common words.

Figure A – 4 (with ontology) shows the clustering result after using the "Royal family" ontology, semantic feature matching and dimensionality reduction. The local clustering performance has a clear improvement, and the "Royal family" cluster (with purple colour) has fewer overlaps with other clusters. Based on the proposed algorithms, selected semantic features from the ontology enhance the semantic representation of the "Royal family" related information objects, which enables them to be distinguished more easily than before. On the other hand, the incorrectly classified information objects could be removed from the rest of clusters, e.g. information objects related to "Royal family" can be removed from the clusters related to "Celebrity" and "Politician", which improves the global clustering performance.

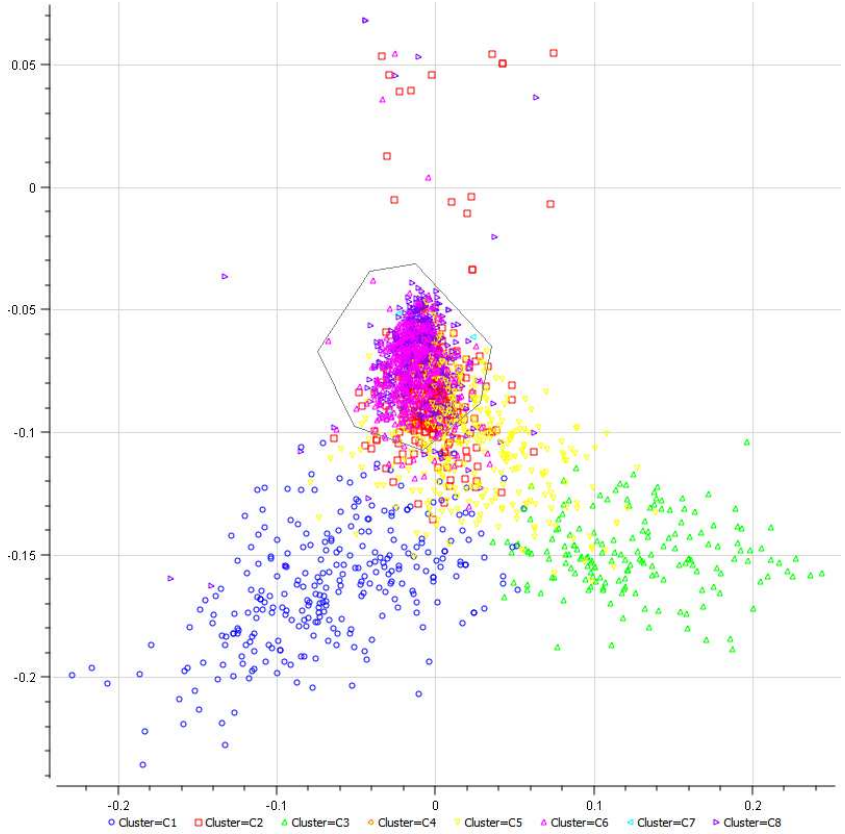


Figure A - 3 A projection of clustering result based on original vectors (without ontology)

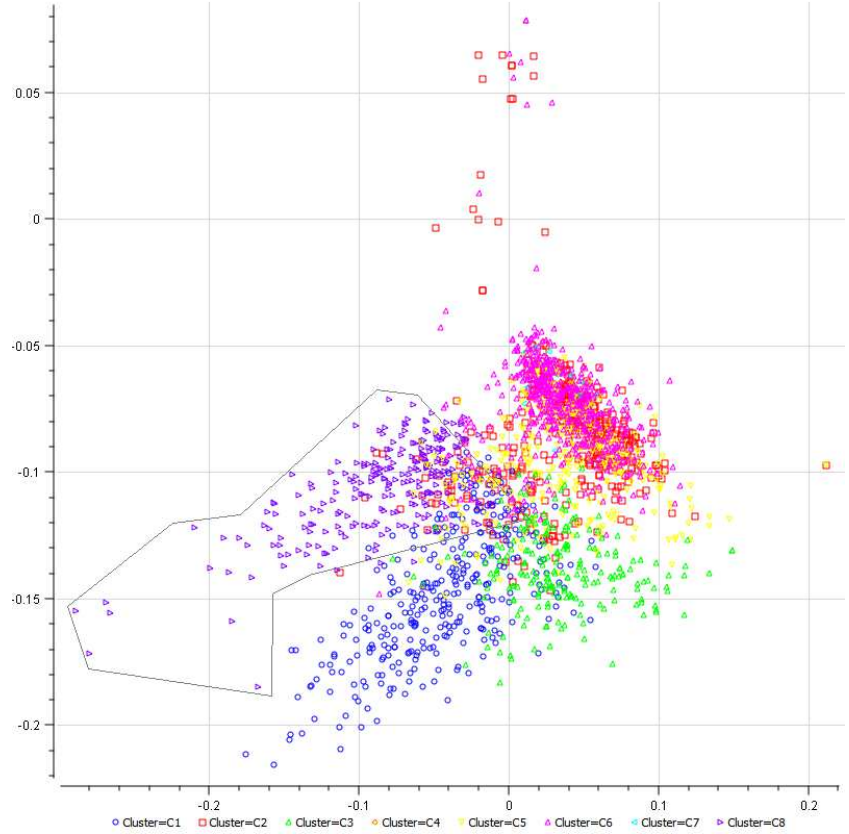


Figure A - 4 A projection of clustering result based on semantic feature based vectors with dimensionality=100 and threshold=4 (with ontology)

Appendix C

Source Code (Samples)

TelegraphSpider.py – to generate the URLs list

```
class TelegraphSpider(CrawlSpider):

    START_PAGE = 1
    END_PAGE = 500
    QUERY = "window phone 7"

    name = "telegraph.co.uk"
    allowed_domains = ["http://www.telegraph.co.uk", ]

    returned_urls = []
    start_urls = []

    for i in range(START_PAGE, END_PAGE):
      urls = "http://www.telegraph.co.uk/search/?queryText=" + QUERY + "&startIndex=" + str((i - 1)*10) +
"&site=telegraph_news&sort=date%3AD%3AL%3Ad1&type=relevant"
      returned_urls.append(urls)
      start_urls.append(urls)
    start_urls = returned_urls
    rules = (
    #Rule(SgmlLinkExtractor(allow=( )), callback=''),
    )

    def parse(self, response):
        hxs = HtmlXPathSelector(response)
```

```
        sites = hxs.select('//div[@class="summary"]/a/@href').extract()
        heads = hxs.select('//div[@class="summary"]/a/h3/text()').extract()
        items = []

        for index, site in enumerate(sites):
            item = ContentbasketItem()
            url = site
            item['url']= site
            item['headline'] = heads[index]
            items.append(item)
        return items
```

ContentGlue.py – to dump crawled content to database

```
class ContentGlue(object):

    def __init__(self,):
        '''
        Constructor
        '''
        self.file = open("tele.json").readlines()

    def urlConvertor(self,):
        # get unique urls
        urls = []
        for line in self.file:
            jsonData = json.loads(line)
            urls.append(jsonData["url"])
        tested_urls = self.urlTester(urls)
        uniques = []
        uniqueUrls = open("converted_tele_2012", 'w')

        for url in tested_urls:
```

```
            if not url in uniques:
                uniques.append(url)
                uniqueUrls.write(url + '\n')
        print "done, removed: %d" % (len(tested_urls) - len(uniques))

    def dbDumper(self):
        lines = open(' tele _items.json').readlines()
        db = DBConnector('localhost', 'root', '123456', 'kes2012')
        i = 0
        for line in lines:
            jsonData = json.loads(line)
            url = jsonData['url']
            headline = re.sub(r'\s+', ' ', ' '.join(jsonData['headline'])).replace("'", "\\'")
            section = re.sub(r'\s+', ' ', ' '.join(jsonData['section'])).replace("'", "\\'")
            description = re.sub(r'\s+', ' ', ' '.join(jsonData['description'])).replace("'", "\\'")
            originalPublicationDate = re.sub(r'\s+', ' ', ' '.join(jsonData['originalPublicationDate'])).replace("'",
"\\'")
            subtitle = re.sub(r'\s+', ' ', ' '.join(jsonData['subtitle'])).replace("'", "\\'")
            content = re.sub(r'\s+', ' ', ' '.join(jsonData['content'])).replace("'", "\\'")
            imageCaption = '\n'.join(jsonData['imageCaption']).replace("'", "\\'")

            images = jsonData['images']
            image_urls = ""
            image_paths = ""
            for image in images:
                image_urls += re.sub(r'\s+', ' ', image['url']).replace("'", "\\'") + '\n'
                image_paths += re.sub(r'\s+', ' ', image['path']).replace("'", "\\'") + '\n'
            # + "'" + headline + "'," +
            sql = "insert tele(url, headline, section, description, originalPublicationDate, subtitle, content,
imageCaption, imageUrl, imagePath) values ('" + url + "'," + "'" + headline + "'," + "'" + section + "'," + "'" +
description + "'," + "'" + originalPublicationDate + "'," + "'" + subtitle + "'," + "'" + content + "'," + "'" +
imageCaption + "'," +"'" + image_urls + "'," +"'" + image_paths + "')"

            if len(content) >= 50:
                db.dbQuery(sql.encode('utf-8'))
```

```
    def urlTester(self, urls):
        item = []
        for url in urls:
            prefix = url.split(':')
            suffix = url.split('.')
            if prefix[0] == "http" and suffix[-1] == "html" :
                checked_url = url
                item.append(checked_url)
        return item
```

OntoSVDModel.py – to execute the Onto-SVD algorithm (the source code only represents part of the mechanism)

```
class OntoSVDModel(object):

    def __init__(self,):
        '''
        Constructor
        '''
        self.mu = MathUtil()

        file_dict = open("dict\\dict")
        self.dictionary = pickle.load(file_dict)

        file_cTF-IDF = open("corpus\\corpus")
        self.corpusTF-IDF = pickle.load(file_cTF-IDF)

        self.rowNum = len(self.dictionary)
        self.colNum = len(self.corpusTF-IDF)

    def vectors2matrix(self):
        termDocMartix = csc_matrix((self.rowNum, self.colNum), dtype=np.float)
        return termDocMartix

    def entityDetector(self):
```

```
        entity_list = open("entities_p_esa").readlines()
        tc = TextCleaner()
        entity_list = tc.stemming(entity_list)

        entity_term_id_list = [] #get entity's term id
        for term_id, term in self.dictionary.items():
            for item in entity_list:
                if term == item[0]:
                    entity_term_id_list.append(term_id)
               docId_entity_id_list = []

        for docId, vector in enumerate(self.corpusTF-IDF):
            for item in vector:
                print "term_id %s, TF-IDF %s" % (item[0], item[1])
                if item[0] in entity_term_id_list: #term_id:item[0], TF-IDF:item[1]
                    docId_entity_id_list.append((docId, item[0]))
        return (entity_term_id_list, docId_entity_id_list) #[(5, 12), (6, 12), (7, 13), (13, 13), (15, 12)]

    def entityMatrixBooster(self, boost_level=2):
        file_M = open("indexs\\m", 'rb')
        M_initial = pickle.load(file_M)

        M_onto = self.vectors2matrix()
        entity_term_id_list = self.entityDetector()[0]
        docId_entity_id_list = self.entityDetector()[1]
        neighbours_id_list = random.sample(entity_term_id_list, boost_level)

        for docId, term_id in docId_entity_id_list:
            for neighbour_id in neighbours_id_list:
                M_onto[neighbour_id,docId] = 0.25
        return np.add(M_initial, M_onto)

    def getTermCoordinates(self, termEigenvectors, k=2):
        if termEigenvectors:
            termCoordinates = []
```

```
            for i in xrange(self.rowNum):
                for j in xrange(k):
                    termCoordinates.append(termEigenvectors[j][i])
            tcs = self.chunks(termCoordinates, k)

            file = open("sem_booster\\term_index", 'wb', True)
            pickle.dump(tcs, file)
            file.close()
            return tcs
        else:
            raise "Before get coordinates, doSVD() needs to be executed."

    def getDocCoordinates(self, docEigenvectors, k=2):
        if docEigenvectors:
            docCoordinates = []
            for i in xrange(self.colNum):
                for j in xrange(k):
                    docCoordinates.append(docEigenvectors[j][i])
            dcs = self.chunks(docCoordinates, k)

            file = open("sem_booster\\doc_index", 'wb', True)
            pickle.dump(dcs, file)
            file.close()
            return dcs
        else:
            raise "Before get coordinates, doSVD() needs to be executed."

    def getQueryCoordinate(self, query_vector, k=2):
        try:
            file_s = open("sem_booster\\s_k", 'rb')
            eigenvalues = pickle.load(file_s)
            file_is  = open("sem_booster\\is_k", 'rb')
            eigenvalues_inv = pickle.load(file_is)
            file_t = open("sem_booster\\u_k", 'rb')
            Uk = pickle.load(file_t)
```

```
        except IOError:
            raise "Before get coordinates, doSVD() needs to be executed."
        query_vector = np.array(query_vector).transpose()
        qcs = []

        for i in xrange(k):
            query_lsi = np.dot(np.dot(np.dot(query_vector, Uk[:,i]), eigenvalues_inv[i]), eigenvalues[i])
            qcs.append(query_lsi)

        file_s.close()
        file_is.close()
        file_t.close()
        return qcs

    def chunks(self, list, size):
        if len(list) % size == 0:
            result = []
            for i in xrange(0, len(list), size):
                #yield list[i:i + size]
                result.append(list[i:i + size])
            return result
        else:
            raise "Check SVD related processes and its k setting."
```

Appendix D

Crowdsourcing Evaluation

Experiment I

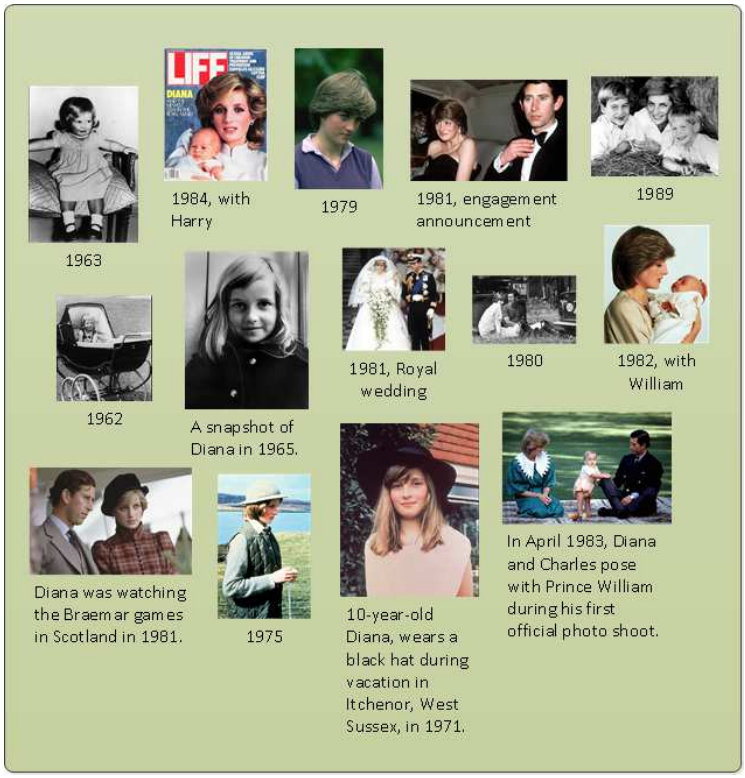


Figure D - 1 Experiment I: Unstructured presentation (Scenario I)

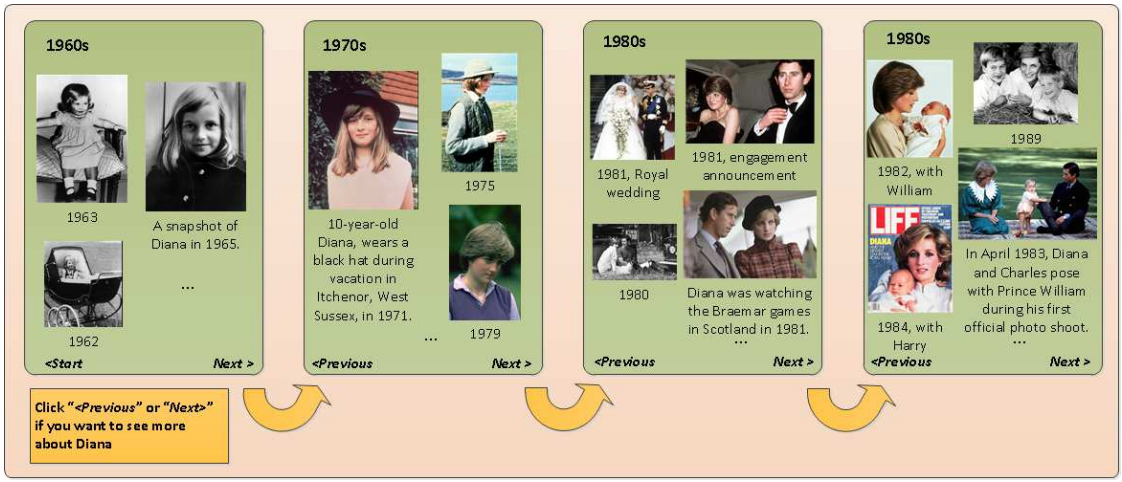


Figure D - 2 Experiment I: LSB-based presentation (Scenario II)

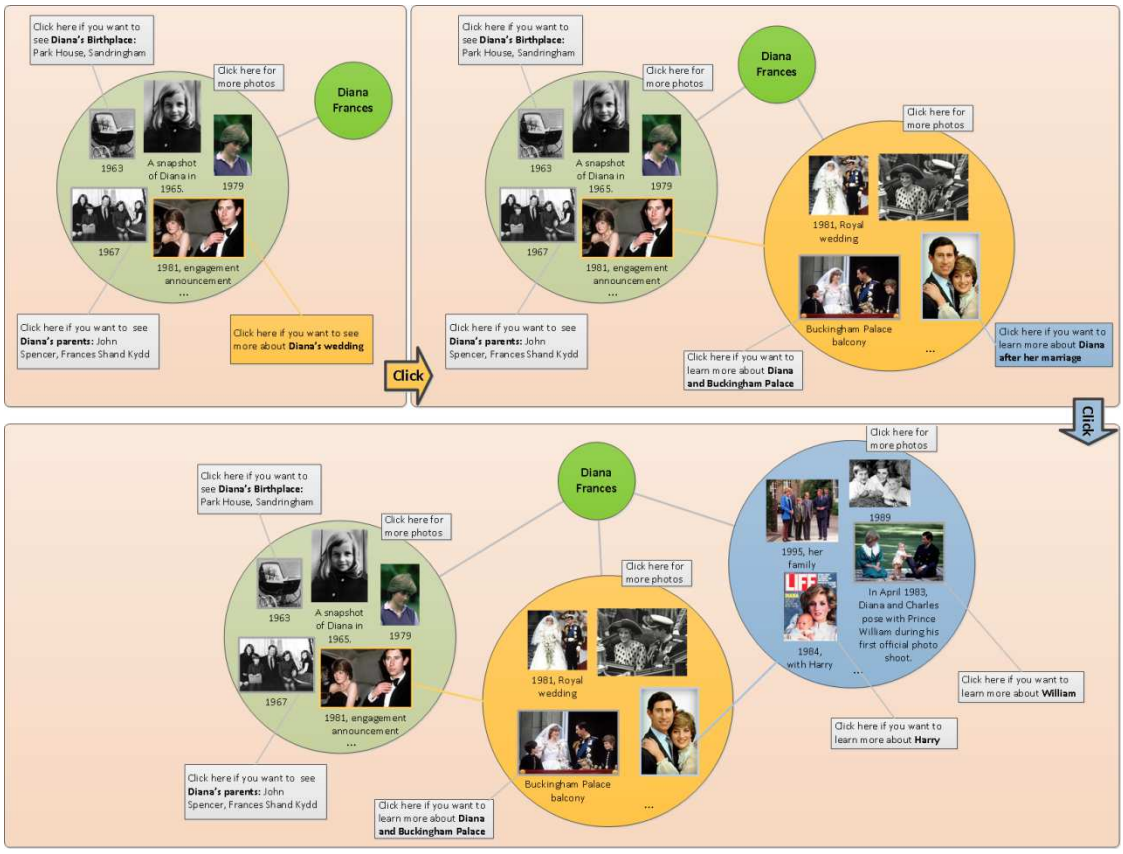


Figure D - 3 Experiment I: Semantic/knowledge-based presentation (Scenario III)

Section A: Amazon Worker ID and email address

A1. Please provide your Amazon Worker ID:

Section B: Scenario I - A

Assume you are a member of British Royal Family, who was close to Diana, Princess of Wales. Many years have passed since the last time you saw her. Today you wish to remember her and browse through some events of her life.

The image shown below is one presentation of the memory collection about Diana. Please carefully read them and answer the questions below.

B1. To what extent do you agree this way of presenting information enables you to explore specific events (e.g. the birth of William and Harry) based on their chronological order?

- ☐ Strongly agree
- ☐ Agree
- ☐ Partly agree
- ☐ Disagree

B2. **To what extent do you agree it enables you to explore specific events based on people, locations, dates or other concepts, e.g. Charles and William, the Royal Wedding, etc.?**

- Strongly agree ☐
- Agree ☐
- Partly agree ☐
- Disagree ☐

B3. **How informative do you find this way of presenting information?**

- Very informative ☐
- Informative ☐
- Fairly informative ☐
- Not informative ☐

B4. **You must have some knowledge about Diana's life. With regard to the information collection provided to you, to what extent do you agree its content represents the most essential facts and events of Diana's life?**

- Strongly agree ☐
- Agree ☐
- Partly agree ☐
- Disagree ☐

B5. **With the information collection provided to you, to what extent do you think it can help you recall more associated information about Diana's life that is not mentioned by the collection?**

- Very helpful ☐
- Helpful ☐
- Fairly helpful ☐
- Not helpful ☐

B6. **With the information collection provided to you, do you think it is easy to read/understand?**

- Very easy ☐
- Easy ☐
- Fairly easy ☐
- Not easy ☐

Section C: Scenario I - B

The image shown below is one presentation of the memory collection about Diana. Please carefully read them and answer the questions below.

C1. To what extent do you agree this way of presenting information enables you to explore specific events (e.g. the birth of William and Harry) based on their chronological order?

- Strongly agree ☐
- Agree ☐
- Partly agree ☐
- Disagree ☐

C2. To what extent do you agree it enables you to explore specific events based on people, locations, dates or other concepts, e.g. Charles and William, the Royal Wedding, etc.?

- Strongly agree ☐
- Agree ☐
- Partly agree ☐
- Disagree ☐

C3. How informative do you find this way of presenting information?

- Very informative ☐
- Informative ☐
- Fairly informative ☐
- Not informative ☐

C4. You must have some knowledge about Diana's life. With regard to the information collection provided to you, to what extent do you agree its content represents the most essential facts and events of Diana's life?

- Strongly agree ☐
- Agree ☐
- Partly agree ☐
- Disagree ☐

C5. **With the information collection provided to you, to what extent do you think it can help you recall more associated information about Diana's life that is not mentioned by the collection?**

- Very helpful ☐
- Helpful ☐
- Fairly helpful ☐
- Not helpful ☐

C6. **With the information collection provided to you, do you think it is easy to read/understand?**

- Very easy ☐
- Easy ☐
- Fairly easy ☐
- Not easy ☐

Section D: Scenario I - C

The image shown below is one presentation of the memory collection about Diana. Please carefully read them and answer the questions below.

D1. **To what extent do you agree this way of presenting information enables you to explore specific events (e.g. the birth of William and Harry) based on their chronological order?**

- Strongly agree ☐
- Agree ☐
- Partly agree ☐
- Disagree ☐

D2. **To what extent do you agree it enables you to explore specific events based on people, locations, dates or other concepts, e.g. Charles and William, the Royal Wedding, etc.?**

- Strongly agree ☐
- Agree ☐
- Partly agree ☐
- Disagree ☐

D3. **How informative do you find this way of presenting information?**

- Very informative ☐
- Informative ☐
- Fairly informative ☐
- Not informative ☐

D4. **You must have some knowledge about Diana's life. With regard to the information collection provided to you, to what extent do you agree its content represents the most essential facts and events of Diana's life?**

- Strongly agree ☐
- Agree ☐
- Partly agree ☐
- Disagree ☐

D5. **With the information collection provided to you, to what extent do you think it can help you recall more associated information about Diana's life that is not mentioned by the collection?**

- Very helpful ☐
- Helpful ☐
- Fairly helpful ☐
- Not helpful ☐

D6. **With the information collection provided to you, do you think it is easy to read/understand?**

- Very easy ☐
- Easy ☐
- Fairly easy ☐
- Not easy ☐

Experiment II

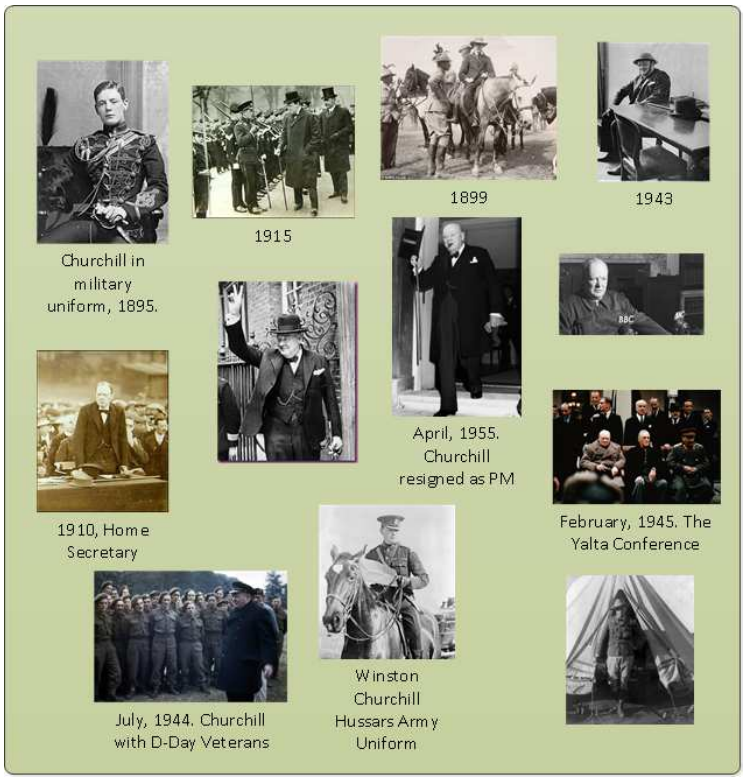


Figure D - 4 Experiment II: Unstructured presentation (Scenario I)

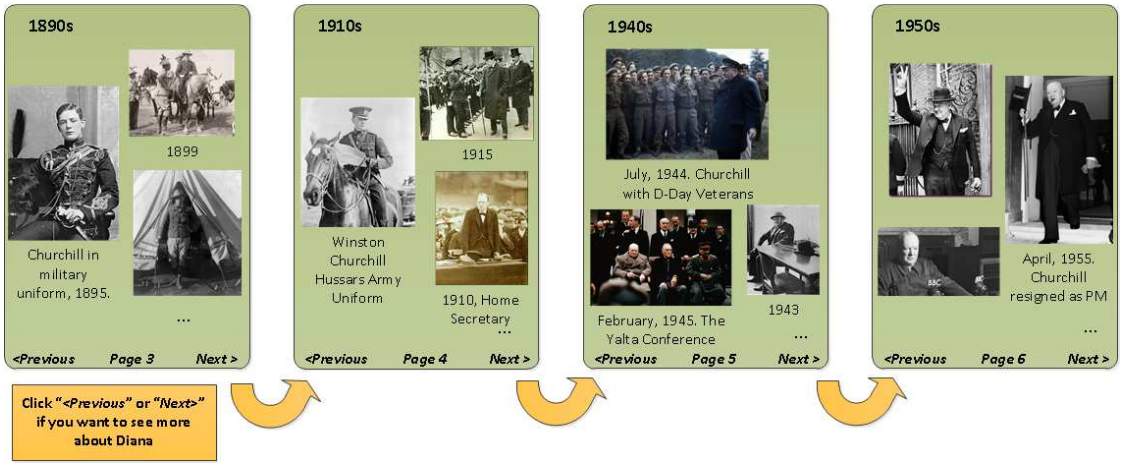


Figure D - 5 Experiment II: LSB-based presentation (Scenario II)

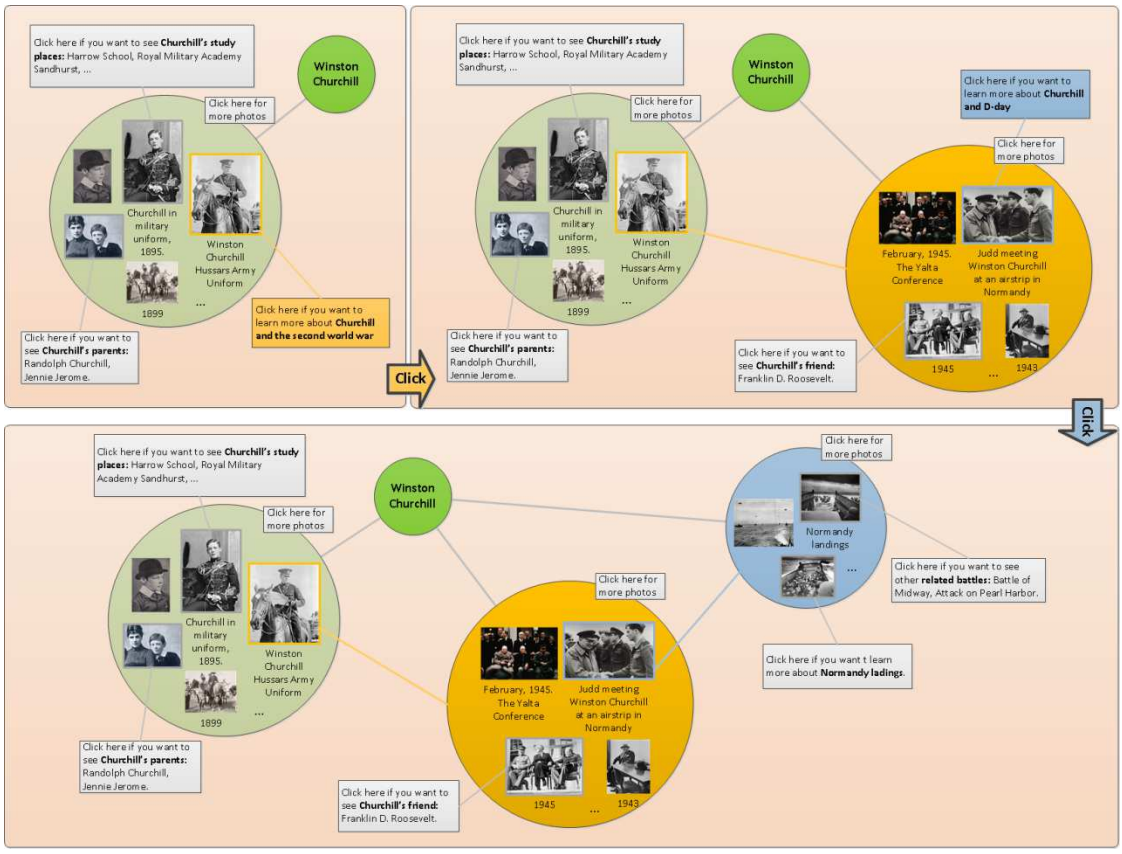


Figure D - 6 Experiment II: Semantic/knowledge-based presentation (Scenario III)

Section A: Amazon Worker ID

A1. Please provide your Amazon Worker ID:

Section B: Scenario II - A

Assume you are a veteran of British Army who attended the Second Word War. Many years have passed since the end of the war. Today, you wish to remember Winston Churchill, the person who was leading the country during the war, and browse the events related to him.

The image shown below is one presentation of the memory collection about Winston Churchill. Please carefully read them and answer the questions below.

B1. To what extent do you agree this way of presenting information enables you to explore specific events (e.g. the Yalta Conference and D-day) based on their chronological order?

- Strongly agree ☐
- Agree ☐
- Partly agree ☐
- Disagree ☐

B2. **To what extent do you agree it enables you to explore specific events based on people, locations, dates or other concepts, e.g. Winston Churchill, Franklin D. Roosevelt, Battle of Midway, etc.?**

- Strongly agree ☐
- Agree ☐
- Partly agree ☐
- Disagree ☐

B3. **How informative do you find this way of presenting information?**

- Very informative ☐
- Informative ☐
- Fairly informative ☐
- Not informative ☐

B4. **You must have some knowledge about Winston Churchill's life. With regard to the information collection provided to you, to what extent do you agree its content represents the most essential facts and events of his life?**

- Strongly agree ☐
- Agree ☐
- Partly agree ☐
- Disagree ☐

B5. **With the information collection provided to you, to what extent do you think it can help you recall more associated information about Winston Churchill's life that is not mentioned by the collection?**

- Very helpful ☐
- Helpful ☐
- Fairly helpful ☐
- Not helpful ☐

B6. **With the information collection provided to you, do you think it is easy to read/understand?**

- Very easy ☐
- Easy ☐
- Fairly easy ☐
- Not easy ☐

Section C: Scenario II - B

The image shown below is one presentation of the memory collection about Winston Churchill. Please carefully read them and answer the questions below.

C1. To what extent do you agree this way of presenting information enables you to explore specific events (e.g. the Yalta Conference and D-day) based on their chronological order?

- Strongly agree ☐
- Agree ☐
- Partly agree ☐
- Disagree ☐

C2. To what extent do you agree it enables you to explore specific events based on people, locations, dates or other concepts, e.g. Winston Churchill, Franklin D. Roosevelt, Battle of Midway, etc.?

- Strongly agree ☐
- Agree ☐
- Partly agree ☐
- Disagree ☐

C3. How informative do you find this way of presenting information?

- Very informative ☐
- Informative ☐
- Fairly informative ☐
- Not informative ☐

C4. You must have some knowledge about Winston Churchill. With regard to the information collection provided to you, to what extent do you agree its content represents the most essential facts and events of his life?

- Strongly agree ☐
- Agree ☐
- Partly agree ☐
- Disagree ☐

C5. **With the information collection provided to you, to what extent do you think it can help you recall more associated information about Winston Churchill's life that is not mentioned by the collection?**

- ☐ Very helpful
- ☐ Helpful
- ☐ Fairly helpful
- ☐ Not helpful

C6. **With the information collection provided to you, do you think it is easy to read/understand?**

- ☐ Very easy
- ☐ Easy
- ☐ Fairly easy
- ☐ Not easy

Section D: Scenario II - C

The image shown below is one presentation of the memory collection about Winston Churchill. Please carefully read them and answer the questions below.

D1. **To what extent do you agree this way of presenting information enables you to explore specific events (e.g. the Yalta Conference and D-day) based on their chronological order?**

- ☐ Strongly agree
- ☐ Agree
- ☐ Partly agree
- ☐ Disagree

D2. **To what extent do you agree it enables you to explore specific events based on people, locations, dates or other concepts, e.g. Winston Churchill, Franklin D. Roosevelt, Battle of Midway, etc.?**

- ☐ Strongly agree
- ☐ Agree
- ☐ Partly agree
- ☐ Disagree

D3. **How informative do you find this way of presenting information?**

- Very informative ☐
- Informative ☐
- Fairly informative ☐
- Not informative ☐

D4. **You must have some knowledge about Winston Churchill's life. With regard to the information collection provided to you, to what extent do you agree its content represents the most essential facts and events of his life?**

- Strongly agree ☐
- Agree ☐
- Partly agree ☐
- Disagree ☐

D5. **With the information collection provided to you, to what extent do you think it can help you recall more associated information about Winston Churchill's life that is not mentioned by the collection?**

- Very helpful ☐
- Helpful ☐
- Fairly helpful ☐
- Not helpful ☐

D6. **With the information collection provided to you, do you think it is easy to read/understand?**

- Very easy ☐
- Easy ☐
- Fairly easy ☐
- Not easy ☐

Experiment III

Figure D - 7 Experiment III: Unstructured presentation (Scenario I)

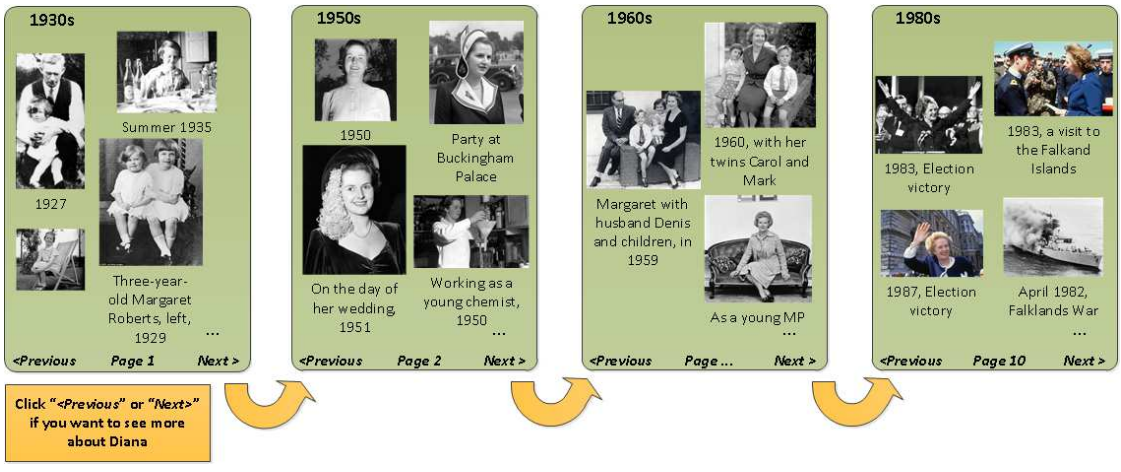


Figure D - 8 Experiment III: LSB-based presentation (Scenario II)

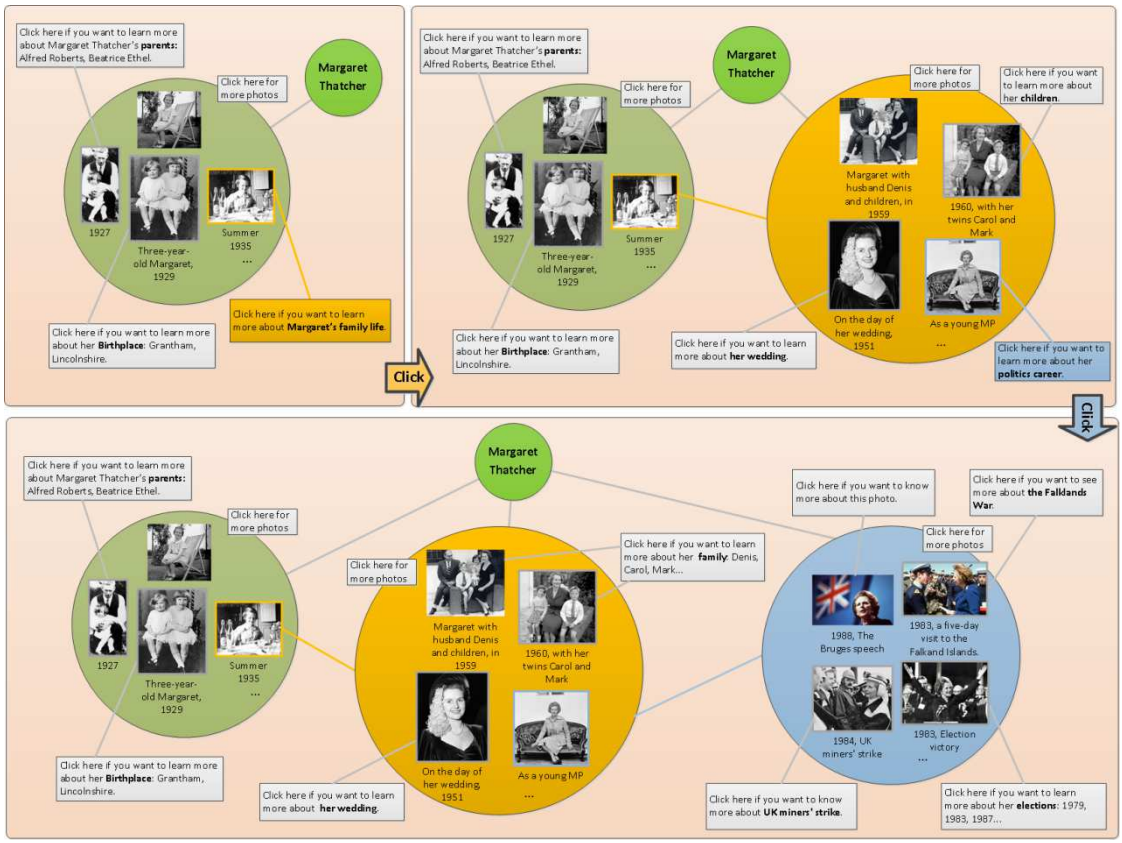


Figure D - 9 Experiment III: Semantic/knowledge-based presentation (Scenario III)

Section A: Amazon Worker ID

A1. Please provide your Amazon Worker ID:

Section B: Scenario IV - A

Margaret Thatcher was a famous British politician. Assume you were a close friend of her, you wish to remember her, and browse the events related to her life.

The image shown below is one presentation of the memory collection about Margaret Thatcher's life. Please carefully read them and answer the questions below.

B1. To what extent do you agree this way of presenting information enables you to explore specific events based on their chronological order, e.g. her growth history, elections, etc.?

- Strongly agree ☐
- Agree ☐
- Partly agree ☐
- Disagree ☐

D6. With the information collection provided to you, do you think it is easy to read/understand?

- Very easy ☐
- Easy ☐
- Fairly easy ☐
- Not easy ☐

Thank you! Your survey responses have been recorded, and the survey code has been sent to your email address.

Evaluation Results (Samples)

New Record	
id	10
Completed	16:08:50
Last page	4
Start language	en
Token	7eptf5zg9vjcp7g5bwnnkm2byez5uu22
Date started	16:06:21
Date last action	16:12:50
IP address	117.221.140.40
WORKER ID	A34XI67018IK8
G2_Q0001	Partly agree
G2_Q0002	Partly agree
G2_Q0003	Fairly informative
G2_Q0004	Partly agree
G2_Q0005	Fairly helpful
G2_Q0006	Very easy
G3_Q0001	Agree
G3_Q0002	Partly agree
G3_Q0003	Informative
G3_Q0004	Agree
G3_Q0005	Helpful
G3_Q0006	Very easy
G4_Q0001	Agree
G4_Q0002	Strongly agree
G4_Q0003	Very informative
G4_Q0004	Strongly agree
G4_Q0005	Very helpful
G4_Q0006	Easy

New Record	
id	11
Completed	16:15:55
Last page	4
Start language	en
Token	su8eve4asdp8y6wqku9tpnnqa438685x
Date started	16:07:29
Date last action	16:15:55
IP address	122.174.251.211
WORKER ID	A3PKJ0CX86ZAWN
G2_Q0001	Agree
G2_Q0002	Partly agree
G2_Q0003	Informative
G2_Q0004	Agree
G2_Q0005	Helpful
G2_Q0006	Easy
G3_Q0001	Agree
G3_Q0002	Partly agree
G3_Q0003	Informative
G3_Q0004	Strongly agree
G3_Q0005	Helpful
G3_Q0006	Very easy
G4_Q0001	Strongly agree
G4_Q0002	Strongly agree
G4_Q0003	Informative
G4_Q0004	Strongly agree
G4_Q0005	Very helpful
G4_Q0006	Fairly easy

New Record	
id	12
Completed	16:12:24
Last page	4
Start language	en
Token	92j2ns4qiyvwhhc5ua55nwg9ckekpwsx
Date started	16:09:42
Date last action	16:16:24
IP address	117.216.7.174
WORKER ID	A39QOA9M7GNF86
G2_Q0001	Partly agree
G2_Q0002	Agree
G2_Q0003	Informative
G2_Q0004	Agree
G2_Q0005	Fairly helpful
G2_Q0006	Easy
G3_Q0001	Strongly agree
G3_Q0002	Disagree
G3_Q0003	Fairly informative
G3_Q0004	Partly agree
G3_Q0005	Fairly helpful
G3_Q0006	Easy
G4_Q0001	Partly agree
G4_Q0002	Strongly agree
G4_Q0003	Informative
G4_Q0004	Partly agree
G4_Q0005	Helpful
G4_Q0006	Fairly easy

New Record	
id	13
Completed	16:27:18
Last page	4
Start language	en
Token	qtnts43nmebsuf4q4qapqjn5jrxdsxkg
Date started	16:10:39
Date last action	16:27:18
IP address	101.63.129.92
WORKER ID	panneerkumar44@yahoo.com
G2_Q0001	Agree
G2_Q0002	Agree
G2_Q0003	Fairly informative
G2_Q0004	Partly agree
G2_Q0005	Helpful
G2_Q0006	Very easy
G3_Q0001	Strongly agree
G3_Q0002	Partly agree
G3_Q0003	Informative
G3_Q0004	Agree
G3_Q0005	Very helpful
G3_Q0006	Very easy
G4_Q0001	Agree
G4_Q0002	Strongly agree
G4_Q0003	Informative
G4_Q0004	Agree
G4_Q0005	Helpful
G4_Q0006	Very easy

New Record	
id	14
Completed	16:16:29
Last page	4
Start language	en
Token	i439mz5hhmukg5pjsjzzwkegaa987j6p
Date started	16:11:53
Date last action	16:16:29
IP address	122.174.21.163
WORKER ID	A1IU5OP7BBZHZ7
G2_Q0001	Disagree
G2_Q0002	Disagree
G2_Q0003	Fairly informative
G2_Q0004	Partly agree
G2_Q0005	Fairly helpful
G2_Q0006	Fairly easy
G3_Q0001	Agree
G3_Q0002	Disagree
G3_Q0003	Informative
G3_Q0004	Partly agree
G3_Q0005	Fairly helpful
G3_Q0006	Easy
G4_Q0001	Agree
G4_Q0002	Agree
G4_Q0003	Very informative
G4_Q0004	Strongly agree
G4_Q0005	Very helpful
G4_Q0006	Easy

New Record	
id	15
Completed	16:18:25
Last page	4
Start language	en
Token	8j3swdjydgqt9b43hb9ccxam46ibvx33
Date started	16:11:57
Date last action	16:18:25
IP address	116.68.74.75
WORKER ID	A11UWQSS3CS0QB
G2_Q0001	Partly agree
G2_Q0002	Partly agree
G2_Q0003	Informative
G2_Q0004	Agree
G2_Q0005	Fairly helpful
G2_Q0006	Easy
G3_Q0001	Partly agree
G3_Q0002	Disagree
G3_Q0003	Very informative
G3_Q0004	Strongly agree
G3_Q0005	Very helpful
G3_Q0006	Very easy
G4_Q0001	Strongly agree
G4_Q0002	Agree
G4_Q0003	Informative
G4_Q0004	Agree
G4_Q0005	Helpful
G4_Q0006	Very easy

New Record	
id	17
Completed	16:18:38
Last page	4
Start language	en
Token	a3txj82pn2y2j8shm794x5an54g7dxem
Date started	16:15:07
Date last action	16:18:38
IP address	117.202.133.79
WORKER ID	A11Z4FC69JLF5U
G2_Q0001	Agree
G2_Q0002	Partly agree
G2_Q0003	Informative
G2_Q0004	Agree
G2_Q0005	Helpful
G2_Q0006	Easy
G3_Q0001	Agree
G3_Q0002	Strongly agree
G3_Q0003	Informative
G3_Q0004	Agree
G3_Q0005	Helpful
G3_Q0006	Very easy
G4_Q0001	Strongly agree
G4_Q0002	Agree
G4_Q0003	Very informative
G4_Q0004	Agree
G4_Q0005	Very helpful
G4_Q0006	Easy

New Record	
id	18
Completed	16:25:53
Last page	4
Start language	en
Token	7kizq5quxhgcvx3g4bvjbcaw22f4569a
Date started	16:16:07
Date last action	16:25:53
IP address	14.195.39.226
WORKER ID	A1DRFDGG2KCOI2
G2_Q0001	Disagree
G2_Q0002	Agree
G2_Q0003	Very informative
G2_Q0004	Agree
G2_Q0005	Helpful
G2_Q0006	Fairly easy
G3_Q0001	Agree
G3_Q0002	Agree
G3_Q0003	Informative
G3_Q0004	Agree
G3_Q0005	Helpful
G3_Q0006	Very easy
G4_Q0001	Agree
G4_Q0002	Strongly agree
G4_Q0003	Very informative
G4_Q0004	Strongly agree
G4_Q0005	Fairly helpful
G4_Q0006	Very easy

New Record	
id	19
Completed	16:22:02
Last page	4
Start language	en
Token	fj8sf883jhrwg9zke944kdn4uzxt397h
Date started	16:17:19
Date last action	16:22:02
IP address	113.193.110.144
WORKER ID	A2QQY4S73JO639
G2_Q0001	Agree
G2_Q0002	Agree
G2_Q0003	Very informative
G2_Q0004	Strongly agree
G2_Q0005	Very helpful
G2_Q0006	Very easy
G3_Q0001	Agree
G3_Q0002	Partly agree
G3_Q0003	Very informative
G3_Q0004	Strongly agree
G3_Q0005	Very helpful
G3_Q0006	Very easy
G4_Q0001	Strongly agree
G4_Q0002	Strongly agree
G4_Q0003	Very informative
G4_Q0004	Strongly agree
G4_Q0005	Very helpful
G4_Q0006	Very easy

New Record	
id	20
Completed	16:34:00
Last page	4
Start language	en
Token	a8kga6xz3e92yma2puecw8bu2d37uuj4
Date started	16:24:10
Date last action	16:34:00
IP address	98.157.203.239
WORKER ID	A1HBIE5LRTQK1L
G2_Q0001	Agree
G2_Q0002	Agree
G2_Q0003	Informative
G2_Q0004	Agree
G2_Q0005	Not helpful
G2_Q0006	Easy
G3_Q0001	Strongly agree
G3_Q0002	Agree
G3_Q0003	Very informative
G3_Q0004	Partly agree
G3_Q0005	Not helpful
G3_Q0006	Very easy
G4_Q0001	Strongly agree
G4_Q0002	Strongly agree
G4_Q0003	Very informative
G4_Q0004	Partly agree
G4_Q0005	Not helpful
G4_Q0006	Easy

New Record	
id	21
Completed	16:44:27
Last page	4
Start language	en
Token	bcw8bgci8dgfx52yahm9uafhge9ps2g4
Date started	16:26:25
Date last action	16:44:27
IP address	14.99.197.246
WORKER ID	A2LYUMT8OYAX25
G2_Q0001	Partly agree
G2_Q0002	Strongly agree
G2_Q0003	Very informative
G2_Q0004	Strongly agree
G2_Q0005	Very helpful
G2_Q0006	Fairly easy
G3_Q0001	Strongly agree
G3_Q0002	Strongly agree
G3_Q0003	Very informative
G3_Q0004	Strongly agree
G3_Q0005	Very helpful
G3_Q0006	Very easy
G4_Q0001	Strongly agree
G4_Q0002	Strongly agree
G4_Q0003	Very informative
G4_Q0004	Strongly agree
G4_Q0005	Very helpful
G4_Q0006	Very easy

New Record	
id	23
Completed	17:32:33
Last page	4
Start language	en
Token	tbmmfxfckwqpeta6x3v7r7spuaareef8
Date started	17:25:21
Date last action	17:32:33
IP address	220.225.28.138
WORKER ID	A2TZ5GVXZCMZMS
G2_Q0001	Agree
G2_Q0002	Agree
G2_Q0003	Very informative
G2_Q0004	Strongly agree
G2_Q0005	Very helpful
G2_Q0006	Very easy
G3_Q0001	Strongly agree
G3_Q0002	Agree
G3_Q0003	Very informative
G3_Q0004	Strongly agree
G3_Q0005	Very helpful
G3_Q0006	Very easy
G4_Q0001	Strongly agree
G4_Q0002	Strongly agree
G4_Q0003	Very informative
G4_Q0004	Strongly agree
G4_Q0005	Very helpful
G4_Q0006	Very easy

New Record	
id	24
Completed	17:32:59
Last page	4
Start language	en
Token	w4f8waea2sxw6g7jzgpgpn9k69anc8v9
Date started	17:25:30
Date last action	17:32:59
IP address	117.193.179.71
WORKER ID	A38U1R9L2OEOH1
G2_Q0001	Partly agree
G2_Q0002	Partly agree
G2_Q0003	Very informative
G2_Q0004	Strongly agree
G2_Q0005	Very helpful
G2_Q0006	Very easy
G3_Q0001	Strongly agree
G3_Q0002	Partly agree
G3_Q0003	Fairly informative
G3_Q0004	Strongly agree
G3_Q0005	Very helpful
G3_Q0006	Very easy
G4_Q0001	Strongly agree
G4_Q0002	Strongly agree
G4_Q0003	Very informative
G4_Q0004	Strongly agree
G4_Q0005	Very helpful
G4_Q0006	Easy