Dedication

For my parents, Mrs Mengjiao Sun and Mr Jüren Dai

献给我的双亲：孙梦姣与戴居仁

Declarations

I declare that I am the sole author of this thesis.

I declare that all verbatim extracts contained in the thesis have been identified as such and sources of information specifically acknowledged.

I certify that where necessary I have obtained permission from the owners of third party copyrighted material to include this material in my thesis.

Yixing (Miki) Sun

Aberdeen, Scotland

USING THE ORGANIZATIONAL AND NARRATIVE THREAD STRUCTURES IN AN E-BOOK TO SUPPORT COMPREHENSION

Yixing Sun

孙逸行

A Dissertation submitted to

The Robert Gordon University

in partial fulfilment of the requirements for the

Degree of Doctor of Philosophy

Supervisor: Prof. David J Harper and Dr. Stuart N. K. Watt

School of Computing

The Robert Gordon University

August 2007

ABSTRACT

Stories, themes, concepts and references are organized structurally and purposefully in most books. A person reading a book needs to understand themes and concepts within the context. Schank's Dynamic Memory theory suggested that building on existing memory structures is essential to cognition and learning. Pirolli and Card emphasized the need to provide people with an independent and improved ability to access and understand information in their information seeking activities. Through a review of users' reading behaviours and of existing e-Book user interfaces, we found that current e-Book browsers provide minimal support for comprehending the content of large and complex books. Readers of an e-Book need user interfaces that present and relate the organizational and narrative structures, and moreover, reveal the thematic structures.

This thesis addresses the problem of providing readers with effective scaffolding of multiple structures of an e-Book in the user interface to support reading for comprehension. Recognising a story or topic as the basic unit in a book, we developed novel story segmentation techniques for discovering narrative segments, and adapted story linking techniques for linking narrative threads in semi-structured linear texts of an e-Book. We then designed an e-Book user interface to present the complex structures of the e-Book, as well as to assist the reader to discover these structures.

We designed and developed evaluation methodologies to investigate reading and comprehension in e-Books, in order to assess the effectiveness of this user interface.

We designed semi-directed reading tasks using a Story-Theme Map, and a set of corresponding measurements for the answers. We conducted user evaluations with book readers. Participants were asked to read stories, to browse and link related stories, and to identify major themes of stories in an e-Book. This thesis reports the experimental design and results in detail. The results confirmed that the e-Book interface helped readers perform reading tasks more effectively. The most important and interesting finding is that the interface proved to be more helpful to novice readers who had little background knowledge of the book. In addition, each component that supported the user interface was evaluated separately in a laboratory setting and, these results too are reported in the thesis.

KEYWORDS

e-Book, user interface, evaluation, comprehension, cognition, learning, theme, story segmentation, information retrieval, story linking, narrative threads, narrative structure, directed reading, reading tasks, topic detection and tracking.

PUBLICATIONS

Sun, Y., D. J. Harper, et al. (2005). Aiding Comprehension in Electronic Books Using Contextual Information. In proceedings of The 9th European Conference on Research and Advanced Technology for Digital Libraries (ECDL), Vienna, Austria, Springer-Verlag. 19-23 September, 2005

Sun, Y. (2004). Discovering and Representing the Organizational and Narrative Structures of e-Books to Support Comprehension. In proceedings of the 27th annual international ACM SIGIR Conference on Research and Development in Information Retrieval, Sheffield, UK, ACM Press. 24th - 28th July, 2004

Sun, Y., D. J. Harper, et al. (2004). Design of an e-Book User Interface and Visualizations to Support Reading for Comprehension. In proceedings of the 27th annual international ACM SIGIR Conference on Research and Development in Information Retrieval, Sheffield, UK, ACM Press. 24th -28th July, 2004

ACKNOWLEDGEMENT

First of all, I would most like to thank my supervisor: Professor David Harper and Dr Stuart Watt, for their continuous guidance, support, and encouragement throughout my study. David has been a wonderful advisor to me and major influence in my academic life. I could not possibly list all that I have learned from him, such as how to do research, give a talk, and work with others. He is the one I can always go to without making an appointment. I can never forget on the day of my viva, he called me three times from abroad, just to give me his blessings. Stuart always amazes me with his broad knowledge, which ranges from programming language, internet technology, to cognitive psychology. He has a store of solutions and insights – the problem is that, I need to know what to ask first!

I would also thank specially Dr Iain Pirie, for his advices in the experimental design and data analysis, which forms important parts of this research. Iain has left a good example for me of being patient and mindful of small things – after all the difference between significant and insignificant is only 0.001 – and some times that is all what we need to know.

I want to thank the head of School of Computing, Professor Susan Craw and Mr David Davidson, and the ORS Award committee, who together provided financial support for this research. I thank the research coordinators and officials, Dr Ayse Göker, Dr Ines Arana, Mr Martin Simpson, and Mrs Maggie Nicol for their support on the project management and progress tracking. I thank the school admin and IT teams for their kind support.

I am grateful for the help and inspiration from many staff and students in my school: Professor Patrik Holt, Mr Alex Wilson, Dr John McCall, Dr Robin Boswell, Dr Bob Noble, Dr James Christie, Miss Stella, Mr Ulises, Mr Ralf, Miss Fiona, Mr Zia, Mr Kefeng and Mr Bicheng.

I want to thank numerous researchers for their insightful and challenging comments on my work at various conferences, including Professor John Tait, Professor Alan Smeaton, Professor Fabio Crestani, Professor Peter Ingwersen, Professor Gary Marchionini, Professor Nick Belkin, Professor Ed Fox, Professor Boots Lillian Cassel, Dr Yoelle Maarek, Dr Will Reader, Dr Dianne Kelly and Dr Nicola Stokes.

I thank the International Bible Society for giving us permission to use the New International Version of Bible for iSee e-Book user evaluation! I appreciate software and gifts from the Logos Research Systems, USA. I am grateful for the ACM SIGIR Conference (2004), the European Summer School on Information Retrieval (2003) and the ECIR conference (2003) for their travel grants and support.

I would like to thank:

- users who offered their precious time and efforts to participate in the user evaluation;
- Mr. Jeffrey Gooding who helped in writing the Bible study material for the reading tasks;
- Mr and Mrs Kumar who coded the task sheets for the user experiments.

The last year of my writing up has been supported by the Energy Centre, where I worked as an assistant ODL developer. I am grateful for Mr Wynand Hoogerbrugge and Mr Duncan Stephen, who granted me the job and entrusted me with many development opportunities. I am indebt of Mr David Robertson, who has proof-read my thesis; and Mr Cameron Liang, who has participated in my user experiments and encouraged me in many ways. I am proud of being in midst of my smart colleagues here.

Thanks also to my buddies at Hebron Evangelical Church for their friendship, hospitality, footsteps of faith, and prayer support. I am grateful to my Bible teacher Mrs Sheila Watt, for her wonderful insight of the words of God and her precious time spent with me. The Bible study and sermons I learnt in church have been a resource and inspiration for my research, particularly in the way when stories and passages were related and cross-referenced to reveal themes.

I thank my friends Tammy & Ronnie, Maggie & Tony, Yafan & Sam, Zhuqi & Matthew, and Jiaolan & Steve for their friendship and support. Dr Zhikang Yin and his wife Aimei has adopted me into their family since the first day I came to Aberdeen, I appreciate so much their love.

Most of all, I thank my parents and my big family in China who supported me over the years and endured hardship with me. I would not be able to finish the project without my mother’s faith in me. “妈妈，感谢您对我的付出和信任，这份成绩是属于您的。我爱你，我的好妈妈。”

After all, I thank my Lord Jesus Christ. This thesis was directly inspired by Him and for Him. “Not by might, nor by power, but by my Spirit” (Zechariah 4:6). Thank you, my Jesus, my Saviour.

Table of Contents

List of Figures

List of Tables

I RESEARCH QUESTIONS AND LITERATURE REVIEW

1

INTRODUCTION

This thesis addresses the problem of providing readers with scaffoldings of multiple structures of e-Book in the user interface to support reading for comprehension. Scaffoldings refer to different kinds of supports that readers receive in their interaction with e-Book. The problem is particularly relevant today, because reading and learning has been increasingly done in electronic format through computers and network. The work addresses the problem from the perspective of user interface presentation of the book contents, in terms of organizational and narrative structure, in software design of e-Books.

The main idea in this thesis is to adapting text comprehension theories and models, story segmentation and story linking technologies, and a narrative text corpus into a design framework that combine these aspects together to support real world reading for comprehension tasks.

This chapter will describe the problem, present the design goals and propose research questions. It concludes by summarizing the contributions of this thesis and drawing a map of the structure of the entire thesis.

1.1 Motivation

Pirolli and Card (Pirolli and Card 1999) indicated that, since the beginning of time, people have attempted to seek, gather, learn, use, and consume information to a degree un-approached by other organisms. Providing people with an improved ability to access and understand available information has been a social aim for many movements at least since the Enlightenment, and it is also the aim of more mundane and practical efforts of improving modern-day productivity. Technological innovation has led to an explosive growth of recorded information. Since the middle of the 20th century, information is increasingly prepared, stored, processed and read using computers. The World Wide Web and Internet technology has changed modern life dramatically in the past decade. Not only are we communicating, shopping, and entertaining online, but we are also reading, writing, learning, researching, and publishing over the Internet. Electronic books and digital libraries have become more and more popular in the modern community. However, e-Books and hypertext should have many more advantages compared to paper-based books than they do today.

People need to recall and compare different events, stories, and experiences to solve everyday problems; students need to learn different definitions and solutions; researchers need to compare different views and theories; readers need to scan, remember, and comprehend different narratives and discourses. Various advances have been made in the areas of formatting and display, layout, navigation and finding information in e-Book software (Egan, Remde et al. 1989; Woodruff, Gossweiler et al. 2000; Crestani and Ntioudis 2002). However, it has yet to be determined how these diverse features may best be merged into the e-Books of the future, and how the interface and functionality of such e-Books will need to be adapted or customized for different tasks (Chignell, Golovchinsky et al. 1993). Providing advanced assistance to enhance book-reading tasks is becoming an important challenge that overlaps the fields of Digital Libraries, Human Computer Interaction, and Information Retrieval.

Hearst (Hearst 1999) argued that information access tasks are used to achieve people's information seeking goals. These tasks span the spectrum from asking specific questions to researching a topic exhaustively. Likewise, reading and comprehending an e-Book also requires assistance for certain reading tasks.

Modern cognitive psychology assumes that the knowledge level can be given a scientific account (i.e. be made predictable) by explaining it in terms of mechanistic information processing (Newell 1990). The cognitive level of analysis focuses on the properties of the information processing machinery that humans use to perceive, think, remember, learn, and act in what we would call purposeful and knowledgeable ways (Pirolli 1999). This assumes that people are not infinitely or perfectly rational. Because we have only finite powers and resources – we operate with only limited information and limited computational ability – a more successful hypothesis about humans is that they exhibit bounded rationality or make choices based on satisficing (Simon 1955). Schank's theory of dynamic memory also suggested that building on existing memory structures is essential to cognition and learning (Schank 1982a; 1982b).

Reading a story is an astonishing feat of information processing, requiring the reader to perform complex operations at a number of levels (Emmott 1997). After decoded the letters, assigned meanings to the words, the reader must also judge how a sentence is linked to the previous text and make inferences based on general knowledge or stored information. As one of the major information access devices, e-Books should provide insight to the content of the book by the advanced information processing and presentation technology, with consideration of cognition process, to help the reader identify and understand the important themes of the book. However, e-Book and hypertext book design guidelines do not provide concrete design rules based on theory, but instead present abstract rules based more on common sense (Foltz 1996). This might be explained by the fact that most of the practices in the field are carried out by computer specialists, rather than psychologists and educators. Neither a general theory of hypertext nor a model of the cognitive process involved in reading exists (Altun 2000). Additionally, little research has been done by researchers and educators to assess hypertext's potential impact on, and implications for, reading and literacy education (Altun 2003; Protopsaltis and Bouki 2005).

A survey of users of e-Books in the UK (Gunter 2005) indicated the market potential of e-Books given that the equipment needed to read them is neither too expensive nor too difficult to use. It is clear, however, that early e-Book users regard electronic reading as something to use primarily for reference work rather than for more extended reading for leisure and entertainment. Most e-Book users still prefer not to read extended passages of text from a screen. Nonetheless, for dipping in and out of reference works e-Books have the advantage of being easier to search and easier to annotate.

To what extent can this advantage be developed? Can e-Book software help readers to recall and understand the context by drawing the relevant narrative pieces together, and providing contextual information of the stories? Could e-Book software enhance a reader's reading experience and improve their comprehension? How can e-Book software take advantage of the latest developments in information retrieval, text processing, cognition and learning technology and offer more features to its readers?

Although various efforts have been made, current e-Book software user interfaces generally only support minimal navigation of organizational structure. In principle, however, e-Book software could help readers to find themes by drawing the similar and related narrative pieces together, and providing contextual information of the events and stories. We emphasized the importance of such research, and argued that presenting narrative structural information will help readers' comprehension process (Sun 2004; Sun, Harper et al. 2004; 2005).

In the following thesis we present the method of retrieving and linking the similar and related stories in a complex book, and the evaluation of such an approach. This research aims to investigate and design a user interface that can support narrative structure navigation in addition to an interactive navigation of organizational structure, and study the impacts of such user interface in readers' reading and comprehension activities.

1.2 What is this Study About and Not About?

We can view a book as a corpus made up of texts, symbols and pictures, by which topics are knitted together through multiple structures (see 1.3.1). Authors describe a theory, a topic or a story using texts, while readers understand the texts and re-structure the theory, the topic or the story in their mind. There is a process from the imagination of the author to the imagination of the reader of a book. E-Book software can play an important role to visualize the text, and to build a bridge between the author and the reader, to shortening the processing time needed for a reader to grasp the author's vision.

When search for the word 'e-Book', you might be surprised at the variety of results and spellings you can find. Regarding spellings, there are 'ebooks', 'e-Books', 'eBooks' and 'E-Books'. Regarding what those names refer to, there are the handheld devices of e-Books, e.g., Gemstar (a trademark for both handheld device and formatted e-Books it supports. The device is discontinued but e-Books in its format are still available); the desktop software for e-Books, e.g., Adobe Reader and Microsoft Reader; and electronically formatted text presented in text (.txt), PDF (.pdf) or hypertext (.htm) format, to be read using a computer.

In this thesis, the software that presents a book on a computer will be called 'e-Book reading software', in short, an 'e-Book'. This study is focused on e-Book (including hypertext book) content analysis, and on user-centred design of the user interface for effective e-Books. The main application of this work is on the design and evaluation of user interface for e-Books to support reading for comprehension.

We will discuss learning and comprehension, and its impact on e-Book user interface design. The results of this study will provide insights in the fields of teaching and learning, artificial intelligence and machine learning.

This study is not about e-Book hardware, devices, formatting, authoring and publishing issues, although some aspects of e-Book in these domains will be reviewed when it is related to the user interface issue.

1.3 Book Structures

1.3.1 Multiple Structures of a Book

An e-Book, especially a literary one, is constructed from the following constituent parts: language, characters (people), stories (events), organizational structure, themes, and references (Sun, Harper and Watt, 2004). These materials are organized in different ways to present different structures of the book.

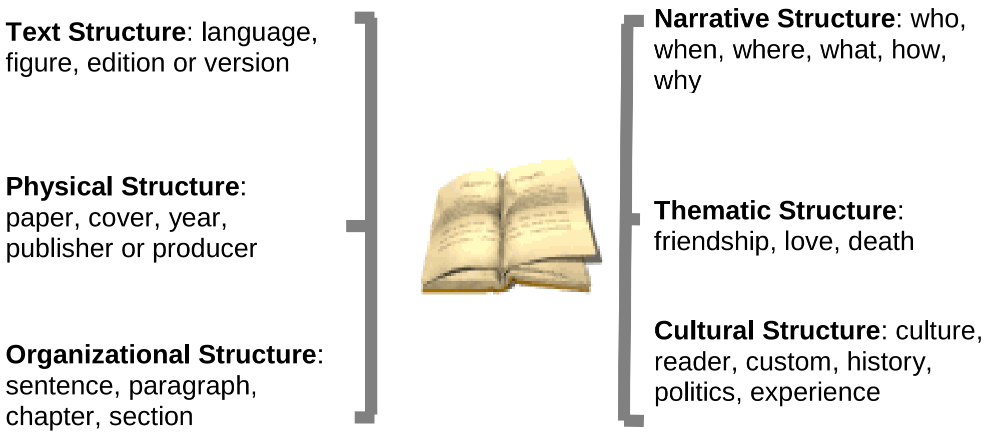

Text Structure: language, figure, edition or version
Physical Structure: paper, cover, year, publisher or producer
Organizational Structure: sentence, paragraph, chapter, section
Narrative Structure: who, when, where, what, how, why
Thematic Structure: friendship, love, death
Cultural Structure: culture, reader, custom, history, politics, experience

Figure 1 Multiple structures of a book

We can view the book as structured according to its many characteristics:

- A text structure gives information about language, edition or version.
- A physical structure gives information on the physical status of the book.
- An organizational structure provides information on the organization of the book.

- A narrative structure provides information about the events and characters.
- A thematic structure provides information on the major themes (see below).
- A cultural structure views the book from its' cultural background, history, country, politics, and individual reader's background, experience and response.

Different domain uses different structure information of a book. For example, librarians are interested in the text and physical structures of a book; linguists are interested in the text, organizational and narrative structures of a book; literaturists and philosophers are interested in the organizational, narrative and thematic structures of a book; historians and socialists are interested in the narrative and cultural structures of a book; and so on.

More fundamentally and commonly, read a story or a book and find out its theme is essential in our childhood language education. What is a theme?

> *"In literature, a theme is a broad idea in a story, or a message or lesson conveyed by a work. A theme is often about life, society or human nature. Themes may be fundamental or universal ideas explored in a literary work. They are usually implied rather than explicitly stated." (Booth 1961)*

Many novels and stories contain more than one theme, and many stories can share one theme. Information on the more basic structures, for example, the organizational and narrative structure, will help readers understand the higher level structures, such as the thematic structure. Providing e-Book software to reveal the narrative and thematic structures of the book, and to help readers gain a better understanding of the book in a shorter time, is the main theme of this study.

In chapter 2 we will review theories of learning in more detail and will propose our solutions in chapter 4.

1.3.2 A Sample of Multiple Structural Information of One Story

Figure 2 illustrates a sample of the multiple structures information of the story of Jacob’s marriage in the context of the Christian Bible (see 4.2.2). The original story was told in Genesis 29, where Jacob married two sisters, Rachel and Leah and had twelve sons. In other parts of the Bible, Jacob and his families are mentioned many times, for example, in Ruth chapter 4 and Matthew chapter 1. These passages form a thread of stories of Jacob’s. But why is the story significant in the Bible?

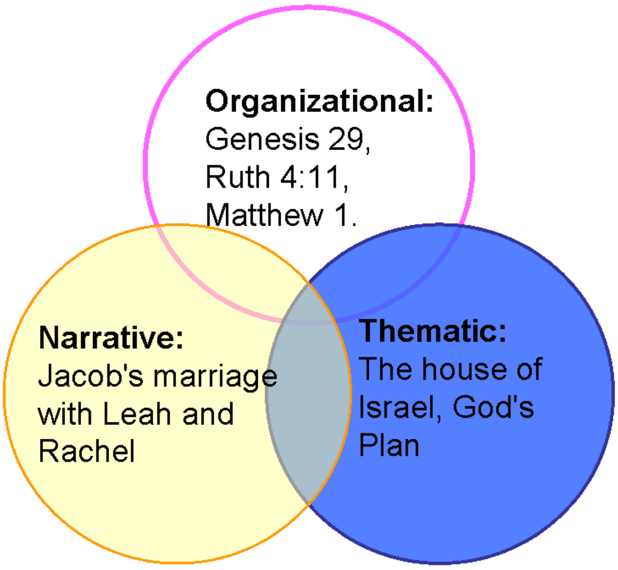


Figure 2 Three Views of One Story

It is because the Jacob, whose name was changed to ‘Israel’ by God, is the father of ‘the house of Israel’. This theme, ‘history of Israel’, like it is in many other books, is not made plainly in words, but is hidden behind a narrative thread of many stories and passages; the narrative thread structure, on the other hand, is linked and cross-referenced through the organizational structure of the Bible, where the stories are located.

1.4 Needs Analysis

1.4.1 What Do People Do with e-Books?

People read for various purposes, such as study, amusement, and research. What is in common when a reader is reading or studying a book, is that she often needs to build her understanding upon the available materials, reflecting on it with her own experiences and other relevant knowledge, and then rebuilding the story in her mind. When a review author is preparing a literature study or review of a particular book, she also needs help in exploring the themes and threads of the book and identifying the cross-references quickly and accurately. These scenarios could be applied to many kinds of different books, including fiction, text books conference proceedings, digital journals, the Bible, etc. These demands are particularly important when the book is long and complex.

Compared to reading a printed book, reading e-Books can provide the reader with some obvious advantages including:

- Easy availability – they can be easily and instantly obtained via the Internet, and printed on demand, thus cheaper and more economic than paper books.
- Enhanced readability – the view of an e-Book (enlarge font size and style, change orientation on device, modify screen contrast) can be customized, and they can enable the user to search for the specific key words, passages, chapters. Navigation can be easy with a keyboard, mouse, or stylus. They can

also be used hands-free, and possibly even eyes-free, with text-to-voice features.

- Additional interaction – they allow book-marking, note taking, drawing, highlighting and annotating capabilities.

However, how these features could assist readers to understand the content of a book is yet to be studied.

In this study, we have used scenarios to guide the development of both our vision and our prototypes. A scenario is a starting-point for design. In computing, a scenario is a narrative describing foreseeable interactions of types of users (characters) and the system or between two software components. Scenarios are often used in usability research. They include information about goals, expectations, actions and reactions (Rosson and Carroll 2001).

1.4.2 Scenario One: Searching and Browsing

The following typical scenario describes a novice reader using e-Book software:

> *Kuku is a young girl from Japan. She is studying the Communication and Media course in a college in the UK. She heard about Christianity when she went to church for Christmas Carol service, but she never read Bible.*
>
> *Encouraged by a Christian friend, she started to read Bible stories using e-Book software on her computer.*
>
> *She was reading the book of Genesis chapter 29 to 30 one day. It is a story about Jacob's marriage with two sisters, Leah and Rachel. The software provides an interactive visualization tool in the user interface to show the structure, content and length of the Bible and where the current story is. Kuku uses the tool to browse the chapters and revisit the chapters she read.*

> *Kuku would like to know more about the family, so she read through the book chapter by chapter. The story seems to be long and becomes boring. While she is reading, she wonders why the story was significant to be mentioned in the Bible. She is interested in what other people said about the story and where in other parts of the Bible this story is mentioned again.*
>
> *Using the software she could search for the similar stories. I.e., in the story of 'Moses Blessed the Tribes' in Deuteronomy chapter 31, the name of twelve sons of Jacob are mentioned; Rachel and Leah's names are mentioned again in Ruth chapter 4, when Boaz Marries Ruth; finally in Matthew chapter 1, the name of Jacob, his son Judah, and the names of Boaz and Ruth are all mentioned in 'The Genealogy of Jesus'. After reading the thread of stories, Kuku realised Jacob was the forefather of the nation Israel and one of the ancestors of Jesus; and his family is chosen by God to become a big nation; the story of Jacob and his family is significant to show God's power and work in ordinary people's lives.*

Similarly, the e-Book software could be used in many ways to serve the need of readers and users. A boy who wanted to write a love letter to his girlfriend could search for all the famous declaration sceneries, e.g., Romeo and Juliet on the balcony, Emma and Mr Knightley in the garden (Austen, 1815), and so on. Students and teachers could use e-Book software for academic text books, journals and conference proceedings, to learn new concepts and broaden their knowledge in a subject. Historians and news reporters could discover and link relevant events that might not necessarily be obviously related. Lawyers could search for similar cases in documentation for the use of law. The detection of similar passages across different articles is also an important challenge in plagiarism detection and prevention.

1.4.3 Scenario Two: Reading and Analysing

On the other hand, experienced or expert readers might find e-Book software useful in a different way:

> *Steve Rankin is a professional dentist and has been studying the Bible for many years; he is often invited to speak in the church and share his insights and understanding of stories in the Bible. He was invited by the Hebron Evangelical Church one day to give a talk on the subject of suffering. He decides to talk about Satan's attack towards Jesus in the Old Testament.*
>
> *He chooses e-Book software to help the preparation because this software provides a Story-Theme map that illustrates the similar stories and their themes in the Bible.*
>
> *Firstly, he chooses 'suffering' as a topic, and then chooses Old Testament as the data set. The system shows a map of subjects which are relevant to 'suffering': suffer, disease, slaughter, mourn, plague, kill, cry, misery, corruption, die and war, etc in the Old Testament. With each subject the system lists the passages where the event occurred, together with the time, location and people information, etc.*
>
> *With this Story-Theme map in hand, Steve then starts to prepare the PowerPoint presentation.*
>
> *He chooses some significant events in the Story-Theme map of 'Suffering' which almost break down the family line of the Genealogy of Jesus. For example: Corruption of Adam's offspring (Genesis chapter 6), Corruption of Abraham's seed (Genesis chapter 12 and 20), Famine (Genesis chapter 50), Saul's jealous over David (1 Samuel chapter 18), the corruption of Israel's Kings (1 and 2 Kings), etc. and finally, Haman attempts to annihilate Jews in Persian (Esther chapter 3), etc.*
>
> *Steven uses these stories to argue that if any of these attempts have been totally successful, the Bible would not stand today and God's plan of salvation would not have been carried out by Jesus.*
>
> *The e-Book software helped him to prepare a slide showing a timeline of events.*

Similarly, the e-Book software could be used in assisting the researchers in other analytical studies. For example, the linguists and literaturists could use this kind of e-Book software for literature comparison, meta-fiction, structuralism, and post-structuralism studies, to analyse and compare different authors, styles, versions, narratives, views, and episodes, etc. Researchers could use it to explore and analyse the relationships between the views, topics, concepts, themes, theories, models, and other units of information the text may carry.

1.5 Research Questions and Design Goals

1.5.1 Research Questions

This thesis addresses five primary research questions:

- How can a framework be designed that incorporates reading for comprehension theory, information retrieval innovations, and software user interface design principles in the design of an e-Book software application? (chapter 4)
- How can the organizational and narrative structures of e-Book be semi-automatically discovered? (chapter 4, 5 and 6)
- How should such structures be presented in an e-Book user interface? (chapter 4)
- How can an e-Book application be evaluated in real world simulated reading and comprehension tasks? (chapter 7) and
- How can the performance of an application in a previously unexplored domain for lexical cohesion analysis, such as the narrative literature, be evaluated and measured? (chapter 4, 5, 6 and 7)

A set of secondary goals arising from the above are also addressed in the thesis:

- How can meaningful and purposeful real world simulated reading tasks be designed? (chapter 2)

- How can the users' comprehension performance with applicable measures be assessed? (chapter 7)
- How can the navigation of narrative structure with the navigation of organizational structure of e-Books be integrated? (chapter 4)

The key assumptions of this research are:

- Providing narrative threads structure information in the user interface (in addition to the organizational structure information) can help users to know more about the stories, events and themes.
- By using the tool, users will be able to select and browse through sections and stories of the book quickly and easily.
- The users will find the tool satisfying to use for e-Book reading, because of the multiple structural and contextual information provided for the overview of the book.

1.5.2 Design Goals

The e-Book user studies and scenarios outline a concrete list of design goals for an e-Book reading software user interface. This user interface should support:

- Visualization of organizational structure in an e-Book;
- Presentation of the narrative structure in an e-Book;
- Discovery of thematic structure through the organizational and narrative structures of an e-Book.

These features will provide scaffoldings to support readers to discover and understand the multiple structures of a book. More details on the design will be discussed in chapter 4.

1.6 Contributions

1.6.1 User Interface Design Framework

With a vision to design an intelligent e-Book user interface for supporting readers browsing and comprehension of large complex books, we built a synthesis of e-Book user interface design framework that combined the user scenarios, comprehension theory (e.g., Kintsch 1998, Hoover and Gough 1990, reviewed in chapter 2), e-Book design guidelines and state of art (e.g., Wilson and Landoni 2002, Puntambekar, Stylianou et al. 2003, reviewed in chapter 3), and information processing technologies (reviewed in chapter 5 and 6 accordingly). Various ideas of scaffolding were applied in the system design.

This characteristic of the study also determined itself to be cross-disciplinary, thus we used and developed different evaluation methodologies, corpora, measures, etc, for various purposes. Details of the design framework are reported in chapter 4.

1.6.2 Evaluation Methodologies

Since this project is cross-disciplinary, we have been using and developing different evaluation methodologies. In order to assess the effectiveness of the user interface in supporting reading and comprehension activities, we designed and developed evaluation methodologies on reading and comprehension of e-Books. In detail:

- We designed two e-Book user interfaces for performance comparison purposes, one has the narrative structure information presented on the user interface, and the other has a query search tool for readers to manually discover the narrative structures.
- We designed real word simulated information seeking tasks (Borlund 2003) to evaluate the system's performance in assisting a user's information retrieval need, and a set of corresponding quantitative measures for these close-end questions;
- We designed semi-directed reading and comprehension tasks to evaluate the system's performance in assisting a user's comprehension need, and a set of corresponding quantitative and qualitative measures for these open-end questions.
- We designed an experimental protocol using a repeated measures experimental design, and carried out a user evaluation with both book readers and computer users.
- We found that our system with narrative structure information presented on the user interface is significantly faster than the comparison system in the information seeking tasks; meanwhile, it is more effective in helping readers to understand the stories and identify major themes of the stories in the e-Book.

In general, users felt the two systems are both easy to learn and easy to use and they enjoyed the reading experiments. The experimental design and results are reported in more detail in chapter 7.

1.6.3 Narrative Structure Detection

We developed semi-automatic detection algorithms of narrative structure of e-Book: we segment the e-Book into story segments and created a collection of stories; we used an information retrieval technique to generate a matrix of similarity for this story collection, in which each story is compared to every other in order to get a thread of ranked similar stories.

We evaluated our approach with comparisons to Lucene, an open source text retrieval tool. Our system is significantly more effective than Lucene as measured by IR Precision and F-measure. This is reported in chapter 6.

1.6.4 Story Segmentation and Novice Corpus

To achieve the story segmentation goal, we improved the TextTiling (Hearst 1997) story segmentation technique by using a symmetric divergence similarity measure instead of the ad hoc Cosine-distance measure. We developed novice narrative corpus and acquired solid 'ground truth' from other resources for system evaluation. We compared and evaluated various similarity measures and other factors in the segmentation tasks, i.e., text window sizes, effects of character names, and cut-off thresholds, etc. Our approach is significantly more effective in identifying topic boundaries in narrative text as measured by IR F-measure, reported in chapter 5.

1.7 Thesis Map

Figure 3 illustrates an overview of the thesis. In this chapter, we have introduced the problem, the research questions and our contributions to the field. Next, we will review the previous works in chapter 2 and 3; outline the design of the user interface in chapter 4; and describe and report the story segmentation and evaluation in chapter 5. In chapter 6, we will report on the narrative structure detection strategy, and evaluate the approach. We will describe the experimental design and user experiments in chapter 7; we will review our research questions and make further analysis and discussion in chapter 8. Finally, we will discuss and present our conclusions in chapter 9.

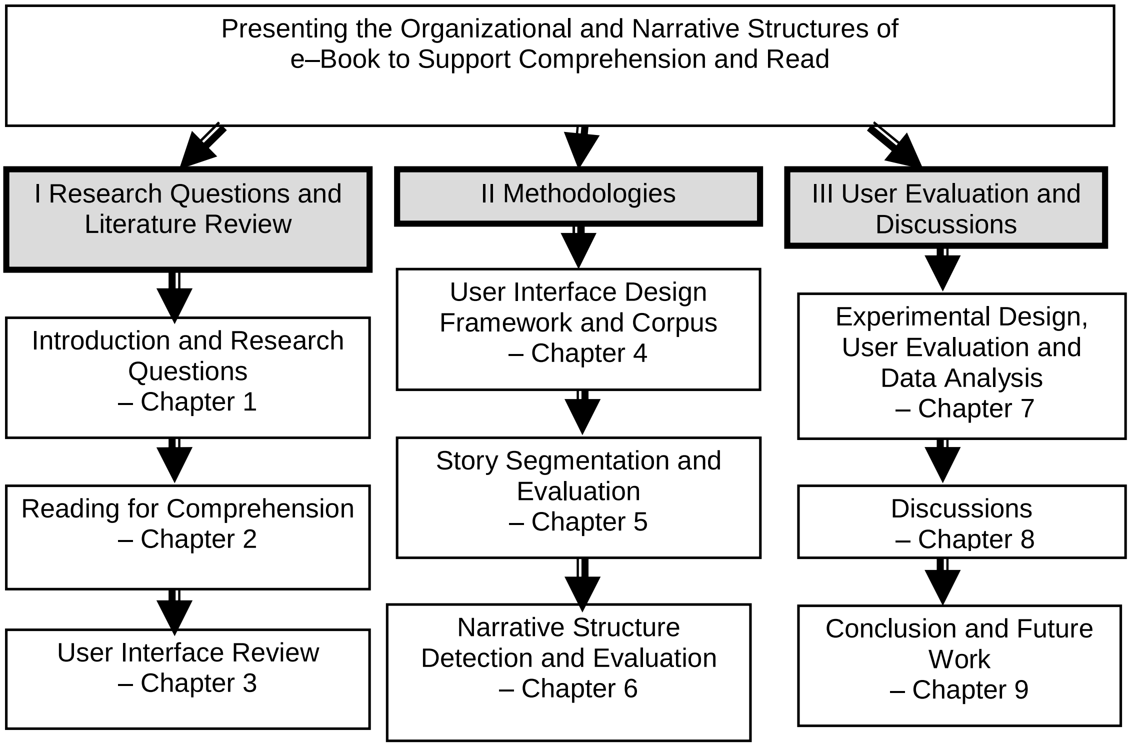

Presenting the Organizational and Narrative Structures of e–Book to Support Comprehension and Read
I Research Questions and Literature Review
II Methodologies
III User Evaluation and Discussions
Introduction and Research Questions – Chapter 1
Reading for Comprehension – Chapter 2
User Interface Review – Chapter 3
User Interface Design Framework and Corpus – Chapter 4
Story Segmentation and Evaluation – Chapter 5
Narrative Structure Detection and Evaluation – Chapter 6
Experimental Design, User Evaluation and Data Analysis – Chapter 7
Discussions – Chapter 8
Conclusion and Future Work – Chapter 9

Figure 3 Thesis Map

2

READING FOR COMPREHENSION: A LITERATURE REVIEW

2.1 Reading – Comprehension

2.1.1 Text Comprehension

Over the past 40 years the research body on text comprehension has been growing rapidly, especially in the fields of psychology and education. The goals of text comprehension research are to understand what aspects of the reader and the text influence the comprehension of a text, and to make predictions of the readability of the text. Through modelling the text and the reader’s knowledge and abilities, researchers have been able to develop both better texts and a better understanding of the human comprehension processes (Foltz, 1996).

As Foltz pointed out, research on text comprehension has examined a variety of factors that influence comprehension. These factors include: the role of coherence and readability in a text (Kintsch and Vipond 1979, Foltz, 1996), the role of the contextual structure of the text (Chung 2000; Sanchez, Lorch et al. 2001), the role of readers’ background knowledge (Britton and Gulgoz 1991; van Dijk and Kintsch

1983), and the role of the narrative schema of the text (Black and Bower 1979). Quite often these factors interact with each other in comprehension and it is hard to say which one is most important and influential.

In Kintsch (1998) model, texts can be viewed in terms of three different levels that support and pose challenges to readers: *word level, sentence level, and passage level*. The word level consists of the words on the page as individual units that have to be recognized. The sentence level combines these words using rules of syntax and semantic structure to form idea units. The passage level relates the network of connections among the ideas, or the extent to which the ideas 'hang together' in a cohesive manner. All three levels provide context and structure to the reader while reading for comprehension.

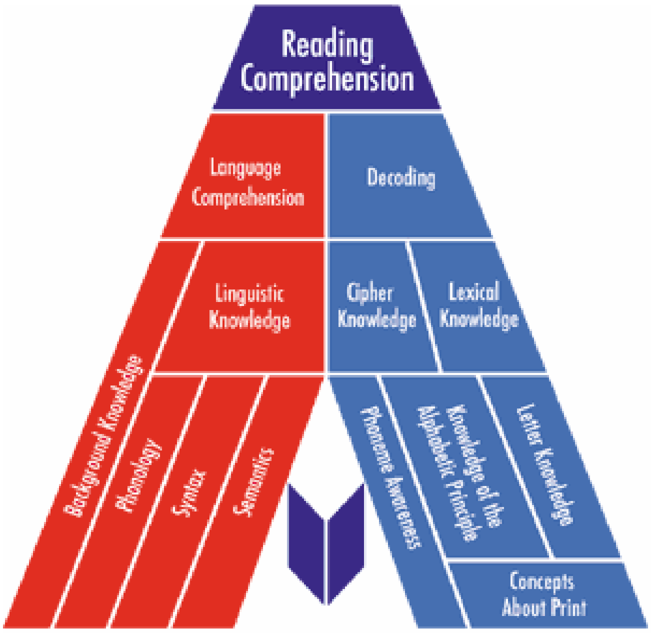


Figure 4 Tower of Comprehension (Wren, Litke et al. 2000)

In Hoover and Gough (1990)'s cognition framework (Figure 4), reading comprehension is the ability to construct linguistic meaning from written representations of language. This ability is based upon two equally important competencies. One is language comprehension – the ability to construct meaning from spoken representations of language; the second is decoding – the ability to recognize written representations of words. These two main foundations of reading are represented by the two supporting legs in Figure 4.

Experiments on text processing show that readers respond to a text in radically different ways depending on their own knowledge and interests. Foltz (1996) pointed out that experienced readers tend to have better skills than novices at exploiting context cues and other textual constraints. They are able to make better hypotheses about the meaning of words (Perfetti and Roth 1981), and are more responsive to the rhetorical structure of the text (Meyer, Brandt et al. 1980). On the other hand, less experienced readers' decoding skills are not as effective, so instead they compensate by using context-dependent hypotheses testing. In this manner, if contextual cues are missing or are confusing, then the performance of poor readers will be degraded to a greater extent than experts. Both these researches prove a strong demand for providing contextual and structure information of e-Book to assist reading for comprehension.

2.1.2 Textual Organizers in Comprehension

Numerous studies have found facilitative effects of textual organizers, or structural information in text comprehension.

A study by Chung (2000) describes an investigation of the effects of logical connectives and paragraph headings on reading comprehension among 577 Hong Kong secondary-six students who learn English as a second language. An English reading comprehension test was used to allocate subjects into one of the three performance groups: High, Medium and Low. The test instruments used to discriminate between the different groups contained ‘normal’ signals. In the signal studies, four versions of authentic text were produced. Version 1 was a non-signalled passage. Versions 2, 3, and 4 were embedded respectively with logical connectives, paragraph headings and these two signals in combination. All four versions had the same content and the same level of difficulty. Results show that those poorest in reading comprehension benefited from signals during the reading. All signals contributed to reading comprehension except for logical connectives, which did not aid microstructure understanding.

Sanchez, Lorch and Lorch (2001) reported how headings influence readers’ memories for text content. College students read and recalled a 12-topic expository text. Half of the participants were trained to construct a mental outline of the text’s topic structure as they read and then use their mental outlines to guide their recall attempts. The remaining participants did not receive such training. Half of the participants read a text containing headings before every subsection; the other half read the same text without headings. The results were that participants who received training and/or read the text with headings remembered text topics and their organization better than participants who received no training and read the text without headings. The results support the hypothesis that signals induce a change in readers’ strategies for encoding and recalling text.

These findings are consistent with the hypotheses reported earlier by Lorch and Lorch (1995). In their study, college students read and recalled a text that contained either no signals or contained headings, overviews, or summaries emphasizing the text’s topic structure. At recall, students either received no cues or were reminded of the text’s topics. Providing cues facilitated recall much more in the three conditions involving signalling than in the no-signals condition. The results confirmed that organizational signals induce readers to change their text-processing strategies.

2.1.3 Comprehension of Hypertext

Hypertext presents a new way to read online text that differs from read a paper book. In hypertext, information can be represented in a semantic network in which multiple related sections of the text can be dynamically linked to one another. As hypertext systems become popular, various approaches have been made to converting existing documents and paper book into hypertext form. For example, SuperBook (Egan, Remde et al. 1989), TACHIR (Agosti, Crestani et al. 1995), and Hyper-TextBook (Crestani and Ntioudis 2001). In the approach of the development of hypertext, models from information retrieval have been applied in order to determine how to structure the information, i.e., probabilistic models of retrieval (Croft and Turtle 1989).

The flexibility and non-linearity of hypertext systems, attributes that seem to hold great promise, have also been viewed as causing confusion and disorientation with users not being able to figure out where they are and where they should go next (Marchionini 1995). These difficulties have been summarized as ‘lost in hyperspace’ phenomenon (Edward and Hardman 1989).

Various efforts have been made to improve readers’ navigation and comprehension of hypertext. E.g., using navigation aids (Park and Kim 2000), table of contents

(Egan, Remde et al. 1989), dynamic index (Chi, Hong et al. 2004) and concept map (Puntambekar, Stylianou et al. 2003), etc. More details of the user interface aspects of these approaches will be reviewed in the next chapter. However, neither a general theory of hypertext nor a model of the cognitive process involved in reading exists (Altun, 2000). Additionally, little research has been done by reading researchers and educators to assess hypertext's potential impact on and implications for reading and literacy education (Altun, 2003, Protopsaltis and Bouki, 2005). There is not yet a general comprehension model of hypertext; researchers have been trying to adapt the linear text comprehension models, for example, in Protopsaltis and Bouki (2005), Folts (1996).

Potelle and Rouet (Potelle and Rouet 2003) investigated the effects of content representation devices and readers' prior knowledge on the comprehension of an expository hypertext. Forty seven students with low or high prior knowledge in Social Psychology were asked to read a hypertext using one of three content representations: a hierarchical map, a network map and an alphabetic list. Then, the participants performed a multiple choice comprehension task, a summary task and a concept map drawing task. The hierarchical map improved comprehension for the low knowledge participants at the global, but not at the local level. There was no effect of content representation on the comprehension of high prior knowledge students.

Although current technologies allow designers to draw fancy nonlinear interactive content representations for hypertext system, growing evidence shows that in this area more is not necessarily better. Simpler navigation tools prove more useful to the learning of low-knowledge users. More sophisticated representations, such as concept maps displaying various types of semantic links, may be appropriate only for high knowledge users.

2.2 Learning and Problem Solving

2.2.1 Dynamic Memory

Schank & Abelson (Schank and Abelson 1977) introduced the concepts of scripts, plans and themes to handle story-level understanding. The central focus of Schank's theory has been the structure of knowledge, especially in the context of language understanding. The premise in Schank's (1982) dynamic memory model, is that remembering, understanding, experiencing, and learning cannot be separated from each other. A dynamic memory understands by attempting to find the closest thing in memory to what it is trying to understand and then adapting its understanding of the old item to fit the new one. In the course of understanding, old experiences are remembered, providing expectations that further drive the understanding process. Understanding, in turn, allows memory to recognize and refine itself, in short, to be dynamic. This understanding and learning cycle is not merely a process for language understanding. It is the process that drives our reasoning – knowing where something fits in with what we already know is a prerequisite to reasoning about it.

Schank and Cleary (Schank 1982a; Schank and Cleary 1995) proposed how an organized memory might be built out of knowledge structures such as scripts and plans. In order to be integrated into a memory, the earlier structures needed to be altered. The major knowledge structure that resulted was called the MOP, for ‘Memory Organization Packet’.

In (Schank 1986), all memory is episodic, i.e., organized around personal experiences rather than semantic categories. Generalized episodes are called scripts – specific memories are stored as pointers to scripts plus any unique events for a particular episode. Scripts allow individuals to make inferences needed for understanding by filling in missing information (i.e., schema).

Schank (1986) uses these scripts as the basis for a dynamic model of memory. This model suggested that events are understood in terms of scripts, plans and other knowledge structures as well as relevant previous experiences. An important aspect of dynamic memory is the set of explanatory processes (XPs) that represent stereotypical answers to events that involve analomies or unusual events. Schank proposes that XPs are a critical mechanism of creativity. Case-based explanation builds new explanations by retrieving stored explanations for previous episodes and adapting them to fit current circumstances and needs.

2.2.2 Case Based Reasoning

In Schank (1982), he described how computers could learn based upon what was known about how people learn. This led to the foundational theories of case-based reasoning and the role of social scripts and stories in the learning process. Learning in case based reasoning occurs as a natural byproduct of problem solving. When a problem is successfully solved, the experience is retained in order to solve similar problems in the future. When an attempt to solve a problem fails, the reason for the failure is identified and remembered in order to avoid the same mistake in the future.

What is case-based reasoning? Basically: To solve a new problem by remembering a previous similar situation and by reusing information and knowledge of that situation (Aamodt and Plaza 1994).

Case-Based Reasoning (CBR) favours learning from experience, since it is usually easier to learn by retaining a concrete problem solving experience than to generalize from it. Still, effective learning in CBR requires a well worked out set of methods in order to extract relevant knowledge from the experience, integrate a case into an existing knowledge structure, and index the case for later matching with similar cases.

2.2.3 Concept Mapping

The concept mapping is another way to approach learning and problem solving. The technique of concept mapping was developed by Novak in the 1970s, as a way to increase meaningful learning in the sciences. Novak's work is based on the theories of Ausubel, who stressed the importance of prior knowledge in being able to learn new concepts. *"The most important single factor influencing learning is what the learner already knows."* (Ausubel 1968). Concept mapping is considered a constructivist learning activity (Novak 1998). Novak states that *"meaningful learning involves the assimilation of new concepts and propositions into existing cognitive structures"* (Novak and Gowin 1984).

Concept mapping is a technique for visualizing the relationships between different concepts. A concept map is a diagram showing the relationships between concepts. The relationship between concepts is articulated in linking phrases, e.g., 'gives rise to', 'results in', 'is required by', or 'contributes to'.

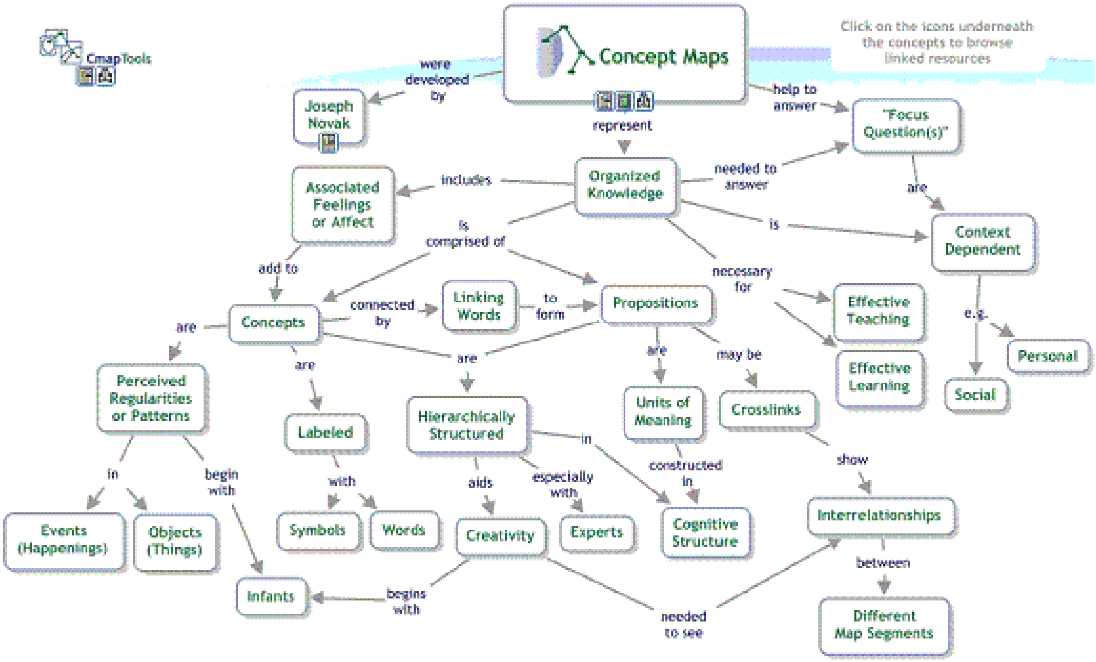


Figure 5 Concept Map of Concept Map (e-Resource 2007)

Concept maps have their origin in the learning approach called constructivism. In particular, constructivists hold that prior knowledge is used as a framework for understanding and learning new knowledge. This is same as case-based reasoning.

2.2.4 Directed Reading and Thinking

"Tell me, and I will forget. Show me, and I may remember. Involve me, and I will understand." (Confucius circa 450 BC). Developed by Stauffer (Stauffer 1969), the Directed Reading/Thinking Activity (DR-TA) is a group comprehension activity that features prediction of the story events prior to reading, reading to prove or modify predictions, and the use of divergent thinking.

The key step in a DR-TA is developing purposes for reading. Purposes or questions represent the directional and motivating influences that get readers started, keep them on course, and produce the vigor, potency and push to carry them through to the end. The DR-TA engages students in a step-by-step process that guides them through informational text. It is designed to move students through the process of reading text. While the teacher guides the process, the student determines the purpose for reading.

To introduce this strategy, the teacher gives examples of how to make predictions. A preview of the section to be read is given by having the students read the title (and perhaps a bit of the text) and make predictions about contents. This encourages independent thinking as knowledge from previous lessons is incorporated into the predictions. Follow-up activities may be completed after the text is read.

The DR-TA provided scaffolding to assist reading by purposefully using the story title as predictions. This approach could be adapted in the e-Book user interface design, where the system plays the role of the teacher. Using the same concept, we designed simulated semi-directed reading tasks for user evaluation. Readers of e-Book were given tasks to identify multiple themes of a story. To help them, we provide an e-Book user interface that firstly lists stories in the book, and secondly links similar stories. Users will be able to identify the common theme of a thread of similar stories, and then identify multiple themes of the original story.

2.3 Evaluation of Comprehension

2.3.1 Learning Outcome

Learning outcomes are what result from a learning process. They are commonly used in educational practice. They are specific measurable achievements and are stated as achievements of the student. Learning outcomes have been used as measures in learning system evaluations by some researchers.

Kai Halttunen & Kalervo Järvelin reported a study of assessing learning outcomes in an experimental, but naturalistic, learning environment compared to more traditional instruction, on the Information Retrieval subject.

Fifty seven participants of an introductory course on IR were selected for this study, and the analysis illustrated their learning outcomes regarding both conceptual change and development of IR skill. Concept mapping of student essays was used to analyze conceptual change and log files of search exercises provided data for performance assessment. Students in the experimental learning environment changed their conceptions more regarding linguistic aspects of IR and put more emphasis on planning and management of search process.

Puntambekar and Stylianou (Puntambekar, Stylianou et al. 2003) also used learning outcomes to evaluate their hypertext learning system CoMPASS (Concept Mapped Project-based Activity Scaffolding System). See the review of system in next chapter for more details and Figure 14, page 45 for a screenshot.

2.3.2 Bloom and Anderson's Taxonomy

In 1956, Bloom headed a group of educational psychologists who developed a classification of levels of intellectual behaviour important in learning. This let to Taxonomy of Educational Objectives, created by Bloom as a means of expressing qualitatively different kinds of thinking (Bloom 1956). Bloom's Taxonomy has since been adapted for classroom use as a planning tool and continues to be one of the most universally applied models across all levels of schooling and in all areas of study.

Bloom identified six levels within the cognitive domain, ranging from simple recall or recognition of facts as the lowest level, through increasingly more complex and abstract mental levels, to the highest level which was identified as evaluation (Figure 6). Bloom found that over 95 % of the test questions students encounter require them to think only at the lowest possible level – the recall of information.

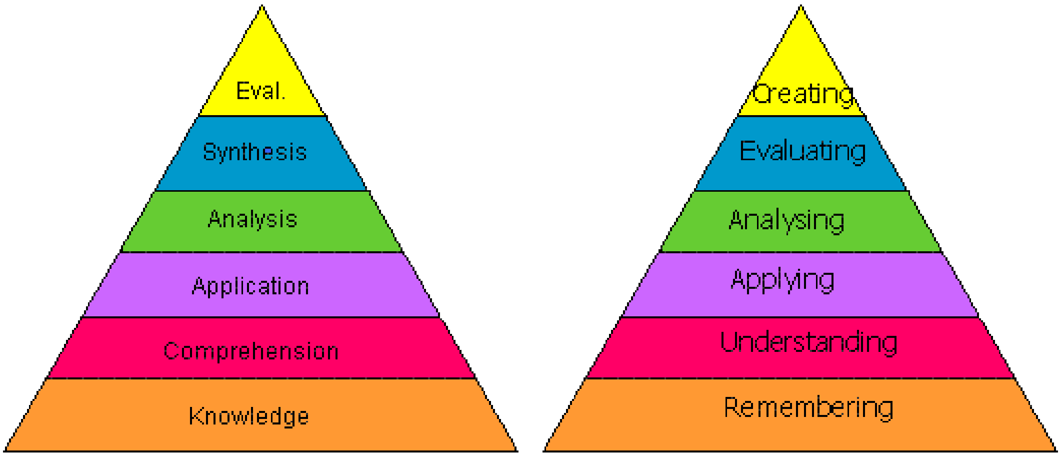


Figure 6 Bloom's (left) and Anderson's Taxonomy (right) (Schultz 2005)

During the 1990's, Anderson (a former student of Bloom) led a team of cognitive psychologists to revisit the taxonomy with the view to examining the relevance of the taxonomy at the beginning of the twenty-first century. As a result of the investigation a number of significant improvements were made to Bloom's original structure (Anderson and Krathwohl 2001). For example, the title of each level is changed from nouns to verbs; the 'synthesis' in the higher level is replaced by 'evaluating', and 'evaluation' on the top is replaced by 'creating'. Table 1 describes the 'new' taxonomies:

Table 1 Definitions of Anderson's Revised Taxonomy

Definition	Verbs
Remembering: can the student recall or remember the information?	Define, duplicate, list, memorize, recall, repeat, reproduce, state
Understanding: can the student explain ideas or concepts?	Classify, describe, discuss, explain, identify, locate, recognize, report, select, translate, paraphrase
Applying: can the student use the information in a new way?	Choose, demonstrate, dramatize, employ, illustrate, interpret, operate, schedule, sketch, solve, use, write
Analysing: can the student distinguish between the different parts?	Appraise, compare, contrast, criticize, differentiate, discriminate, distinguish, examine, experiment, question, test
Evaluating: can the student justify a stand or decision?	Appraise, argue, defend, judge, select, support, value, evaluate
Creating: can the student create new product or point of view?	Assemble, construct, create, design, develop, formulate, write

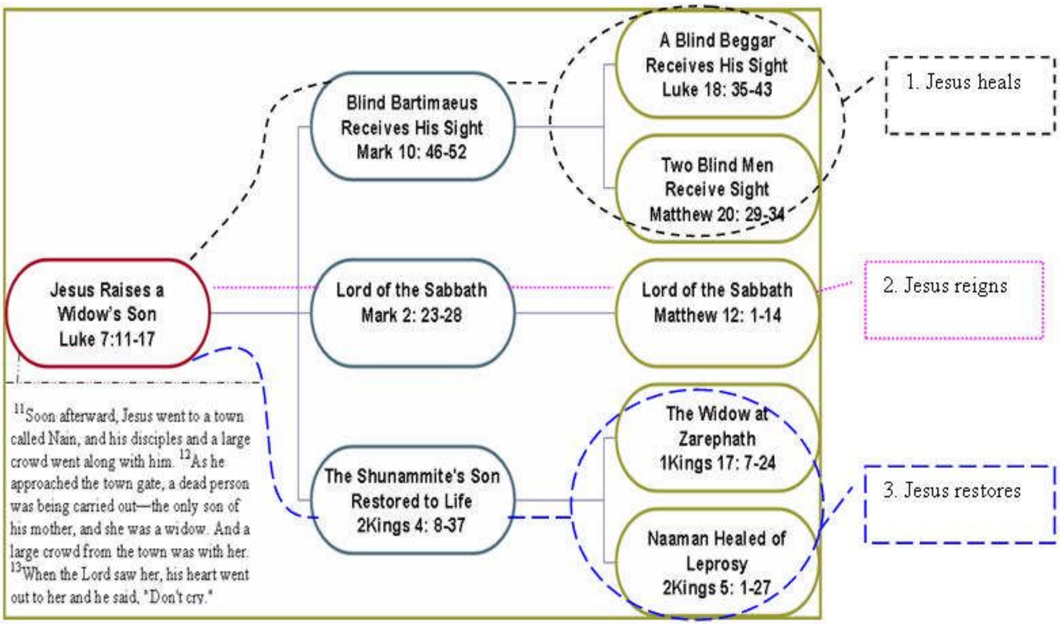
With a view to evaluate readers' comprehension of the narrative structure of an e-Book rather than its' language, we choose to focus on the middle levels of the taxonomy structure, which are: understanding, applying and analysing, and aim to develop evaluation methods that apply the suggested verbs in the right column in the above table (see below).

2.4 Using Story-Theme Mapping to Evaluate Comprehension

In this chapter we reviewed the theory of learning and the evaluation of learning. Research in the comprehension of text and hypertext provided strong evidence of the demand for structural information, contextual cues and contextual knowledge in reading; and provided practical experiences in modelling readers to examine the effects of these factors. These theories will help us to 'scaffold' our e-Book application in one way, and help us to design the evaluation and experiments in the other. In this study, we propose to combine these findings into a design framework of e-Book user interface, provide scaffolding such as a story linking tool and a navigation tool of e-Book, and finally support readers to gain a better understanding of the e-Book (see chapter 4).

Following the concept mapping concept, the directed reading and thinking method, and the Anderson's taxonomy, we designed a Story-Theme map and a set of corresponding evaluation methods for readers to draw on their finding of themes.

For example, a reader reading the story of *Jesus raises a widow's son* might well find many similar stories in the Christian Bible (see 4.2.2):


A Blind Beggar Receives His Sight Luke 18: 35-43
1. Jesus heals
Blind Bartimaeus Receives His Sight Mark 10: 46-52
Two Blind Men Receive Sight Matthew 20: 29-34
Jesus Raises a Widow's Son Luke 7:11-17
Lord of the Sabbath Mark 2: 23-28
Lord of the Sabbath Matthew 12: 1-14
2. Jesus reigns
The Widow at Zarephath 1Kings 17: 7-24
The Shunammite's Son Restored to Life 2Kings 4: 8-37
3. Jesus restores
Naaman Healed of Leprosy 2Kings 5: 1-27

[11]Soon afterward, Jesus went to a town called Nain, and his disciples and a large crowd went along with him. [12]As he approached the town gate, a dead person was being carried out—the only son of his mother, and she was a widow. And a large crowd from the town was with her. [13]When the Lord saw her, his heart went out to her and he said, "Don't cry."

Figure 7 A Story-Theme Map Example

When similar stories are linked together with the dash lines, it is easier to identify multiple themes, i.e., *Jesus heals*, *Jesus reigns*, and *Jesus restores*. This process usually happens in a person's mind while they are reading. To record such a process on paper is an interesting task for reading and comprehension.

We designed reading tasks based on story-theme mapping to evaluate readers' comprehension of the stories they read. To identify a theme, a reader needs to find a common issue that is projected throughout the story. Usually she has to read a good portion of the story in order to do this. To construct a Story-Theme Map, a reader needs to have a focal point, a central story, which forms a starting point for other stories to be compared and thought about. During the reading task the reader has to *read* a number of stories, *understand* the themes of them, *compare* the different stories, *select the related* stories, *evaluate* the relevance of stories, and *identify* the major themes of stories. Verbs in italic reflect Anderson's taxonomy listed in Table 1, discussed in previous section.

More details of the reading tasks are reported in section 7.2.4 and 7.4. Appendix A3.7 and A3.14 lists two reading tasks used in the user evaluation.

In next chapter we will look into some real systems and their influences and limits on reading and learning behaviours.

3

E-BOOK USER INTERFACES: A SYSTEM REVIEW

3.1 Review of e-Book Design Guidelines

The development of hypertext systems has created research into how to design and use hypertexts. Much of this research has focused on computer science and interface issues rather than the cognitive aspects of hypertexts. Several books and articles have been published containing guidelines and rules for hypertext development (Martin 1990; Nielsen 1990). We started from a review of the EBONI (Electronic Books ON-screen Interface) e-Book design guidelines (Wilson and Landoni 2000), and then reviewed the state of art of e-Book user interfaces by comparing them with the user's demands. This review helps to clarify the working area of this research.

In the EBONI Electronic Textbook Design Guidelines, Wilson and Landoni (2000) summarised a list of baseline standards for the design of hypertext books. The guidelines were established to encourage the use of styles and techniques that have been most successful in terms of usability. The document covers two distinct areas – on-screen and hardware design guidelines. We ignore the hardware part in this study.

The on-screen design guidelines advised in three respects: providing an overview of the e-Book, formatting of the e-Book, and the readability or usability of the e-Book.

On the overview of the book, it advised to provide orientation clues (indications of a reader's place in the book) and content clues (abstracts, keywords, tables of contents and index).

On formatting and readability, the guideline advised: designing typographical aspects carefully; using short pages; choosing a readable font; using colour to create a consistent style and aid scan-ability.

The guidance on usability includes: providing book-marking and annotating functions; treating the book as a closed environment: no links to external sources unless clearly labelled; providing a searching tool.

The guidance also encouraged including reference recommendation and browsing using hypertext to enhance navigation and cross-referencing.

However, the EBONI guidelines, like other hypertext guidelines, concentrate on issues such as how much text should be contained in a node, what hypertext features to use, and how the information should be structured. These guidelines do not provide concrete design rules based on reading theory, but instead present abstract rules based more on common sense (Foltz, 1996).

3.2 Reading Behaviours

There are various reading behaviours for various purposes, for example, scanning, key words spotting, browsing, slow reading, analytical reading, critical reading, and so on. Below we briefly review the system approaches in supporting of the common reading behaviours.

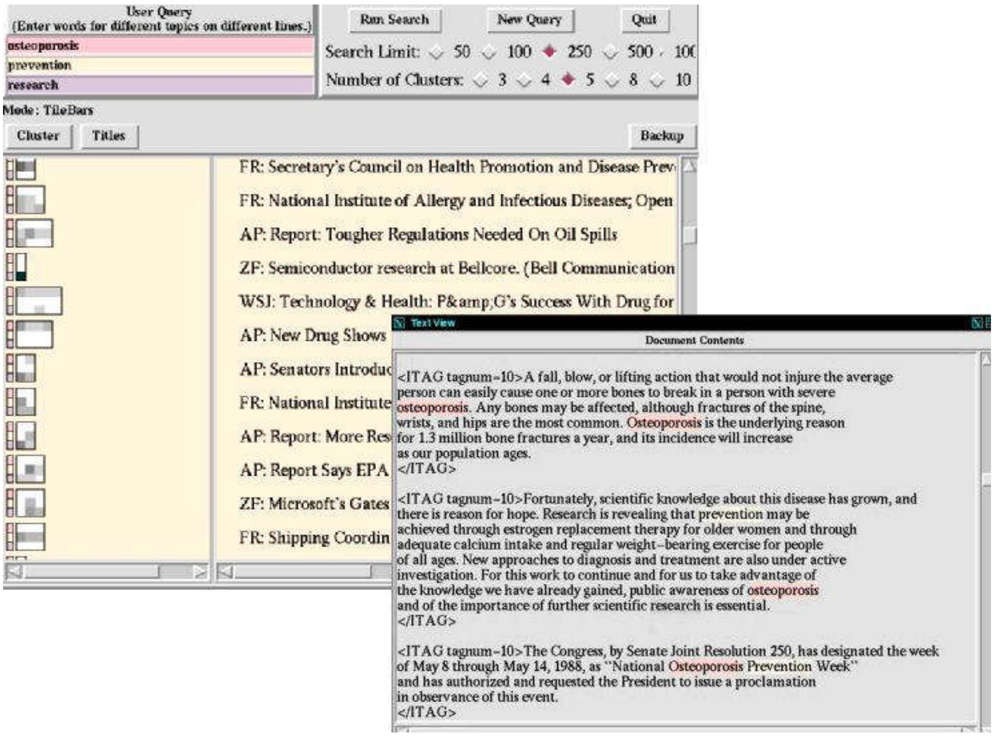
3.2.1 Scan: Reading to Find/Select Information

A scanning reading behaviour is often observed when the reader is looking for particular information. Internet search engines like Google and Yahoo, and online digital libraries like Gutenberg (Hart 1992), offer user interfaces to search, browse and select desired documents to view. People will, for example, scan the results from a search engine to select their targeted information. This kind of reading behaviour could be assisted by providing relevance profiles of documents, highlighting the query words, integrating overview and details of the documents, etc.

IR researchers have proposed many novel visualization methods to visualize the content of text documents as it relates to a query. For example, Byrd demonstrated using scrollbars in an interface that presents electronic documents with the distribution of query terms shown on the scrollbar using colour-coding (Byrd 1999). This design can support navigation in a document and aid users in gaining an overview of the distribution of query terms within the document. Other influential prior work is TileBars (Hearst 1995), and ProfileSkim (Harper, Coulthard et al. 2002; Harper, Koychev et al. 2003a; Harper, Koychev et al. 2003b).

TileBars (Figure 8) provides a compact and informative iconic representation of the documents’ contents with respect to the query terms. TileBars allow users to make informed decisions about not only which documents to view, but also which passages of those documents, based on the distributional behaviour of the query terms in the documents. The shaded squares, called as TileBars (see the left column of the figure

8), indicate the relative length of the document, the frequency of the term sets in the document, and the distribution of the term sets with respect to the document and to each other.



User Query
(Enter words for different topics on different lines.)
osteoporosis
prevention
research

Search Limit: ○ 50 ○ 100 ● 250 ○ 500 ○ 10(
Number of Clusters: ○ 3 ○ 4 ● 5 ○ 8 ○ 10

Mode: TileBars

FR: Secretary's Council on Health Promotion and Disease Prev
FR: National Institute of Allergy and Infectious Diseases; Open
AP: Report: Tougher Regulations Needed On Oil Spills
ZF: Semiconductor research at Bellcore. (Bell Communication
WSJ: Technology & Health: P&amp;G's Success With Drug for
AP: New Drug Shows
AP: Senators Introduc
FR: National Institute
AP: Report: More Res
AP: Report Says EPA
ZF: Microsoft's Gates
FR: Shipping Coordin

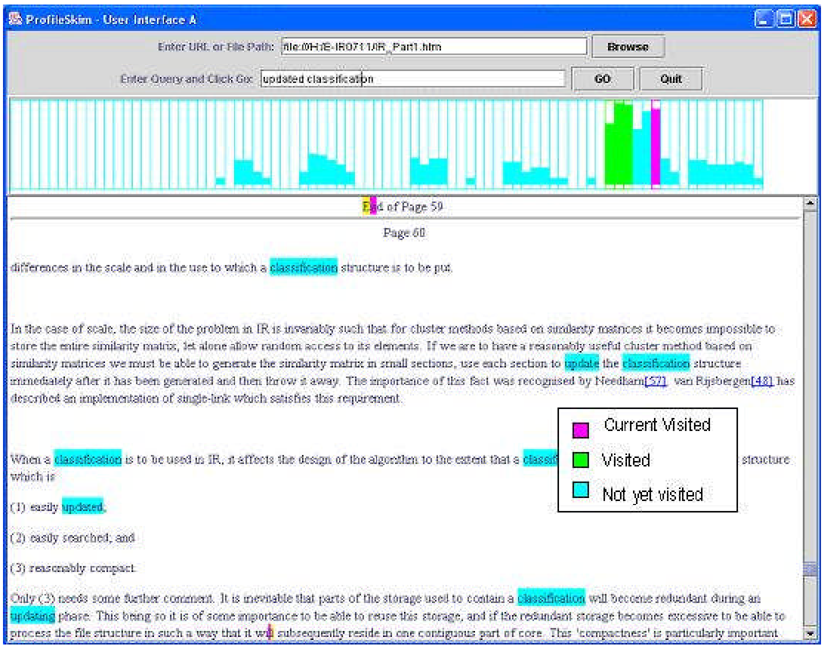
<ITAG tagnum=10>A fall, blow, or lifting action that would not injure the average
person can easily cause one or more bones to break in a person with severe
osteoporosis. Any bones may be affected, although fractures of the spine,
wrists, and hips are the most common. Osteoporosis is the underlying reason
for 1.3 million bone fractures a year, and its incidence will increase
as our population ages.
</ITAG>

<ITAG tagnum=10>Fortunately, scientific knowledge about this disease has grown, and
there is reason for hope. Research is revealing that prevention may be
achieved through estrogen replacement therapy for older women and through
adequate calcium intake and regular weight–bearing exercise for people
of all ages. New approaches to diagnosis and treatment are also under active
investigation. For this work to continue and for us to take advantage of
the knowledge we have already gained, public awareness of osteoporosis
and of the importance of further scientific research is essential.
</ITAG>

<ITAG tagnum=10>The Congress, by Senate Joint Resolution 250, has designated the week
of May 8 through May 14, 1988, as "National Osteoporosis Prevention Week"
and has authorized and requested the President to issue a proclamation
in observance of this event.
</ITAG>

Figure 8 Screenshot of TileBars Searching and Reading Windows

ProfileSkim (Figure 9) was developed by a group led by the author's supervisor, in which the author contributed as a researcher and developer. It adopted the approaches which were developed through TileBars and SCAN (Whittaker, Hirschberg et al. 1999) systems, to support content-based document-browsing using visualization techniques, and used passage retrieval (Kaszkiel 2000) and language modelling technology (Croft and Lafferty 2003; Ponte and Croft 1998) to develop its within-document search engine. The key concept underpinning the tool is relevance profiling, in which a profile of retrieval status values is computed across a document in response to a query. Within the user interface (see Figure 9), an interactive histogram graph provides an overview of this profile, and through interaction with the graph the user can select and browse *in situ* potentially relevant passages within the document. The language modelling basis for the relevance profiling provides a good indication of passage relevance, judging by the performance of language modelling when used in document retrieval.



file:/0H:/E-IR0711/IR_Part1.htm

updated classification

End of Page 59

Page 60

differences in the scale and in the use to which a classification structure is to be put.

In the case of scale, the size of the problem in IR is invariably such that for cluster methods based on similarity matrices it becomes impossible to store the entire similarity matrix, let alone allow random access to its elements. If we are to have a reasonably useful cluster method based on similarity matrices we must be able to generate the similarity matrix in small sections, use each section to update the classification structure immediately after it has been generated and then throw it away. The importance of this fact was recognised by Needham[57]. van Rijsbergen[48] has described an implementation of single-link which satisfies this requirement.

When a classification is to be used in IR, it affects the design of the algorithm to the extent that a classif structure which is

(1) easily updated;

(2) easily searched; and

(3) reasonably compact.

Only (3) needs some further comment. It is inevitable that parts of the storage used to contain a classification will become redundant during an updating phase. This being so it is of some importance to be able to reuse this storage, and if the redundant storage becomes excessive to be able to process the file structure in such a way that it will subsequently reside in one contiguous part of core. This 'compactness' is particularly important

Figure 9 Screenshot of ProfileSkim

A user-centred, task-oriented, comparative evaluation has been conducted in the evaluation of ProfileSkim. FindSkim, which provides similar functionality to the web browser’s ‘Find’ command, was developed as a control condition of ProfileSkim in the user experiments. Analysis of the user experiments confirmed that ProfileSkim was significantly more efficient than FindSkim, as measured by time needed to complete a benchmark task. The study also indicated that ProfileSkim was as least as effective as FindSkim in identifying relevant pages, as measured by traditional information retrieval measures. Based on qualitative data from questionnaires, there was strong evidence showing that the participants were more satisfied when using ProfileSkim than FindSkim.

Some interfaces for electronic documents show a graphical overview of the document separated from the detailed content of the document. For example, SeeSoft (Eick, Steffen et al. 1992) maps program source code into an overview by presenting one line of code by a thin row in the overview, colour-coded to display useful information, for example, the version, about the program. This is a good approach to show the overview of the document. But when there is complex information to present, the colour coding might be confusing and difficult to follow.

3.2.2 Read: Reading on the Screen

Research in hypertext and text display has produced hypotheses about how textual information should be displayed to users. One study of an on-line documentation system (Girill 1991) compares display of fine-grained portions of text (i.e., sentences), full texts, and intermediate-sized units. Girill found that divisions at the fine-grained level were less efficient to manage and less effective in delivering useful answers than intermediate-sized units of text. He also found that using document boundaries was more useful than ignoring document boundaries, as was done in some hypertext systems, and that pre-marked sectional information, if available and not too long, was an appropriate unit for display.

Tombaugh, Lickorish and Wright (Tombaugh, Lickorish et al. 1987) explored issues relating to ease of readability of long texts on CRT screens. Their study explored the usefulness of multiple windows for organizing the contents of long texts, hypothesizing that providing readers with spatial cues about the location of portions of previously read texts would aid in their recall of the information and their ability to locate quickly information that has already been read once.

Hornbaek and Frokjaer studied the usability of linear, fisheye and overview + detail interfaces (Hornbaek and Frokjaer 2001).

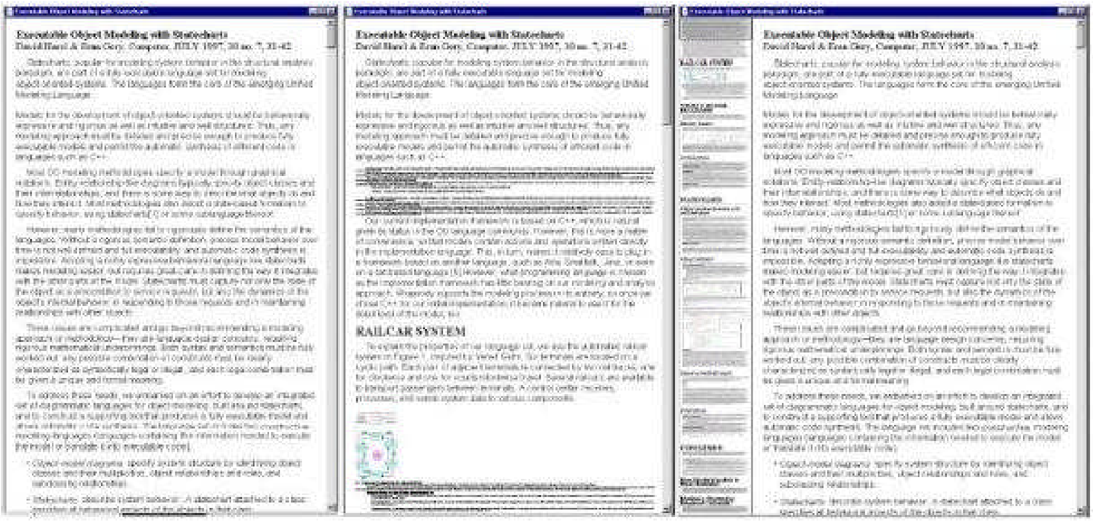

Executable Object Modeling with Statecharts
David Harel & Eran Gery, Computer, JULY 1997, 30 no. 7, 31-42
Executable Object Modeling with Statecharts
David Harel & Eran Gery, Computer, JULY 1997, 30 no. 7, 31-42
RAILCAR SYSTEM
Executable Object Modeling with Statecharts
David Harel & Eran Gery, Computer, JULY 1997, 30 no. 7, 31-42

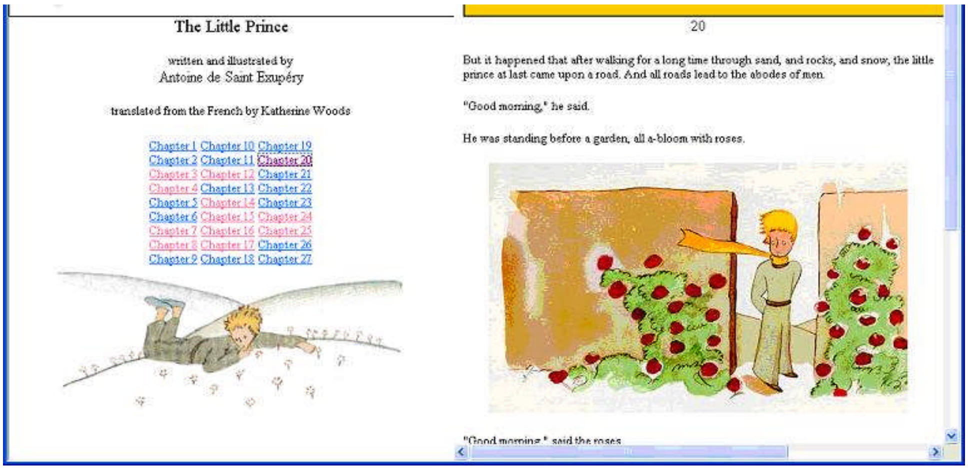
Figure 10 Linear, Fisheye and Overview Details Interfaces

‘Overview’ is used to describe a picture of the whole document collection, i.e., an e-Book. This could be a cover of the book, a table of content, or a set of index, an indicator of the total length of the book and the reader’s current location, in order to give the reader a quick idea about the size, content and the structure of the book.

They discovered that, using an overview + detail interface helped the reader to understand the documents better than using a linear or a fisheye interface.

3.2.3 Simple Navigation: Reading to Explore

With the development of hypertext system, hypertext environments are now increasingly being used in education. The flexibility and non-linearity of hypertext seem to be attractive, but it also easy to cause confusion to users. Users need to know where they are and where to go next.



The Little Prince

written and illustrated by
Antoine de Saint Exupéry

translated from the French by Katherine Woods

Chapter 1
Chapter 2
Chapter 3
Chapter 4
Chapter 5
Chapter 6
Chapter 7
Chapter 8
Chapter 9
Chapter 10
Chapter 11
Chapter 12
Chapter 13
Chapter 14
Chapter 15
Chapter 16
Chapter 17
Chapter 18
Chapter 19
Chapter 20
Chapter 21
Chapter 22
Chapter 23
Chapter 24
Chapter 25
Chapter 26
Chapter 27

20

But it happened that after walking for a long time through sand, and rocks, and snow, the little prince at last came upon a road. And all roads lead to the abodes of men.

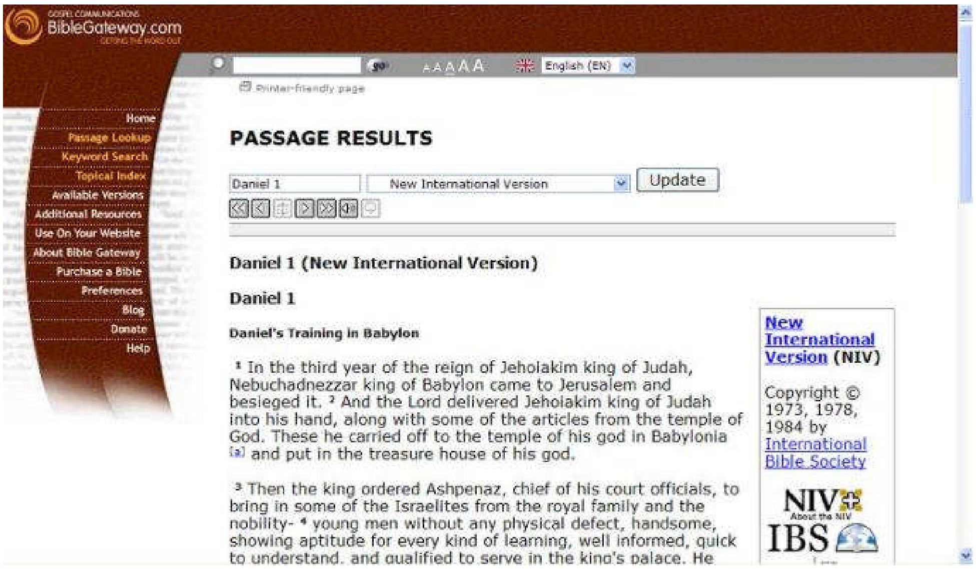
"Good morning," he said.

He was standing before a garden, all a-bloom with roses.

"Good morning," said the roses.

Figure 11 Screenshot of Hypertext Book *Little* Prince (Saint-Exupery 1943)

Most hypertext books and e-Book systems provide simple buttons and dynamic table of contents to support navigation within book chapters and pages. For example, in Figure 11, the left frame provides a list of chapters of the e-Book, and the right frame displays the content of a chosen chapter.



PASSAGE RESULTS

Daniel 1 | New International Version

Daniel 1 (New International Version)

Daniel 1

Daniel's Training in Babylon

1 In the third year of the reign of Jehoiakim king of Judah, Nebuchadnezzar king of Babylon came to Jerusalem and besieged it. 2 And the Lord delivered Jehoiakim king of Judah into his hand, along with some of the articles from the temple of God. These he carried off to the temple of his god in Babylonia [a] and put in the treasure house of his god.

3 Then the king ordered Ashpenaz, chief of his court officials, to bring in some of the Israelites from the royal family and the nobility- 4 young men without any physical defect, handsome, showing aptitude for every kind of learning, well informed, quick to understand, and qualified to serve in the king's palace. He

New International Version (NIV)

Copyright © 1973, 1978, 1984 by International Bible Society

Figure 12 Screenshot of BibleGateway's Hypertext Bible

The BibleGateway (e-Resource 1995) is a database driven online hypertext system (Figure 12). Readers can read, search, compare, and explore over 20 versions of Bible in many languages. A page can be displayed through keywords search or passage lookup. There is also a front page available with lists book titles and chapters. As with the Gutenberg (Hart 1992) online e-Books project, there is no overview or orientation information, i.e., table of contents in the browser window.

3.2.4 Comprehensive Navigation: Reading to Learn and Understand

Egan, Remde et al. (Egan, Remde et al. 1989) developed a dynamic table of contents (TOC) in the SuperBook project. The SuperBook provided a ‘fisheye’ style viewer of table of contents window that allows for the expansion and contraction of the chapter and section levels.

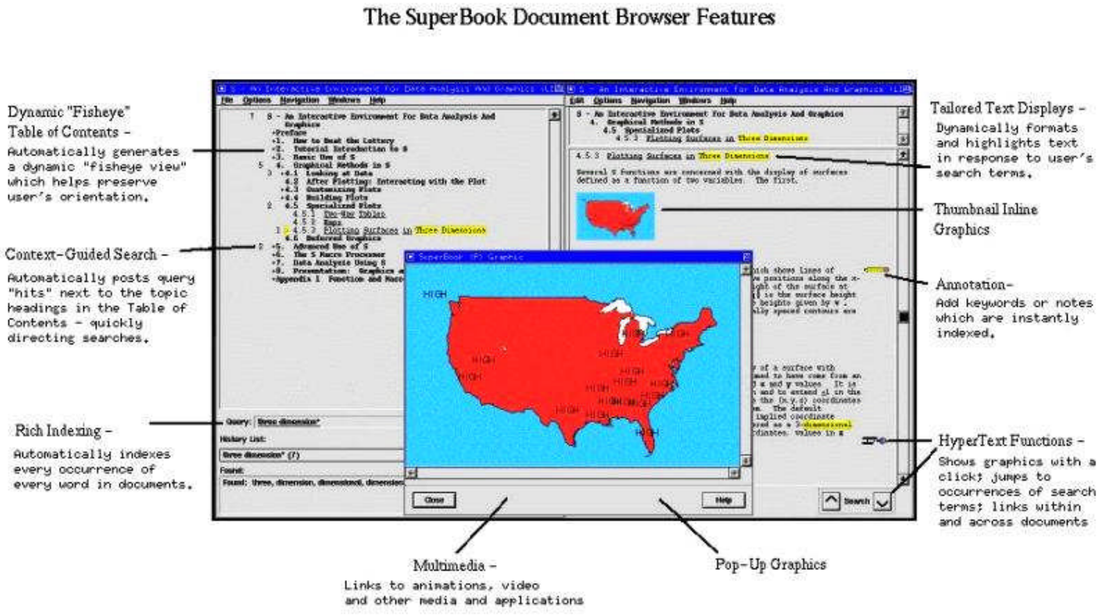
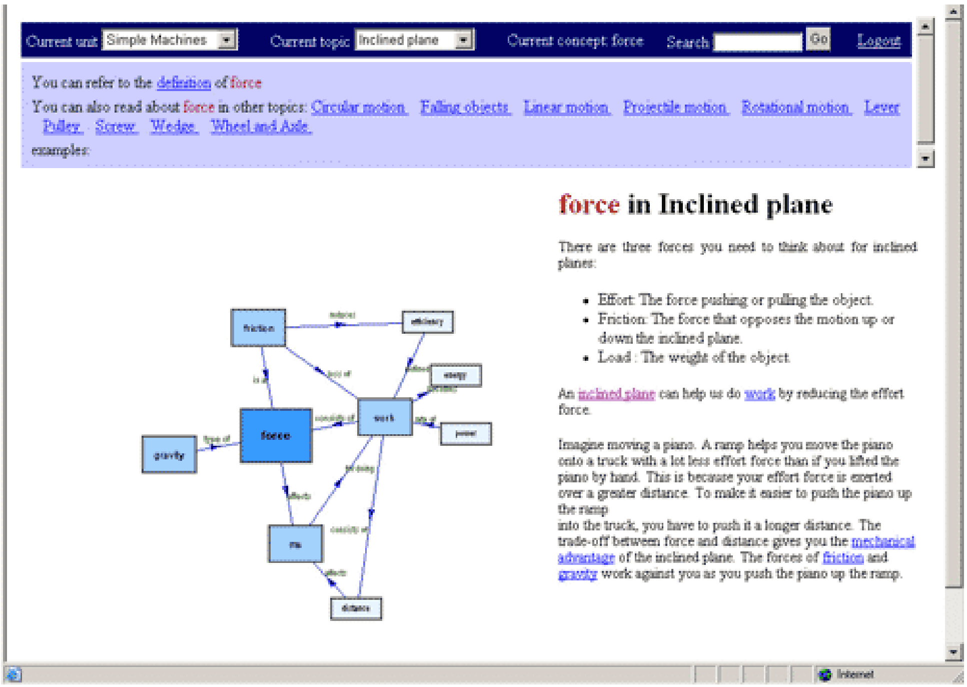

The SuperBook Document Browser Features
Dynamic "Fisheye" Table of Contents - Automatically generates a dynamic "fisheye view" which helps preserve user's orientation.
Context-Guided Search - Automatically posts query "hits" next to the topic headings in the Table of Contents - quickly directing searches.
Rich Indexing - Automatically indexes every occurrence of every word in documents.
Tailored Text Displays - Dynamically formats and highlights text in response to user's search terms.
Thumbnail Inline Graphics
Annotation- Add keywords or notes which are instantly indexed.
HyperText Functions - Shows graphics with a click; jumps to occurrences of search terms; links within and across documents
Multimedia - Links to animations, video and other media and applications
Pop-Up Graphics

Figure 13 Screenshot of SuperBook

The TOC (left column of the above figure) is hierarchically and dynamically presented, but not for the subject index. Its algorithm was based on term frequency.

The study of SuperBook showed that an expandable table of contents and a word lookup tool improved performance by 25% compared to searching in a paper manual.

Chi, Hong et al. (Chi, Hong et al. 2004) described another e-Book system, ScentIndex that enhanced the subject index of an e-Book by conceptually reorganizing it to suit particular information needs. Users first entered keywords describing the concepts they are trying to retrieve and learn. ScentIndex then computed those index entries that were conceptually related and displayed these index entries for the user. They demonstrated that the ScentIndex was faster and more accurate in assisting users with retrieving and comprehending information in the subject index of a book compared with a paper version.



You can refer to the definition of force

You can also read about force in other topics: Circular motion Falling objects Linear motion Projectile motion Rotational motion Lever Pulley Screw Wedge Wheel and Axle

examples:

force in Inclined plane

There are three forces you need to think about for inclined planes:

- Effort: The force pushing or pulling the object.
- Friction: The force that opposes the motion up or down the inclined plane.
- Load : The weight of the object.

An inclined plane can help us do work by reducing the effort force.

Imagine moving a piano. A ramp helps you move the piano onto a truck with a lot less effort force than if you lifted the piano by hand. This is because your effort force is exerted over a greater distance. To make it easier to push the piano up the ramp
into the truck, you have to push it a longer distance. The trade-off between force and distance gives you the mechanical advantage of the inclined plane. The forces of friction and gravity work against you as you push the piano up the ramp.

Figure 14 CoMPASS (Puntambekar 2005)

Puntambekar, Stylianou, et al. (Puntambekar, Stylianou et al. 2003) described a hypertext system CoMPASS (Concept Mapped Project-based Activity Scaffolding System, Figure 14) that presents students with external graphical representations in the form of concept maps as well as textual representations both of which change dynamically. In a study in which middle school students used CoMPASS to learn their science subject, they showed that the map helped students to stay focused on their goals, compared to an index version of the system. To test the effectiveness of the system, a pre-test and a post-test were conducted. Results showed that the CoMPASS system was more effective in helping students to understand the subject concepts. Also, students who used the index system did worse in the post-test compare to their pre-test.

Although a variety of hypertext systems have been developed over the past twenty years, research in hypertext has often failed to show any significant advantages for reading a hypertext compared to the equivalent text in linear form.

3.2.5 Other Features

Developments in summarization, reference recommendation, and annotation are other major influences on e-Book systems.

The Reader’s Helper (Graham 1999) analyzed documents and produced a relevance score for each of the reader’s topics of interest, thereby helping the reader decide whether the document was actually worth skimming or reading. Moreover, during the analysis process, topic-of-interest phrases were automatically annotated to help the reader quickly locate relevant information. A new information visualization tool, the Thumbar, was used in conjunction with relevancy scoring and automatic annotation to provide a continuous, dynamic thumb-nail representation of the document. In the Thumbar, a graphical overview of World Wide Web pages was shown next to the display of the page itself. Concepts in the user's profile were also highlighted both on the overview and on the web page.

Woodruff, Gossweiler et al. (Woodruff, Gossweiler et al. 2000) also presented a particular augmentation of digital books that provided readers with customized recommendations and introduced a preliminary user interface for the display of recommendations.

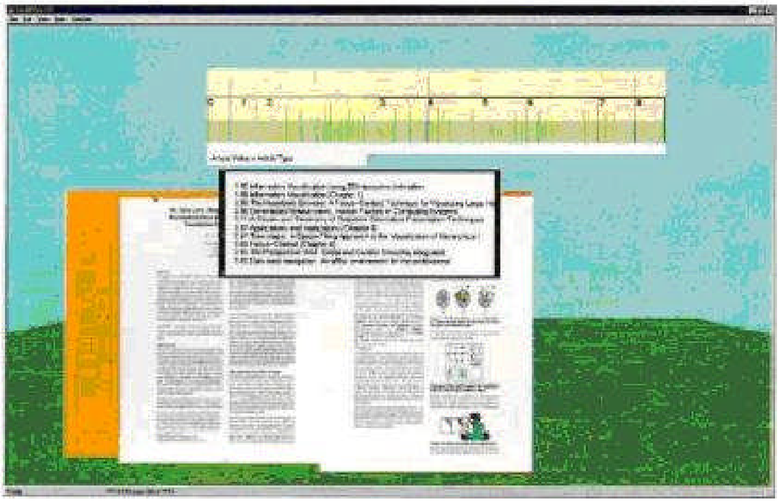


Figure 15 Book, Book Ruler and Recommendation List

They systematically explored the application of spreading activation over text and citation data to generate useful recommendations, and concluded that for the tasks performed in the chosen corpus, spreading activation over text was more useful than citation data. The fused spreading activation techniques outperformed traditional text-based retrieval methods in their study.

3.3 Summary

If in the past the researchers and designers of e-Book user interface have focused on the display, navigating and querying, now is the time to shift attention to supporting user understanding of the book contents. For this reason, the methods proposed in this thesis differ significantly from the prior work.

The intended work from this thesis is designed to support perusing e-Book readers and self-learners. Based on the analysis, the software will enable the user to browse and navigate e-Book text through two distinct interactive scaffolding tools, one for an overview visualization of the organizational structure, and the second to browse narrative threads. These scaffoldings will not only improve the usability of the software, but also help the reader to understand the organization of the book, to find the interesting relationship between main characters and stories, and to understand the main concepts quickly and easily.

Visualization is a good method to improve the usability of a system by the interactive interface. However, one thing we should remember, good music makes the audience forget the instruments, and a good movie makes the audience forget the cinema. Likewise, a well designed user interface is an ‘invisible’ user interface, it disappears to enable the user concentrate on their work, exploration or pleasure (Baeza-Yates and Ribeiro-Neto 1999; Shneiderman 1987).

II METHODOLOGIES

4

E-BOOK USER INTERFACE DESIGN FRAMEWORK

4.1 Design Framework

4.1.1 Presenting e-Book Structures to Support Comprehension

We reviewed the cognition theory of reading and the state of art of e-Book systems in chapter 2 and 3; recognized the gap between the text material to read and the comprehension process. Therefore, we aim to design e-Book software to bridge such a gap and improve comprehension.

In Figure 16, we present a model of user interface supported comprehension. In this model, the text of a book is processed and analysed by computer software; the structural information is extracted and presented in the software user interface; different functions and features are selected for presenting different structural information. This kind of user interface, when well equipped with information visualization to reveal the organizational and narrative threads structures, can help users to build their own mental model in order to understand book content such as stories and themes.

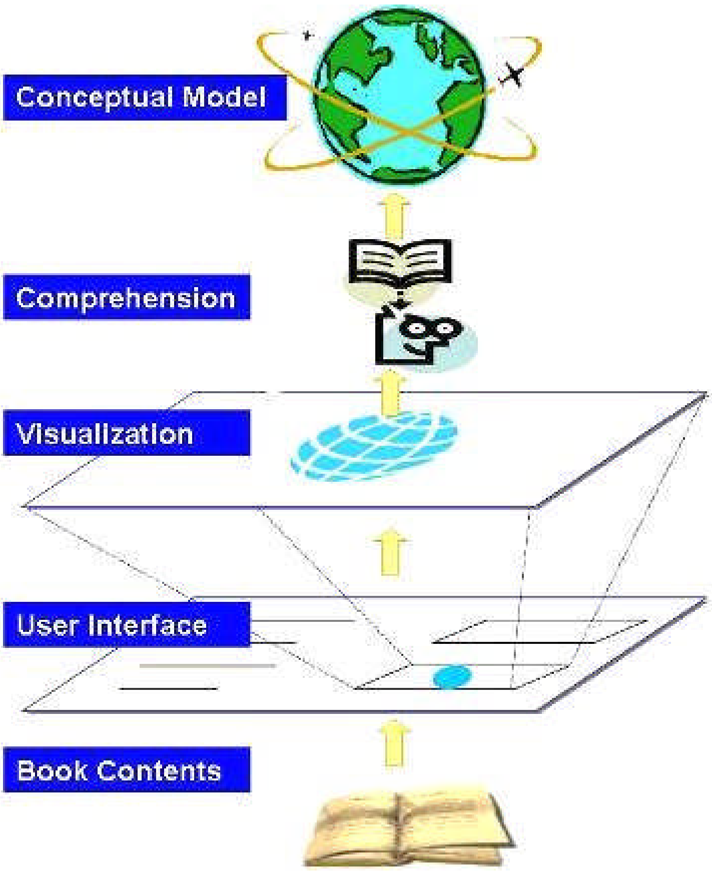


Figure 16 User Interface Supported Conceptual Model

4.1.2 Design Framework

We based our design of the user interface of an e-Book on the user scenarios introduced in chapter 1, and the review of other work in chapter 2 and 3.

Design framework is often used in software engineering process, by which the objectives of the application, the abstract classes, and the components are drawn together to support implementation and future reuse (Deutsch, 1989, Welsh et al. 2000). It is also used to guide user interface design and evaluation (Marchionini, 2000; Lee, 2002; Ingwersen and Jarvelin, 2005). We present a simple yet unique design framework for e-Book user interface in the Figure 17.

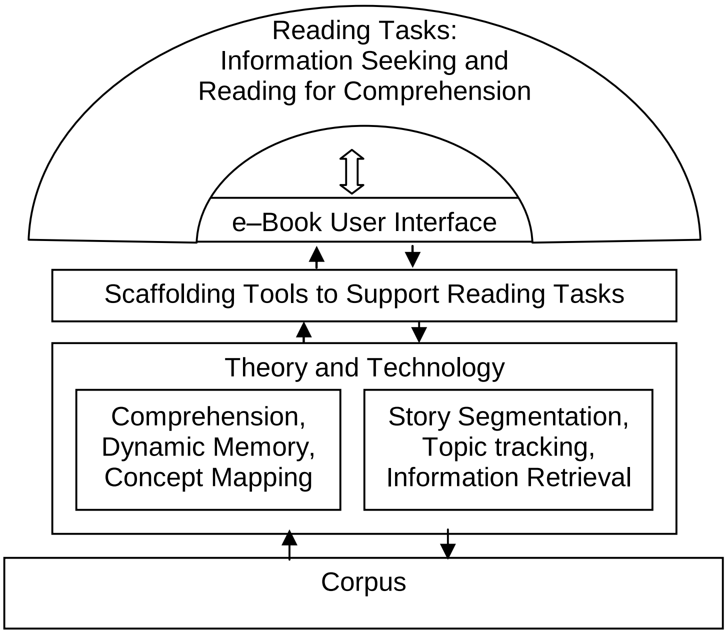

Reading Tasks:
Information Seeking and
Reading for Comprehension
e–Book User Interface
Scaffolding Tools to Support Reading Tasks
Theory and Technology
Comprehension, Dynamic Memory, Concept Mapping
Story Segmentation, Topic tracking, Information Retrieval
Corpus

Figure 17 Design Framework of e-Book User Interface

The purpose of this design framework is to allow us to design and provide many alternative types of scaffolding for different reading tasks. This framework enumerates possible design options for different categories. In summary they are:

- Reading tasks: considerations about different reading tasks in simulated real world situations. Design options in this category are scanning, reading for comprehension, conceptual exploration, etc
- Scaffolding: consideration about different details or granularity of the navigation tools to support reading tasks. Design options in this category range from simple navigation buttons, table of contents, visualization tools, to complex narrative threads detection and narrative structure browsing.

- Theory and technology: various scaffolding approaches should be built upon the cognition theory and latest development in e-Book application.
- Corpus: comprehensive and complex text should be chosen in supporting the reading tasks.

We consider the readers' need the most important part in a software design process, especially the demanding support of the reading for comprehension, so it occupies the top position in the architecture. The user interface is the medium between user and the software; the user can communicate with the software through a well-designed user interface, which is powered by visualization techniques. This user interface and visualization are supported by information retrieval search engines, including a within document retrieval search engine, a text processing engine for topic tracking, etc.

Various ideas of scaffolding were applied in the system design. Scaffolding refers to the different kinds of supports that readers receive in their interaction with the e-Book. These are built upon the e-Book corpus foundation.

Many hypertext systems share in common the problem of 'lost in hyperspace' (Edward, D. M. and Hardman, L., 1989). The readers of hypertext book have little clue of orientation information and structural information of the many nodes available, and have difficulty in deciding where to go next (2.1.3). Presenting readers with multiple structural information of a text can not only help them to have a better direction, but also to have a better understanding of the text (2.1.2).

4.1.3 Software Implementation Procedure

The e-Book system developed to investigate the issues raised by this thesis have the important tasks of discovering and presenting the e-Book narrative structure, in order to reveal more contextual information to the readers. The e-Book software is entirely written in Java, except where specifically indicated. There are four main aspects to the implementation of the software and interface:

- Firstly, processing the texts and discovering basic organizational structure of the e-book (discussed in this chapter);
- Secondly, partitioning the e-Book texts into basic narrative segments based on topics and stories (chapter 5);
- Thirdly, semi-automatically detecting narrative structure of the e-Book by linking up similar narrative segments (chapter 6); and
- Finally, presenting the narrative structure information through the user interface of the e-Book (this chapter).

These four aspects are supported by a few other common components, for example, generating an index of stories from resources, and term indexing, which include:

- Tokenizing the texts
- Identifying the stop-words
- Identifying the character names
- Conflation and stemming, using Porter’s algorithm (Porter 1980)

- Creating 'inverted file' structure for the text (Baeza-Yates and Ribeiro-Neto 1999; Frakes and Baeza-Yates 1992).

Following the design guidelines of an e-Book (Wilson and Landoni 2000), we decided to use the chapter as the main unit for the e-Book viewing and browsing, as it is of reasonable length and is a central part of the book structure.

We have designed this e-book for typical perusing book readers. The main goals of the design are to help the users to know '*where am I*', '*what are the structures*', '*where are the stories*', '*who are the characters*', '*which stories are related to this story*', and '*what are the themes*'.

We used storyboard designs on paper to form our earlier user interface prototypes, and asked readers of e-Books for their feedback through informal discussions and brain storming. We did not report this process in detail in this thesis, but we listed the storyboard designs in Appendix A1.

Our user scenarios in chapter 1, and the storyboards for user interface design as listed in Appendix A1, both point to a graphical user interface to reveal thematically structures of a book. However, in the study we did not present such system design, since it would be similar to the Concept-Mapping systems (Puntambekar 2005). We have rather focused on the narrative structure detection and using aspects of e-Books. In evaluation, we asked readers to draw a Story-Theme Map in their reading task. Results showed that readers like the idea of mapping the related stories with the relevant themes.

The collection of drawings by those participants would be a good resource for our future development, towards automatic construction of Story-Theme Maps in an e-

Book. Research in ontology engineering (Agirre, Ansa et al. 2000; Resnik 1999; Stevenson 2002) might provide insights and solutions for this problem in future development.

4.2 Collections and Other Resources

4.2.1 Collection Characteristics

In *The Turn*, Ingwersen and Jarvelin (2005, p8) argued that laboratory based evaluation in IR is often blamed for lack of variety in collections. "*The test collections, albeit nowadays large, are structurally simple (mainly unstructured text) and topically narrow (mainly news domain)*". This raised the issue of exploring new collections that are previously undeveloped.

To address these concerns, we have chosen our corpus carefully considering the research questions. Such a corpus must meet the following requirements:

- It must be significant in size, structure and contents;
- It must be of importance, interesting and popular;
- It must have meaningful narrative structures to provide the necessary information for automatic or semi-automatic narrative threads generalization;
- It must provide a ground for comprehension and read activities;
- Its language must be easily understood by people from various backgrounds, to support user-centred system design and further evaluation; and,
- Its electronic resource and relevant material must be easily accessible for research purposes.

In Table 2 we compared a few different book types, i.e., academic text book, fiction, journals and newspapers, conference proceeding, and religious works. The Holy Bible (4.2.2) is one book that includes all the required characteristics of other books: it has theories and explanations, as in the academic books; it has stories and histories, as in the historical and fictional works; it has recorded events, as in the newspapers; it has many different authors' writings, as in the journals and conference proceedings; it has a huge amount of commentary studies, as in the literature studies. The e-Bible is also the earliest book available free on Internet in computer readable format.

Table 2 Corpus Characteristics

Book Type	Size	Importance	Structure	Learnable	Readable	Available
Academic text and online learning materials	+	+	?	+	?	–
History	+	+	?	+	+	?
Fiction/Literature	+	?	?	+	+	+
Religion works (e.g., Bible)	+	+	+	+	?	+
Newspaper	+	–	–	?	+	+
Journal/Conference Proceeding	+	?	+	?	?	–

'+': Yes; '?': Not Sure; '–': No.

4.2.2 Collections: The Holy Bible

4.2.2.1 Versions

The Holy Bible is a collection of sixty six books considered by Christians as the *Word of God* and the *Guidance of Life*. It is the most printed and best selling book in human history, and a major influence on today's civilization. We downloaded two versions of the Holy Bible for this study: the King James Version (KJV, also known as Authorized Version) in plain text format (e-Resource 1997), and the New International Version (NIV) in HTML format (e-Resource 1995).

The KJV Bible in text format was originally distributed by the Centre for Computer Analysis of Texts (CCAT) at the University of Pennsylvania. We first chose it for the system prototype and evaluation purposes because it meets all the criteria that were discussed in previous section. Later we obtained permission from the International Bible Society and downloaded the NIV Bible in HTML format for user evaluation, because it is a modern English translation and has story divisions for easy reading. We did not use it in earlier development due to the copyright restrictions. Below, we will provide a brief summary of the organizational and narrative structural information of the Bible; this information is often referred to later in the system development and evaluation.

Though it was not the first Bible published in English, the King James Version was the first English translation of Bible by a committee of scholars in England. Similarly, the New International Version was translated by a committee of over 100 scholars in United States, working from the best available Hebrew, Aramaic, and Greek texts. It is the most popular version for modern readers. Importantly, story divisions and headings were added in order to help readers identify and follow the themes of the Bible.

4.2.2.2 Structures and Reference Style

The Holy Bible was written by approximately 40 people over the course of 1500 years. The books in the Bible are organized by type: The Pentateuch (the first 5 books), History (12 books), Poetry and Wisdom Literature (5 books), The Prophets

(17 books), Gospels (4 books), Acts (1 book), Letters (21 books), and Revelation (1 book). Many books in the Bible are named after their (apparent) authors. The variety of authors and their time of living created a book that is rich in narrative and literature styles, e.g., poetry, history, or letters. There are obvious language changes from time to time and from author to author, thus from book to book.

The Bible is divided into two big sections – the *Old Testament* and the *New Testament* – by the birth of *Jesus Christ*. The Old Testament contains 39 books from the Pentateuch to the Prophets; the New Testament contains 27 books from the Gospels to the Revelation.

In this thesis, we use a simple reference style to refer to the passages in the Bible: the book name, followed by the chapter number, colon, the number of starting verse, hyphen, and the number of ending verse. When a reference refers to a story, we will also include the story title in italic. For example: The story of '*The Birth of Jesus Christ*' in book Matthew, chapter 1, verses 18 to 25 is referenced as: *The Birth of Jesus Christ*, Matthew 1: 18 – 25.

4.2.2.3 Themes

These books in the Bible form one major theme and are in continuous narrative streams. The main theme, *God's glory and His plan for mankind's salvation*, can be divided into many topics/events, e.g., *the power of God*, *the establishment of the Old and New Testament*, *the incarnation of God*, *the promised birth of Jesus*, *the life and teachings of Jesus Christ*, *the miracles of Jesus*, *the death and resurrection of Jesus*, *the divinity of Jesus*, etc.

In spite of its popularity, the Holy Bible is not an easy book to read. This could be due to its historical span, its variety of authors, its symbolic language, and its multiple themes. These characteristics, however, provide a good text corpus for system evaluation and user evaluation.

4.2.3 Usage of the Collections and Other Resources

We use the KJV Bible in three ways: we adapted the free downloaded text to make a text collection in suitable format to support an early system prototype of the e-Book application (more details are discussed later in this chapter). We used the four Gospels as the corpus for the story segmentation evaluation (chapter 5) and for the evaluation of the narrative structure detection of e-Books (chapter 6).

We used the NIV Bible in three ways: we extracted information of story titles and breaks and created an index of stories in the Bible. We used it as a ‘ground truth’ for the story segmentation experiments (chapter 5); we used it to support the display of a list of stories in the e-Book user interface (details to follow in this chapter and chapter 7); we also used the entire book as e-Book corpus for the user evaluation (chapter 7).

It is worth mentioning the Gospel Harmony which played an important role in the evaluation of this study. A Gospel Harmony takes all events and topics recorded in the four Gospels (Matthew, Mark, Luke and John), combines them into a single account, in estimated chronological order, and lists them in each book side by side. It includes every chapter and verse of each Gospel, leaving nothing out. There are different versions of the Gospel Harmonies available; e.g., Nevin and Alfred (Nevin and Alfred 2002) and Galvin (Galvin 1986). We use the former one since it is available for download from the Blue Letter Bible website (e-Resource 2002). It is used in two ways in this study. The Gospel Harmony is used as ‘ground truth’ in the narrative structure detection experiments (chapter 6); semi-randomly selected stories of Gospel Harmony are also used in the designing of the reading tasks in the user evaluation of the user interfaces, as well as acted as the ‘ground truth’ in the information seeking task assessment (7.2.4.2).

We also downloaded lists of names for men and women in the Bible to help us identify character names in the text corpus (e-Resource 2000). These names are used in earlier user interface prototype (this chapter) and the story segmentation evaluation (chapter 5).

4.3 Discovering the Organizational Structure

4.3.1 Semi-automatic Discovery of the Organizational Structure of e-Book

Semi-automatic discovery of the organizational structure of this particular e-Book is relatively straightforward. The corpus, the Holy Bible, has a special organizational structure: the verses in the English Bible, regardless of the version, correspond to an original Hebrew or Greek sentence. When translated into English, each could be a sentence, a half sentence or a few sentences long.

The original text file downloaded is a big text file (about 4MB) for the entire book. It represents each verse as a line, and identifies each verse with three prefix numbers and a letter providing section, book, chapter, and verse information accordingly. For example: the book Genesis in the Old Testament, chapter 1, verse 1 is printed as:

*01**O** 1 1 In the beginning the God created heaven and earth.*

The book Revelation in the New Testament, chapter 22, verse 21 is printed as:

*66**N** 22 21 The grace of our Lord Jesus be with you all. Amen.*

The text file has 31,102 lines (verses) in total. We used a Java program to divide the text file into individual chapters, each organized in folders named after each book. We removed the numbers that identified each verse from the text, keeping the original verse, and organizing verses as individual lines in the text file. In this new collection, each chapter can be identified by file name and folder name (corresponding to the book). This helps the e-Book browser rapidly locate and load the chapter into the user interface.

For the user evaluation purpose, we downloaded a HTML-formatted Bible in New International Version book by book. The original html file looks like:

```
<h4>Mark 1</h4>

 <h5>John the Baptist Prepares the Way </h5>

 <sup id="en–NIV–24214">1</sup>The beginning of the gospel about Jesus Christ,
the Son of God. <p />

<sup id="en–NIV–24215">2</sup>It is written in Isaiah the prophet:<br />"I will
send my messenger ahead of you, <br />who will prepare your way"— <br />

 <sup id="en–NIV–24216">3</sup>"a voice of one calling in the desert,<br
/>'Prepare the way for the Lord, <br />make straight paths for him.' "

<sup id="en–NIV–24217">4</sup>And so John came, baptizing in the desert
region and preaching a baptism of repentance for the forgiveness of sins.

<sup id="en–NIV–24218">5</sup>The whole Judean countryside and all the
people of Jerusalem went out to him. Confessing their sins, they were baptized by
him in the Jordan River.

<sup id="en–NIV–24219">6</sup>John wore clothing made of camel's hair, with
a leather belt around his waist, and he ate locusts and wild honey.
```

> *<sup id="en–NIV–24220">7</sup>And this was his message: "After me will come one more powerful than I, the thongs of whose sandals I am not worthy to stoop down and untie.*
>
> *<sup id="en–NIV–24221">8</sup>I baptize you with water, but he will baptize you with the Holy Spirit."*

Structural headings, paragraphs, and sentence break information have special tags allocated, i.e., *<h1>, <p>, <br>*, and thus can be identified and extracted. We wrote a Perl program to parse these HTML files and partition its books into chapters. Original styles are kept in the chapter files, in HTML format. Meanwhile, we created an index of stories from the HTML files. For example, the above text could be indexed as:

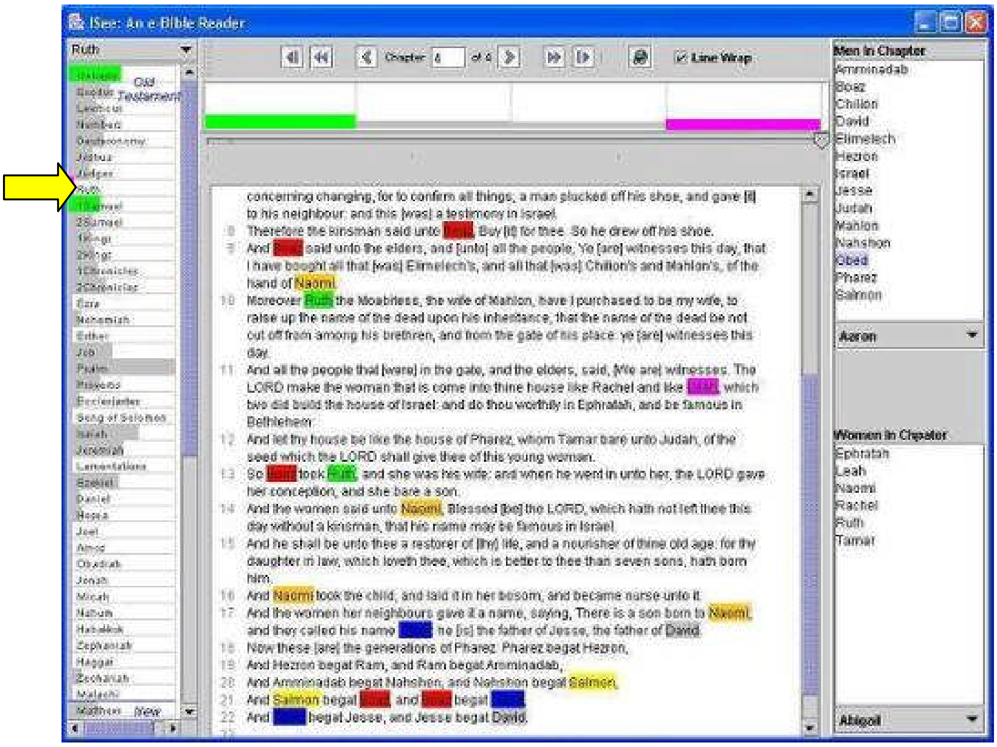
> *'John the Baptist Prepares the Way', Mark 1: 1-8.*

which included the title of story, the title of book, the number of chapter, and the number of starting and ending verses.

4.3.2 Presenting the Organizational Structure

For the first user interface prototype, we designed an interactive visualization tool for browsing books in the Bible. Figure 18 shows a screenshot of the system.

This e-Book browser was named 'iSee', means 'an *i*ntelligent *s*tructure r*e*vealing *e*-Book'; it also means '*I see the structures of the e-Book and therefore I understand*'. The vertical organizational structure visualization for books (shown by the arrow) provides an overview of the entire Bible based on books, and allows navigation between them. It shows the divisions of the Bible, and the order and title of the books. The shaded bars show the length of each book, with the current book highlighted in magenta and visited books highlighted in green.



concerning changing, for to confirm all things; a man plucked off his shoe, and gave [it] to his neighbour: and this [was] a testimony in Israel.
8 Therefore the kinsman said unto Boaz, Buy [it] for thee. So he drew off his shoe.
9 And Boaz said unto the elders, and [unto] all the people, Ye [are] witnesses this day, that I have bought all that [was] Elimelech's, and all that [was] Chilion's and Mahlon's, of the hand of Naomi.
10 Moreover Ruth the Moabitess, the wife of Mahlon, have I purchased to be my wife, to raise up the name of the dead upon his inheritance, that the name of the dead be not cut off from among his brethren, and from the gate of his place: ye [are] witnesses this day.
11 And all the people that [were] in the gate, and the elders, said, [We are] witnesses. The LORD make the woman that is come into thine house like Rachel and like Leah, which two did build the house of Israel: and do thou worthily in Ephratah, and be famous in Bethlehem:
12 And let thy house be like the house of Pharez, whom Tamar bare unto Judah, of the seed which the LORD shall give thee of this young woman.
13 So Boaz took Ruth, and she was his wife: and when he went in unto her, the LORD gave her conception, and she bare a son.
14 And the women said unto Naomi, Blessed [be] the LORD, which hath not left thee this day without a kinsman, that his name may be famous in Israel.
15 And he shall be unto thee a restorer of [thy] life, and a nourisher of thine old age: for thy daughter in law, which loveth thee, which is better to thee than seven sons, hath born him.
16 And Naomi took the child, and laid it in her bosom, and became nurse unto it.
17 And the women her neighbours gave it a name, saying, There is a son born to Naomi; and they called his name Obed: he [is] the father of Jesse, the father of David.
18 Now these [are] the generations of Pharez: Pharez begat Hezron,
19 And Hezron begat Ram, and Ram begat Amminadab,
20 And Amminadab begat Nahshon, and Nahshon begat Salmon,
21 And Salmon begat Boaz, and Boaz begat Obed,
22 And Obed begat Jesse, and Jesse begat David.

Figure 18 First Prototype

The horizontal organizational structure visualization for chapters provides an overview of each book based on chapters, and again allows navigation between them. It shows the number of chapters in each book, the current chapter, and the length of each chapter through the height of the bars.

Navigation was also provided by a dropdown list of books and navigation buttons for readers' convenience, i.e., 'previous book' and 'next chapter', etc. We used icons for these buttons, but the names of the buttons are shown when the mouse moves over them.

The reading window occupies the main central area in the user interface. It is designed to display the book chapter by chapter for easy reading. It also provides word highlighting for character names (details in next section). The browser window works closely with the other parts of the user interface. The current page could be loaded and updated by:

- Choosing a book from dropdown list of books, the first chapter of the book will be displayed by default;
- Clicking on a bar from the vertical visualization, representing all the books; and on a bar from the horizontal visualization, representing all the chapters in the book;
- Navigating through the navigation buttons or the sliding bar, namely, next chapter, previous chapter, next book, previous book, first book, last book.

The name listed on the right panels will be discussed in the next section.

4.4 Discovery and Presentation of the Narrative Structure

4.4.1 Detecting and Presenting the Main Characters

As mentioned in 4.2.3, we used a large list of names of men and women in the Bible (e-Resource 2000) to assist the detection of character names in the e-Book.

The identification of the character names in the text is similar like identifying stop-words in text retrieval process. We wrote a Java program to process the e-Book chapter files. For each chapter, we scanned it line by line; for each word, we compared it with the stop-word list and the character name lists to identify whether it was a stop word, a content word, a name of a man or a name of a woman; if it was a content word, we also stem it for further retrieval process. We used Porter's stemming algorithms (Porter 1980). We then created an inverted file (Baeza-Yates and Ribeiro-Neto 1999; Frakes and Baeza-Yates 1992) for each chapter. An inverted file is like an index in the back of a book that lists index terms alphabetically together with the page numbers where they can be found. Instead of page numbers, the inverted file structure lists a document (chapter) identifier, together with the positions of the term in the document, the term frequency, the attribute (e.g., gender for names) of the term, etc.

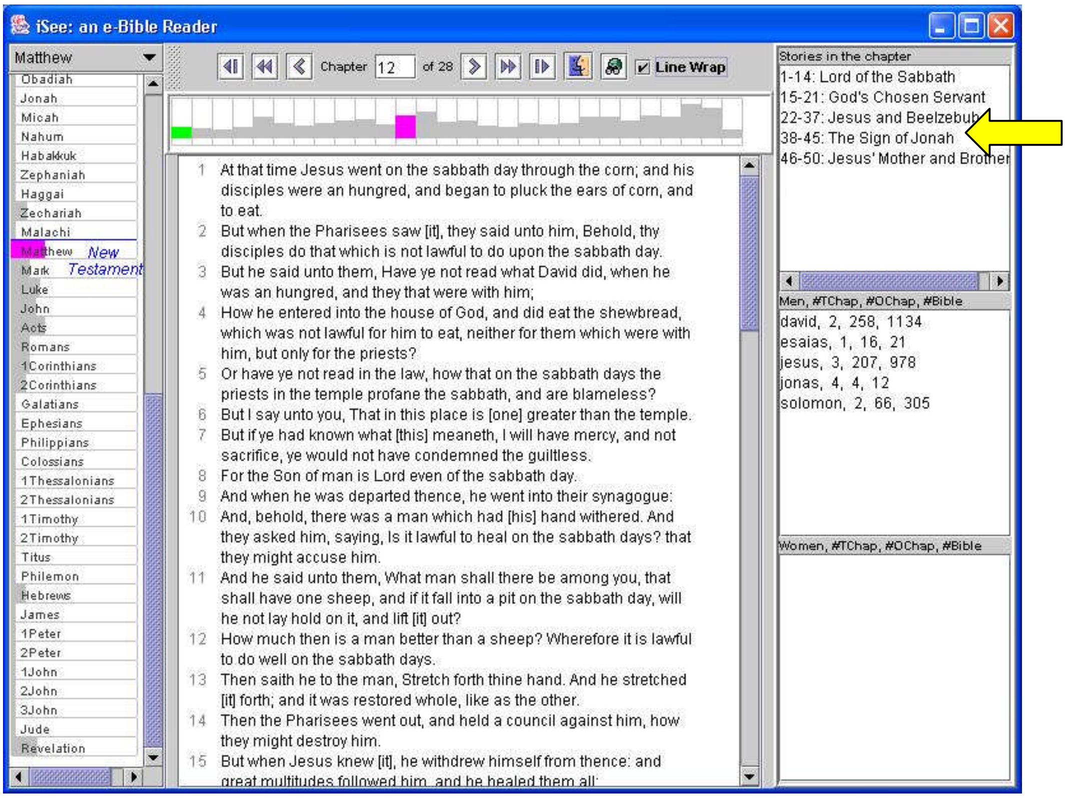
When a name is searched for, we retrieve the name from the index stored in the inverted file to get location information of the word. We provide statistics data of the name, for example, how many times the name appeared in the Bible, the section, the book and the chapter. This could be an indication of the importance of the character since usually a most mentioned name is the most important character in a book.

Also using the index, the user interface provided a list of women and a list of men in the current chapter (right column in Figure 18, page 63), which is dynamically updated when chapter is changed. Names can be highlighted once searched.

4.4.2 Discovery of the Narrative Segments

In chapter 5, we will discuss the story segmentation techniques we used to identify narrative segments in the King James Version text in more detail.

In the user interface prototype shown in Figure 19, we used the story segments identified by the editors of the New International Version of the Bible. We extracted the story break information and created an index of stories. In the user interface, we displayed this index as a ‘mini’ table of contents for each chapter (shown by the arrow in Figure 19). The reader can then look up the story title in the ‘mini’ table of contents, and get a better view of current chapter.



1 At that time Jesus went on the sabbath day through the corn; and his disciples were an hungred, and began to pluck the ears of corn, and to eat.
2 But when the Pharisees saw [it], they said unto him, Behold, thy disciples do that which is not lawful to do upon the sabbath day.
3 But he said unto them, Have ye not read what David did, when he was an hungred, and they that were with him;
4 How he entered into the house of God, and did eat the shewbread, which was not lawful for him to eat, neither for them which were with him, but only for the priests?
5 Or have ye not read in the law, how that on the sabbath days the priests in the temple profane the sabbath, and are blameless?
6 But I say unto you, That in this place is [one] greater than the temple.
7 But if ye had known what [this] meaneth, I will have mercy, and not sacrifice, ye would not have condemned the guiltless.
8 For the Son of man is Lord even of the sabbath day.
9 And when he was departed thence, he went into their synagogue:
10 And, behold, there was a man which had [his] hand withered. And they asked him, saying, Is it lawful to heal on the sabbath days? that they might accuse him.
11 And he said unto them, What man shall there be among you, that shall have one sheep, and if it fall into a pit on the sabbath day, will he not lay hold on it, and lift [it] out?
12 How much then is a man better than a sheep? Wherefore it is lawful to do well on the sabbath days.
13 Then saith he to the man, Stretch forth thine hand. And he stretched [it] forth; and it was restored whole, like as the other.
14 Then the Pharisees went out, and held a council against him, how they might destroy him.
15 But when Jesus knew [it], he withdrew himself from thence: and great multitudes followed him, and he healed them all:

Stories in the chapter
1-14: Lord of the Sabbath
15-21: God's Chosen Servant
22-37: Jesus and Beelzebub
38-45: The Sign of Jonah
46-50: Jesus' Mother and Brother

Men, #TChap, #OChap, #Bible
david, 2, 258, 1134
esaias, 1, 16, 21
jesus, 3, 207, 978
jonas, 4, 4, 12
solomon, 2, 66, 305

Women, #TChap, #OChap, #Bible

Figure 19 Assisting Readers with Stories

Note that the text displayed in the original King James Version text did not have story information. For the user evaluation, however, we used the NIV Bible for reading tasks. The chapter is displayed in HTML format with story headings and paragraph divisions to assist reading. See the iSee user interface in Figure 21, page 68 for a comparison.

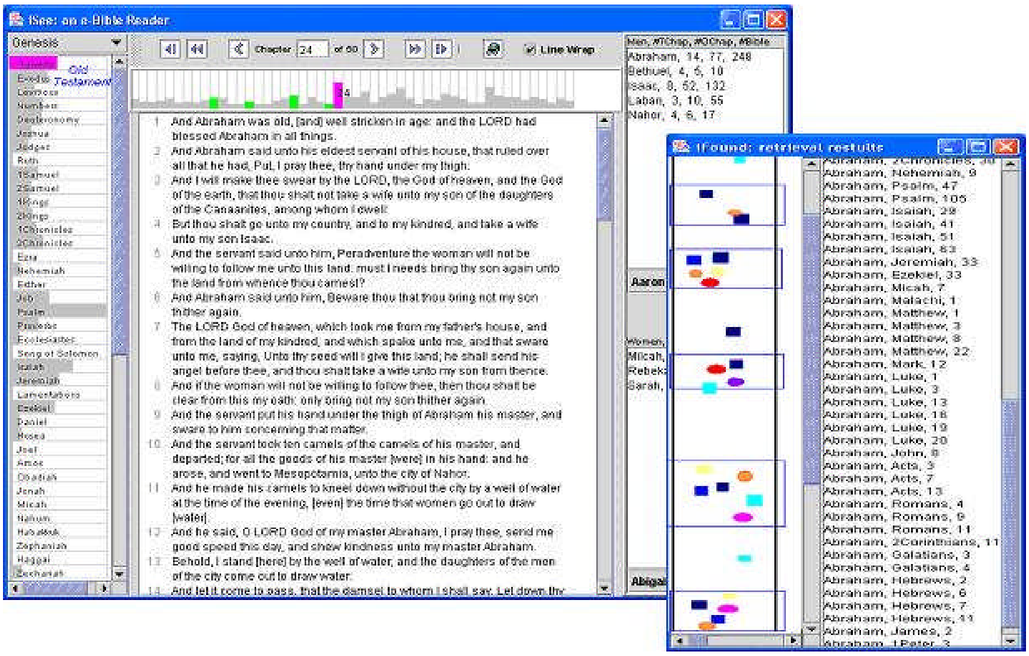
4.4.3 Discovery of the Narrative Structure

Driven by the user scenarios discussed in chapter 1, we designed the e-Book user interface comprising an e-Book browser, a corresponding overview visualization of the e-Book organizational structure, a character index, a story index, and a narrative thread searching engine. The technique we used to discover the narrative structure of e-Book based on the narrative segments will be reported and discussed in chapter 5 and 6.

4.4.3.1 Discovering Narrative Structure through Character Names

We designed a prototype user interface that supports finding narrative structure of e-Book through the names of the main characters. Firstly, the names of men and women are identified and listed (Figure 18); secondly, the occurrences of names in the text were counted and displayed (Figure 20); thirdly, a list of all the passages where a name appears can be retrieved (Figure 20, the smaller window).

We also explored a number of possible methods of visualize the characters, for example, using squares for men and circles for women, with different colours indicating different names, etc. A cluster of names would then indicate a story where the names are mentioned together.



Men, #TChap, #OChap, #Bible
Abraham, 14, 77, 248
Bethuel, 4, 6, 10
Isaac, 8, 52, 132
Laban, 3, 10, 55
Nahor, 4, 6, 17

1 And Abraham was old, [and] well stricken in age: and the LORD had blessed Abraham in all things.
2 And Abraham said unto his eldest servant of his house, that ruled over all that he had, Put, I pray thee, thy hand under my thigh:
3 And I will make thee swear by the LORD, the God of heaven, and the God of the earth, that thou shalt not take a wife unto my son of the daughters of the Canaanites, among whom I dwell:
4 But thou shalt go unto my country, and to my kindred, and take a wife unto my son Isaac.
5 And the servant said unto him, Peradventure the woman will not be willing to follow me unto this land: must I needs bring thy son again unto the land from whence thou camest?
6 And Abraham said unto him, Beware thou that thou bring not my son thither again.
7 The LORD God of heaven, which took me from my father's house, and from the land of my kindred, and which spake unto me, and that sware unto me, saying, Unto thy seed will I give this land; he shall send his angel before thee, and thou shalt take a wife unto my son from thence.
8 And if the woman will not be willing to follow thee, then thou shalt be clear from this my oath: only bring not my son thither again.
9 And the servant put his hand under the thigh of Abraham his master, and sware to him concerning that matter.
10 And the servant took ten camels of the camels of his master, and departed; for all the goods of his master [were] in his hand: and he arose, and went to Mesopotamia, unto the city of Nahor.
11 And he made his camels to kneel down without the city by a well of water at the time of the evening, [even] the time that women go out to draw [water].
12 And he said, O LORD God of my master Abraham, I pray thee, send me good speed this day, and shew kindness unto my master Abraham.
13 Behold, I stand [here] by the well of water; and the daughters of the men of the city come out to draw water:
14 And let it come to pass, that the damsel to whom I shall say, Let down thy

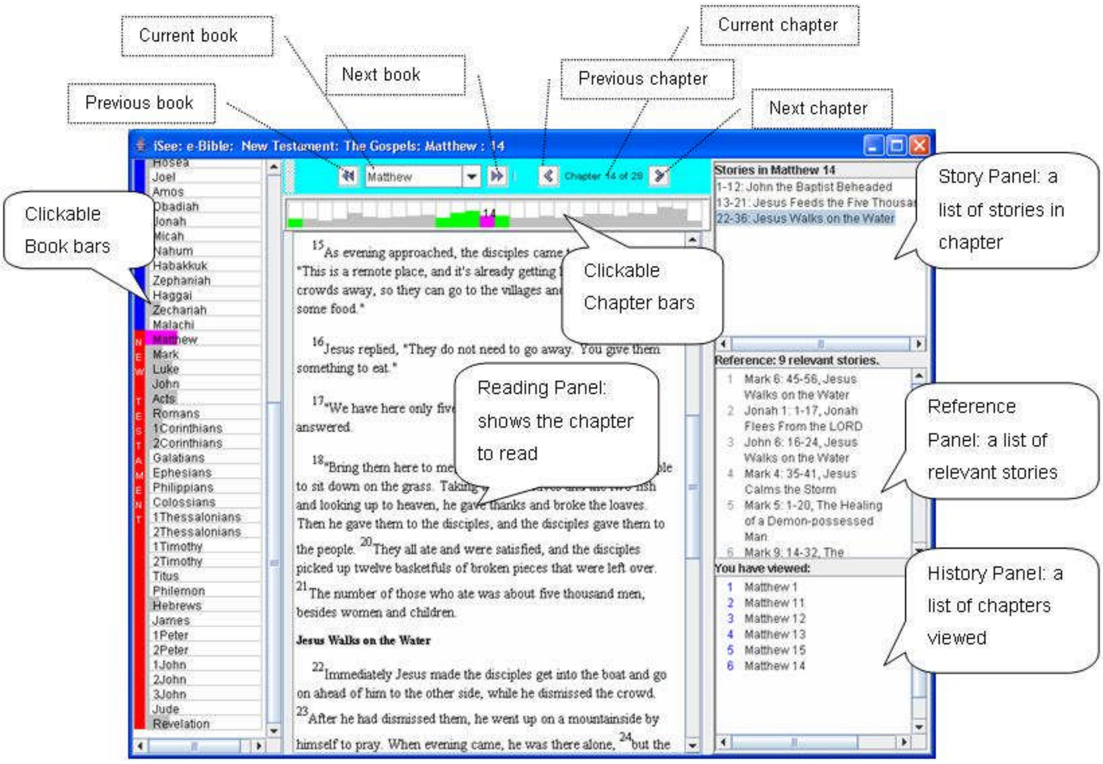
Abraham, 2Chronicles, 20
Abraham, Nehemiah, 9
Abraham, Psalm, 47
Abraham, Psalm, 105
Abraham, Isaiah, 29
Abraham, Isaiah, 41
Abraham, Isaiah, 51
Abraham, Isaiah, 63
Abraham, Jeremiah, 33
Abraham, Ezekiel, 33
Abraham, Micah, 7
Abraham, Malachi, 1
Abraham, Matthew, 1
Abraham, Matthew, 3
Abraham, Matthew, 8
Abraham, Matthew, 22
Abraham, Mark, 12
Abraham, Luke, 1
Abraham, Luke, 3
Abraham, Luke, 13
Abraham, Luke, 16
Abraham, Luke, 19
Abraham, Luke, 20
Abraham, John, 8
Abraham, Acts, 3
Abraham, Acts, 7
Abraham, Acts, 13
Abraham, Romans, 4
Abraham, Romans, 9
Abraham, Romans, 11
Abraham, 2Corinthians, 11
Abraham, Galatians, 3
Abraham, Galatians, 4
Abraham, Hebrews, 2
Abraham, Hebrews, 6
Abraham, Hebrews, 7
Abraham, Hebrews, 11
Abraham, James, 2
Abraham, 1Peter, 3

Figure 20 Assisting Readers to Explore the Main Characters

Although we analysed the effects of character names in the story segmentation evaluation, we did not include it in the user evaluation of the user interface, limited by the reading tasks and reading time available for each participant. The impact of the character names is an interesting area for future development.

4.4.3.2 Discovering Narrative Structure through Narrative Threads

We designed two comparison systems for the user evaluation, shown in Figure 21 and Figure 22.



[15]As evening approached, the disciples came t
"This is a remote place, and it's already getting l
crowds away, so they can go to the villages an
some food."

[16]Jesus replied, "They do not need to go away. You give them something to eat."

[17]"We have here only fiv
answered.

[18]"Bring them here to me ble
to sit down on the grass. Taking fish
and looking up to heaven, he gave thanks and broke the loaves. Then he gave them to the disciples, and the disciples gave them to the people. [20]They all ate and were satisfied, and the disciples picked up twelve basketfuls of broken pieces that were left over. [21]The number of those who ate was about five thousand men, besides women and children.

Jesus Walks on the Water

[22]Immediately Jesus made the disciples get into the boat and go on ahead of him to the other side, while he dismissed the crowd. [23]After he had dismissed them, he went up on a mountainside by himself to pray. When evening came, he was there alone, [24]but the

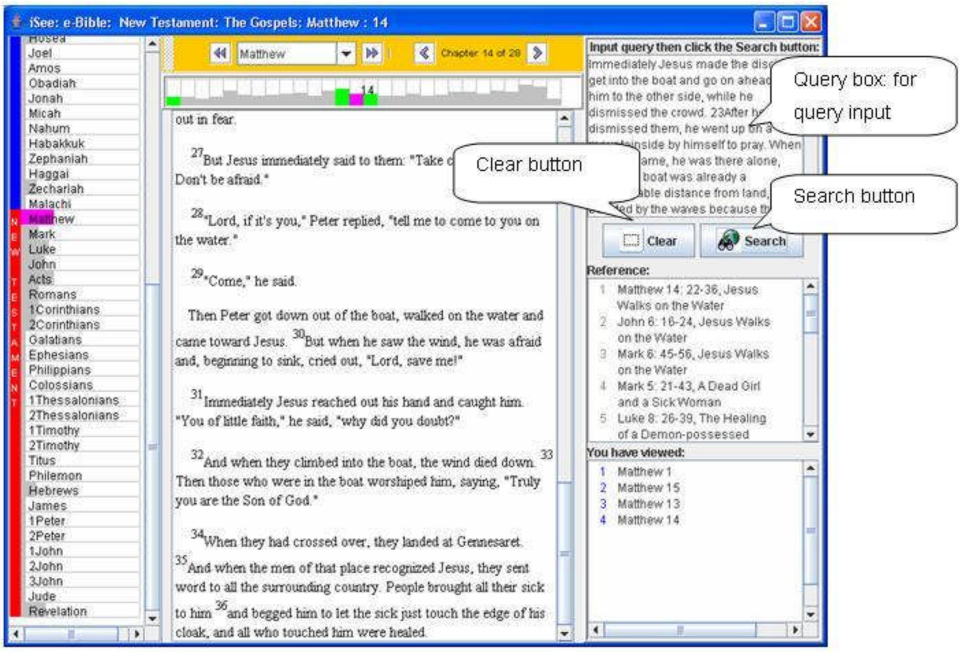
Stories in Matthew 14
1-12: John the Baptist Beheaded
13-21: Jesus Feeds the Five Thousa
22-36: Jesus Walks on the Water

Reference: 9 relevant stories.
1 Mark 6: 45-56, Jesus Walks on the Water
2 Jonah 1: 1-17, Jonah Flees From the LORD
3 John 6: 16-24, Jesus Walks on the Water
4 Mark 4: 35-41, Jesus Calms the Storm
5 Mark 5: 1-20, The Healing of a Demon-possessed Man
6 Mark 9: 14-32, The

You have viewed:
1 Matthew 1
2 Matthew 11
3 Matthew 12
4 Matthew 13
5 Matthew 15
6 Matthew 14

Figure 21 iSee: an Intelligent Story-Theme Linking e-Book

Following our previous prototypes, we called the e-Book browser with story linking features iSee, meaning that 'an *i*ntelligent *S*tory-Th*e*me linking *e*-Book'. Its main function was to support a user rapidly browsing an e-Book, understanding the organizational structure of the book, linking related stories and identifying themes. It did this by listing stories on the page (in the story panel), and by linking similar or related stories in a thread (in the reference panel). Readers could then browse related stories *in situ* without considering how to do it.



out in fear.

[27]But Jesus immediately said to them: "Take c
Don't be afraid."

[28]"Lord, if it's you," Peter replied, "tell me to come to you on the water."

[29]"Come," he said.

Then Peter got down out of the boat, walked on the water and came toward Jesus. [30]But when he saw the wind, he was afraid and, beginning to sink, cried out, "Lord, save me!"

[31]Immediately Jesus reached out his hand and caught him. "You of little faith," he said, "why did you doubt?"

[32]And when they climbed into the boat, the wind died down. [33] Then those who were in the boat worshiped him, saying, "Truly you are the Son of God."

[34]When they had crossed over, they landed at Gennesaret. [35]And when the men of that place recognized Jesus, they sent word to all the surrounding country. People brought all their sick to him [36]and begged him to let the sick just touch the edge of his cloak, and all who touched him were healed.

Input query then click the Search button:
immediately Jesus made the disc
get into the boat and go on ahead
him to the other side, while he
dismissed the crowd. 23After h
dismissed them, he went up on a
inside by himself to pray. When
ame, he was there alone,
boat was already a
able distance from land,
ed by the waves because th

Reference:
1 Matthew 14: 22-36, Jesus Walks on the Water
2 John 6: 16-24, Jesus Walks on the Water
3 Mark 6: 45-56, Jesus Walks on the Water
4 Mark 5: 21-43, A Dead Girl and a Sick Woman
5 Luke 8: 26-39, The Healing of a Demon-possessed

You have viewed:
1 Matthew 1
2 Matthew 15
3 Matthew 13
4 Matthew 14

Figure 22 iSearch

The comparison system was named iSearch (Figure 22), since it was driven largely by a query search tool powered by Lucene 1.9.1 (Cutting, 2006). Stories in the Bible were index in advance using Lucene’s Indexer. Reader can submit a query on the iSearch user interface and retrieve a list of related stories.

The iSee system took the advantage of the semi-automatically discovered narrative threads; the other system, iSearch, simply provided a query searching tool for reader to discover the narrative threads manually. To help focus the user study on reading task and task effectiveness assessment, we removed all the other features that were not necessary in the evaluation, i.e., the character names detection and highlighting, and added a history panel that assisted reader to track pages which had been visited.

The main similarities between the two systems were: both e-Book user interfaces displayed the NIV Bible in HTML format and supported same functions of browsing through the interactive visualization tools; stories in the Bible were indexed in advance to identify the story titles and breaks, both of which were used to rank and select the related stories retrieved. At first sight the two user interfaces were very similar, and the reading and browsing functions were made equivalent in iSee and iSearch.

More differences between the iSee and iSearch will be discussed in the chapter 7, which describes the system evaluation in more detail.

5

STORY SEGMENTATION AND EVALUATION

5.1 Introduction

A well designed structure for electronic texts can help readers focus on and get better understanding of the ideas and views of the text. In order to discover and present the narrative structure of an e-Book, there are three implementation steps:

- Partition the e-Book text into story segments and create a collection of the story segments;
- For each of the story segment in the collection, compare it with every other story and generate a similarity score for each pair;
- Rank the list of similar story segments and generate a narrative thread for that story.

In this chapter we will report our approaches and findings of the first step towards narrative structure detection. We will discuss the next two steps in chapter 6.

In the field of Information Retrieval, there has been a surge of interest in the role of passages in full text. Subtopic structure is sometimes marked in technical texts by headings and subheadings. Brown and Yule (1983) stated that this kind of division is one of the most basic in discourse. However, many expository texts consist of long sequences of paragraphs with very little structural demarcation.

Converting text to hypertext, in what is called post hoc authoring, requires division of the original text into meaningful units as well as meaningful interconnection of the units (Marchionini, Liebscher et al. 1991). Salton et al. (1996) have recognized the need for multi-paragraph units in the automatic creation of hypertext links as well as theme generation.

Discourse analysis studies the interdependencies between utterances (words, phrases, and clauses) and how these dependencies contribute to the overall coherence of a text. Segmentation is one method of exploring discourse structure (Stokes 2004). Dividing long text into short blocks with suitable sub-headings can help generate and distinguish the narrative threads, and thus help a reader to recognize, understand, and remember the themes.

Finding stories in the text helps to resolve two further problems in the discovery of the narrative structure: retrieving stories and linking related stories that form a narrative thread. Moreover, showing story boundaries can improve the presentation of the text in the e-Book user interface.

5.2 Review of Previous Works and Corpora

5.2.1 Previous Works

With the observation and assumption that a change in subject is accompanied by a change in vocabulary, Hearst (Hearst 1994; Hearst 1997) investigated a technique called TextTiling which subdivided texts into multi-paragraph units. A multi-paragraph unit represents one story or sub-topic in the text, and it is also called a story segment. TextTiling is based on a block comparison algorithm, in which adjacent pairs of text blocks (we call them windows in this thesis) are compared for overall lexical similarity. The TextTiling segmentation requires that a score, called the lexical score, be computed for the gap between every pair of text blocks (Figure below, white boxes). A lower similarity score indicates a possible topic shift.

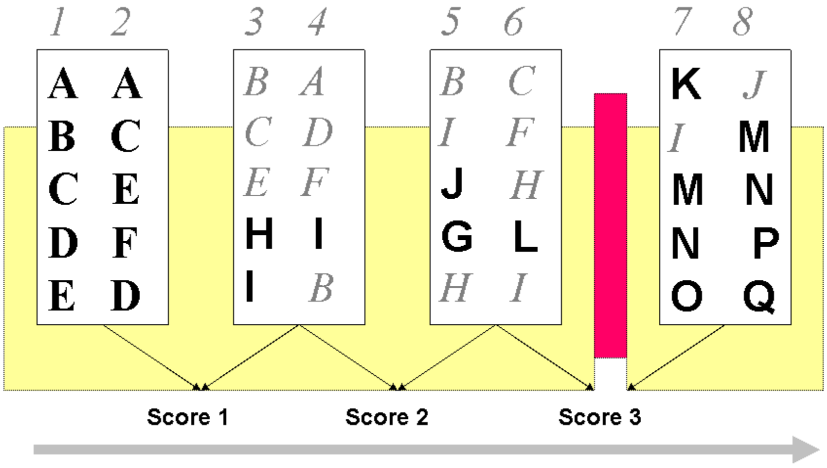


Figure 23 Illustration of the TextTiling comparison algorithm

In Figure 23, letters signify lexical items; numbers signify sentence/line numbers; and boxes represent text blocks. By comparing the neighbouring blocks we find repeated items (*italic*) and newly introduced items (*Arial black*). When the vocabularies in two blocks are significantly different, we assign a topic boundary at the gap (dark colour vertical bar), which divides the text into story segments (light coloured frames in the background).

Taking the same assumption as Hearst, Richmond, Smith et al. (1997) weight word significance using the ‘burstiness’ of content words in text. Burstiness is an observable characteristic of important topic words, where multiple occurrences of topic words tend to occur in close proximity to each other in a text. Richmond et al.

define a significance weight for each word in a text where words which observe a ‘bursty’ distribution will be weighted higher than other words. They claim that incorporating this measure of word significance into the segmentation process leads to improved accuracy over Hearst’s TextTiling algorithm without sacrificing language independency.

Story segmentation is a sub-category of text segmentation and is often labelled as ‘coarse-grained’ text segmentation. The objectives of the text segmentation research include:

- Model discourse structure
- Detect topic shift in linear text
- Improve IR, summarization and text displaying tasks

Most of the work in story segmentation stems from Hearst’s TextTiling algorithm, which determines boundaries based on a comparison of neighbouring text blocks. Although initially the story segmentation is devised mainly to model the discourse structure or to restore the authorial structure, recent developments are more interested in the news story segmentation, that is, the segmentation of broadcast news programmes into distinct news stories based on given topics (Allan 2002).

One of our objectives – detecting and presenting the narrative structure of e-Books to help comprehension – requires an accurate and effective story segmentation technique that focuses on the identifying the stories in a narrative thread in the discourse.

5.2.2 Traditional Corpora

Table 3 shows a number of corpora that have been used for story segmentation evaluation. These include expository texts, news articles, news streams and narratives.

Table 3 Relevant Works and Corpus

Corpora	Reference
20 spontaneous oral narratives	(Passonneau and Litman 1993)
Expository texts: 12 magazine articles	(Hearst 1994; 1997)
A 800 sentences from articles of the Times; a 200 sentences psychology paper.	(Richmond, Smith et al. 1997)
Wall Street Journal news articles; CNN broadcast news in TDT corpus	(Beeferman, Berger et al. 1999)

1996 HUB-4 Broadcast News Corpus;
TDT corpus (speech);
four texts from the Gutenberg free e-Books library
	(Reynar 1998; 1999)

News articles and informative text in the Brown corpus	(Choi 2000)
TDT 2002 corpora	(Chen, Brants et al. 2003)

Passonneau and Litman (1993) argued that a discourse consists not simply of a linear sequence of utterances, but of meaningful relations among the utterances. They presented quantitative results of a two part study using a corpus of 20 spontaneous, narrative monologues. The first part evaluated the statistical reliability of human segmentation of the corpus, where speaker intention is the segmentation criterion. Results showed that human subjects can reliably perform linear discourse segmentation with spoken narratives, using an information notion of speaker intention. The agreement between the human judges is highly significant. They then used the subjects' segmentations to evaluate the correlation of discourse segmentation algorithms with three linguistic cues (referential noun phrases, cue words, and pauses), using information retrieval metrics. The performance of the algorithms was comparable with humans’ as measured by recall, but much lower as measured by Precision.

In Hearst’s (1994) study, she used a collection of 12 magazine articles that satisfied a length criteria (between 1,800 and 2,500 words), and that contained little structural demarcation. Richmond et al claimed that they have improved the TextTiling algorithms with a new measure of word significance, which exhibits improved accuracy without sacrificing language independency. They evaluated the approach with news articles and a psychology paper.

The Topic Detection and Tracking (TDT) research started with a pilot study in 1997 and has continued with open evaluations in 1998 – 2004. The TDT1 pilot study corpus comprises 15,863 news stories spanning from the 1st of July 1994 to the 30th of June 1995. The TDT2 corpus consists of 64,000 stories spanning the first six months of 1998 taken from six different news sources and has three different versions (Stokes 2004). Given the special characteristic structure of news, each of the news documents only focuses on one event/topic, which might be identified and matched to a given topic. We will come back to this in next chapter.

5.3 Methodologies and Algorithms

5.3.1 Methodologies

We used a story segmentation algorithm (see next section) which is similar to the original TextTiling block comparison algorithm (Hearst 1997) for the purpose of narrative thread detection in our e-Book application. But differently, we model our text windows using statistical language modelling, and specifically a simple ‘bag of words’ model (Frakes and Baeza-Yates 1992). This is done by three steps:

- Model the text windows using unigram word models
- Calculate the probabilities of each term using the maximum likelihood estimate (*mle*)
- Use various similarity measures to compute the dissimilarity between windows.

We observed that a topic transition is usually accompanied by two facts: introducing new words, and, stopping using old words (Figure 23). Therefore, a symmetric divergence measure would be effective in detecting the topic boundaries, because it measures these two facts equally. In this chapter we present the algorithms and evaluate three similarity measures (see 5.3.4) by which the consecutive text window models are compared:

- Jensen difference measure, which is a symmetric divergence measure (Taneja 2001) that measures the association of two probability distributions.
- Kullback-Leibler (1951) measure, and
- Cosine-distance measure (Salton, Allan et al. 1993; Salton and Buckley 1998) which is used in the TextTiling algorithm.

Another observation is that a new story often starts by introducing new character names. It may improve the story segmentation by giving names in the vocabulary more consideration. Critically, we are able to adjust word counts of the occurrence of

Biblical characters (e.g. Jacob) when computing term probabilities (see 5.3.7). These character names are identified by reference to a gazette of Biblical names (e-Resource 2000).

We also explored the effect of the size of the text window combined with the cut-off level of the potential boundaries, and the effect of the choice of words weighted in the algorithm.

5.3.2 Story Segmentation Algorithms

Our design of the story segmentation algorithms were based on Hearst's TextTiling algorithm, we took into account the name of characters in the text.

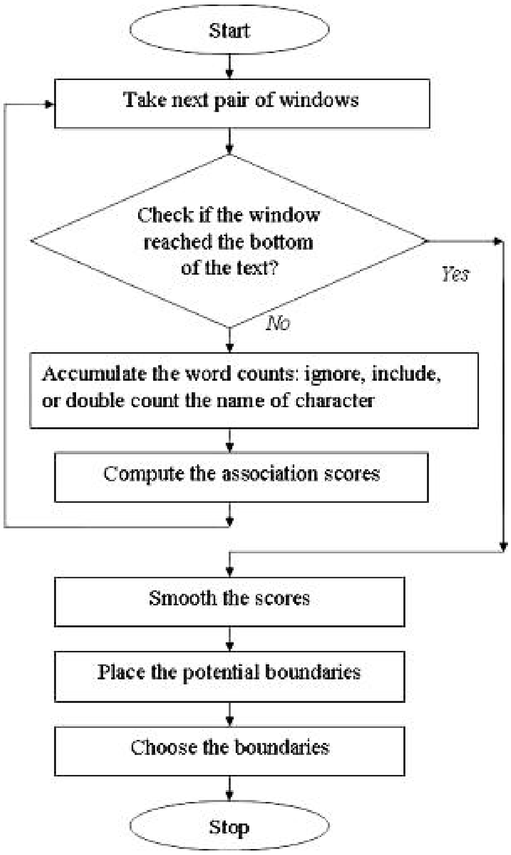

Start
Take next pair of windows
Check if the window reached the bottom of the text?
Yes
No
Accumulate the word counts: ignore, include, or double count the name of character
Compute the association scores
Smooth the scores
Place the potential boundaries
Choose the boundaries
Stop

Figure 24 Flowchart of the segmentation algorithm

Figure 24 shows the sequential, modular flowchart of the algorithm. The details of every step in the algorithm are discussed in the following sections.

Given the structure of the text collection, we use the verses instead of sentences. Note that in the algorithm our text windows differ from Hearst's text blocks concept. Our text window is made up of a fixed number of verses, where the length may vary; Hearst's text block is made up of a fixed number of mock sentences with fixed number of words (e.g., count sequential 20 words to form a sentence).

5.3.3 Illustration of the Entire Processing

For a given chapter of x verses, and a given window size y, we take every y verses as a window, compare it with the next window, and get a lexical association measure for the window pair; shift the windows down one verse and compare them again, do it again till the window reached the last verse. Thus we get an association score for each verse from verse y+1 to x–y+1. These are the original scores. After smoothing the scores and converting the smoothed scores to the relevant height score, we get a number of potential boundaries. We then use a cut-off threshold to select desired boundaries. Details of the similarity measures, smoothing methods, cut-off threshold, etc, will be discussed in the following sections.

Suppose we have a text file of the chapter 20 of Luke, which has 47 verses. There are five stories identified by editors of the New International Version Bible, and the story breaks are at the beginning of verse 9, 20, 27, and 41. Figure 25 shows the automatic boundary detection procedure on the chapter with the text window size set to 6, and two cut-off thresholds were placed to filter the potential boundaries.

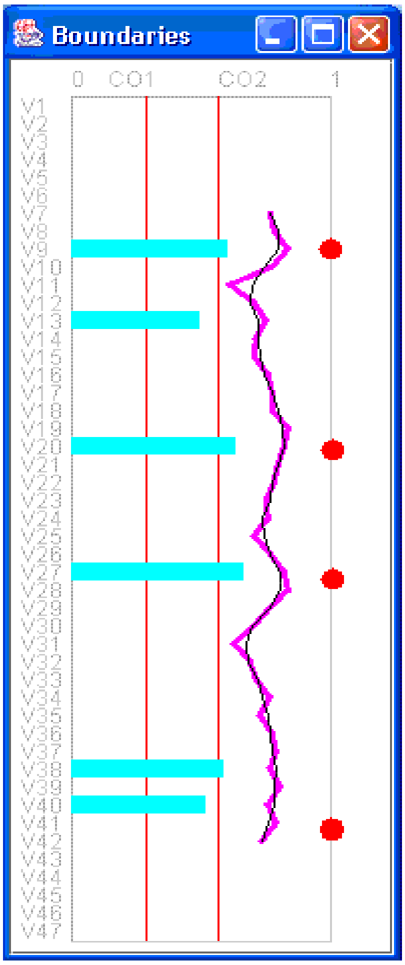


Figure 25 Illustration of the boundary detection process

In this diagram, the magenta curve in the right part of the figure represents the original scores; the black curve represents the smoothed scores; the horizontal bars are the potential boundaries detected; dots indicate the true boundaries. Two cut-off thresholds are placed to filter the potential boundaries, as shown by the vertical red lines in the middle.

The lower cut-off (CO_1) assigns more boundaries (verse 9, 13, 20, 27, 38, 40), while the higher cut-off (CO_2) assigns fewer boundaries (verse 9, 20, 27, 40). In this case, the results of higher cut-off are closer to the true story breaks placed by human beings than the lower cut-off.

These automatically detected boundaries can then be compared with human's judgment to see the effectiveness of the process, as measured by Precision, Recall and F-measure. Note that we assume the beginning of a chapter is also the beginning of the first story in the chapter. Verse 1 is automatically assigned as a boundary during the process.

5.3.4 Similarity Measures (SM)

5.3.4.1 Modelling the Text Windows

Traditionally, IR uses a matching function to measure the association between two texts or cluster profiles (Van Rijsbergen 1979). In this paper we model our text windows using statistical language modelling, and specifically a simple 'bag of words' model. Three similarity measures are used to compute the association scores of the text windows, in order to measure the dissimilarity between them.

In Ponte and Croft (Ponte and Croft 1998) approach, they model (*M*) documents using probability (*P*) of term (*t*) distributions:

$P_{mle}(t \mid M(d)) = \frac{tf(t,d)}{dl(d)}$	Equation 1

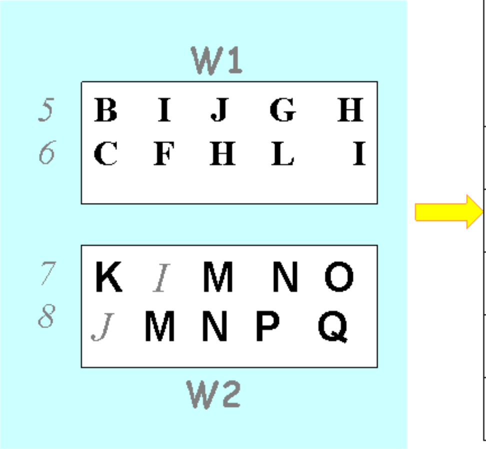
where *tf (t, d)* is the term frequency of *t* in document *d*, and *dl (d)* is the length in words of *d*.

We applied it with the text windows. Assume that the two text windows W_1 and W_2 can be represented as probability distributions, thus we can use a maximum likelihood estimator (*mle*) for the model (*M*) of a text window (*W*). In our version,

$$P_{mle}(t_i \mid M(W_j)) = \frac{tf(t_i, W_j)}{dl(W_j)}$$	Equation 2

where *tf (t, W$_j$)* is the term frequency of *t* in text window *W$_j$*, and *dl (W$_j$)* is the length in words of *W$_j$*.

Figure 26 shows an illustration of this calculation taking a pair of windows from the example illustrated in Figure 23.



t_i	$P(t_i\|W_1)$	$P(t_i\|W_2)$
B	1/10	0
C	1/10	0
I	2/10	1/10
…	…	…
Q	0	1/10

Figure 26 Illustration of Modelling the Text Windows

If a term does not appear in a window, it will cause a probability of zero for this term, and this is a well-known problem in language modelling. One way to solve the problem is to smooth the probability formula. There are various smoothing methods, for example, see Zhai and Lafferty's paper (2004). In this chapter, we modify each term frequency by adding a parameter α:

$$P_{mle}(t_i \mid M(W_j)) = \frac{tf(t_i, W_j) + \alpha}{dl(W_j)} \quad (\alpha = 0.5)$$	Equation 3

We simply treat the non-occurrence term as if it appeared *'half a time'* or *'less than once'* in the text by setting the parameter to 0.5.

5.3.4.2 Symmetric Divergence Measures (Jensen)

Recall that we observed that a topic transition is usually accompanied by two facts: introducing the new words, and, stopping using the old words (see Figure 26; compare the W_1 and the W_2). The dissimilarity between the two text windows can be measured based on these two facts in two ways: symmetric comparison which considers the two facts equally, and asymmetric comparison, which considers only one fact, i.e., how many old words remain.

Using symmetric comparison, a dissimilarity score between two windows can be computed using the *Jensen* difference divergence measure (Taneja 2001). This is a symmetric divergence measure that measures the extent to which windows differ.

$$I(W_1 \| W_2) = \sum_{i=1}^{n} [\frac{p_i \log p_i + q_i \log q_i}{2} - \left(\frac{p_i + q_i}{2}\right) \log \left(\frac{p_i + q_i}{2}\right)] \qquad \text{(}\textit{Jensen}\text{)}$$

where $p_i = P_{mle}(t_i|W1)$ and $q_i = P_{mle}(t_i|W2)$.

This formula yields a score between 0 and 1. The higher the dissimilarity score, the higher the possibility of a topic shift.

5.3.4.3 Kullback-Leibler Measure

Based on our observation, we also suspect that in order to increase the accuracy of the result, we should pay more attention on how well the first window predicts the second, thus, how many old words repeated in the second window. This could be done by applying the Kullback-Leibler model.

In probability theory and information theory, the Kullback-Leibler divergence (or information divergence, or relative entropy) is a natural distance measure from a ‘true’ probability distribution *p* to an arbitrary probability distribution *q* (Kullback and Leibler 1951; Bigi 2003). Typically *p* represents data, observations, or a precise calculated probability distribution; and *q* represents a theory, a model, a description or an approximation of *p*.

We adapted the Kullback-Leibler divergence model to measure the distance of the probability distributions of the second text window from the first.

$I(W_1 \parallel W_2) = \sum_{i=1}^{n} \left[q_i \log \frac{q_i}{p_i} \right] = \sum_{i=1}^{n} [-q_i \log p_i - (-q_i \log q_i)]$	(Kullback-Leibler)

where $p_i = P_{mle}(t_i|W1)$ and $q_i = P_{mle}(t_i|W2)$.

This formula yields a score with a positive value. Using this formula, a higher score indicates a higher dissimilarity, hence a higher possibility of a topic shift.

5.3.4.4 Cosine-Distance Measure

Hearst (1997) used the ad hoc Cosine-distance measure for scoring the text blocks in the TextTiling algorithm. The concept of the Cosine-distance measure is explained in (Salton and Buckley 1998).

$Cos\ (W_1, W_2) = \frac{\sum_{i=1}^{n} p_i q_i}{\sqrt{\sum_{i=1}^{n} p_i^2 \sum_{i=1}^{n} q_i^2}}$	Equation 4

where $p_i = tf(t_i|W1)$ and $q_i = tf(t_i|W2)$. Since the term frequency *tf* is always a positive value, this measure yields a score ranging from 0 to 1. Lower scores indicate higher possibility of a topic shift.

At first we thought using the maximum likelihood estimate (*mle*) in this formula will improve the performance, but soon we realized that it would not make any difference, as some information has been cancelled in the formula (see Appendix A2 for further details). In fact *mle* is a normalized form of *tf*, albeit interpreted as a probability.

We compared this measure with *Jensen* and *Kullback-Leibler* and, to be consistent with scores of the *Jensen*, we modified the result by subtracting it from 1, and call it *Cosine*.

$1 - Cos(W_1, W_2) = 1 - \frac{\sum_{i=1}^{n} p_i q_i}{\sqrt{\sum_{i=1}^{n} p_i^2 \sum_{i=1}^{n} q_i^2}}$	(*Cosine*)

where p_i = *tf(*t_i*|W1)* and q_i = *tf(*t_i*|W2)*. This measure also yields a score ranging from 0 to 1; unlike the original Cosine-distance measure, higher scores indicate higher possibility of topic shift.

5.3.5 Smoothing the Association Scores

To smooth the association scores to remove small dips, we used a simple method that replaces the original score with an average of that score with its previous and next neighbour. The result is that noise is reduced while the larger peaks and troughs remain (although slightly smaller). The thinner dark curve in Figure 25 shows the effect.

5.3.6 Detecting Story Boundaries

Like the TextTiling algorithms, story boundary identification is done by assigning a height score, the height of the peak (if one occurs), to each sentence (verse) gap. The height score corresponds to how strongly the cues for a subtopic changed on both sides of a window pair and is based on the distance from the valley on both sides of the peak to that peak. These height scores are illustrated as the horizontal bars in Figure 25.

The algorithm must determine how many segments to assign to a document, since every verse is a potential segment boundary. Hearst indicated that, any attempt to make an absolute cut-off, even one normalized for the length of the document, is problematic since there should be some relationship between the structure and style of the text and the number of segments assigned to it. She suggested making the cut-off a function of the characteristics of the depth (or height) scores for a given document, using the average (*A*) and standard deviation (*S*) of their scores (thus assuming that the scores are normally distributed). One version of this function entails drawing a boundary only if the depth (or height) score exceeds the value of *A* minus *S* (the liberal measure, LC). This function can be varied to achieve correspondingly varying Precision/ Recall trade-offs. A higher Precision but lower Recall can be achieved by setting the limit to be depth scores exceeding *A–S/2* (the conservative measure, HC).

In this study, we explored a range of cut-off values to examine the effect of cut-off levels, which are calculated using the following formula:

$$CO_n = A + (0.5 \times n - 2) \times S \qquad (1 \leq n \leq 7)$$

Equation 5

Since the text we used in this study is a collection of text documents, we computed the *A* and *S* based on the height scores for the whole collection (89 text files altogether) instead of a single document.

5.3.7 Role of Characters

We also explored the roles of the names of characters involved in the story segmentation task. Our observation is that a new story often starts by introducing new characters. As the story develops, the names are replaced by the corresponding pronouns in the text to smooth the narrative. It may be possible to improve story segmentation by giving additional weight to the character names. In the evaluation we introduced the parameter Choice of words (CW) to explore the role of characters with the following options:

- Ignoring names in the vocabulary (CW=1);
- Treating names and non-name words equally (CW=2); and
- Increasing the weight of names in the vocabulary over non-names – in practice we allocate double count to a name when it occurs (CW=3).

5.4 Evaluation

5.4.1 Hypotheses

Based on our research and discussion on the story segmentation theory, we designed this systematic evaluation to examine the performance of the TextTiling algorithms for narrative text and specifically Biblical text.

As explained in 5.3.4.2, when computing the association score between the two text windows, the symmetric measure should be more effective because it is based on the comparison of the probability distributions of the two text windows, thus more theoretically sound than the ad hoc Cosine-distance measure. We devised the following hypotheses for this study:

Hypothesis H1: The performance of the symmetric Jensen difference divergence measure is better than the Cosine-distance measure.

Hypothesis H2: The performance of the asymmetric Kullback-Leibler divergence measure is better than the Cosine-distance measure.

Because names are prominent for story segmentation, we also devised the following hypothesis:

Hypothesis HN1: Giving names more weight than normal words (CW3) is better than treating them equally (CW2).

Hypothesis HN2: Treating names and normal words equally (CW2) is better than ignoring them (CW1).

As we observed that when a story is long, there might be vocabulary changes in the middle of the story (see Figure 23, comparing the window 1 – 2 with the window 5 – 6). We suspect that the size of the text windows is an important factor which interacts with the length of the stories in story segmentation. The effectiveness of the size of window and the levels of cut-offs is also to be explored by the following hypotheses:

Hypothesis HC: The cut-off option affects the performance of the story segmentation method.

Hypothesis HW: The window size affects the performance of the story segmentation method.

Hypothesis HL: The window size affects the length of the detected stories. The smaller window size would result in the detection of more story breaks, and hence produce shorter stories.

5.4.2 Experimental Design

We used a 3 × 7 × 7 × 3 *between groups factorial design* to evaluate the effectiveness of four factors on the story segmentation task of 89 documents (see 5.4.4). The four factors are: Similarity Measure (SM), Window Size (WS), Cut-off Thresholds (CO) and Choice of Words (CW). The experiment yields 39,249 records: 89 documents (chapters) × 3SM × 3CW × 7WS × 7CO.

We segmented each of the text documents (chapters) in our collection using the story segmentation algorithms combined with each of the three similarity measures, Jensen, Kullback-Leibler and Cosine. We assigned story boundaries for each chapter, compared them with the story boundaries identified by human judges, and then computed the Precision, Recall and F-measures (See the following section for details). We also explored the affects of the following parameter settings:

Choice of Words (CW1 – CW3): this defines the words to be weighted in the dissimilarity score calculation (except stop words). In order to test the importance of the names of characters in the algorithm, we set up three options: CW1 considers normal words only, ignores names; CW2 considers all the words equally including normal words and names. CW3 weights names more than normal words by allocating names double counts.

Window Size (WS2 – WS8): this defines how many verses we use to form a window.

Boundary detection Cut-off (CO_1 – CO_7): this defines the cut-off value by which the story boundaries are chosen based on the dissimilarity scores.

In both the Hearst and Richmond experiments, minor errors are allowed when computing the Precision and Recall. In Hearst’s paper, a potential boundary is considered to be correct if it is within 3 token-sequences away from a paragraph break. In Richmond’s paper, every boundary was identified to within an accuracy of two sentences. We observed that in narrative texts there are usually one or two sentences connecting or smoothing the shift of stories, so we allow one verse error in our experiments.

5.4.3 Measures

In the experiments, we evaluate the performance of our algorithm by three indices: Precision, Recall, and F-measure. Precision and Recall measure segmentation accuracy for each chapter; they are defined as follows:

Precision (*P*) = no. of correctly estimated segment boundaries / total no. of estimated segment boundaries	Equation 6
Recall (*R*) = no. of correctly estimated segment boundaries / no. of true segment boundaries	Equation 7

The F-measure is computed using α=0.5 as a parameter, with a balance of the Precision and the Recall. F is computed using this formula:

$F(P,R) = \dfrac{PR}{\alpha R + (1-\alpha)P}$	Equation 8

5.4.4 Corpus and Ground Truth

Given our interest in narrative structures of story-telling text, we choose the four Gospels in The Holy Bible in the King James Version (KJV) as the text corpus. These four books have rich stories and characters and were written by four different authors with different narrative styles. The collection is made up with 89 individual text files; each represents a chapter in a particular book. Each line in the text file represents one verse in numeric order. The length of each chapter ranges from 444 to 1862 words. On average there are 1074 words per chapter. The lengths of verses are various. There is no story division and sub-heading in the KJV version.

We used the same books in the New International Version to generate the 'ground truth', where the story breaks and headings are placed by a group of scholars and editors. With permission from International Bible Society, we downloaded the NIV Bible (e-Resource 1995) book by book in html format. See 4.3.1 for an example of the HTML file. The total number of stories in the four books is 368. The number of verses in each chapter ranges from 17 to 80, with an average of 46 verses. The number of stories in each chapter ranges from 1 to 9, with an average of 4 stories.

We wrote a Perl program to parse the html files, i.e., to identify the book title and the chapter number (<h4> tags), the story title (<h5> tags), and the verse number (<sup> tags). In each book we extracted the information of chapter number, the story title, the number of starting verse, and the number of ending verse of each story, and save it into an index file. We then get an index of all the stories in the four Gospels and their locations. This approach of generating 'ground truth' is adapted in chapter 7 in different ways, to support the user interface design and user evaluation of the e-Book systems, which will be explained later.

As shown in Figure 27, the length of a story ranges from 2 to 42 verses, with an average of 10 verses. More than 64% of stories are between 4 and 12 verses long. The correlation between the length of chapter and the number of stories per chapter is statistically significant (r(89)=0.6, p<.001), showing that a longer chapter will have more stories than a shorter chapter. However, the length of a story is not particularly affected by the length of the chapter (r(368) = .08, p>.1).

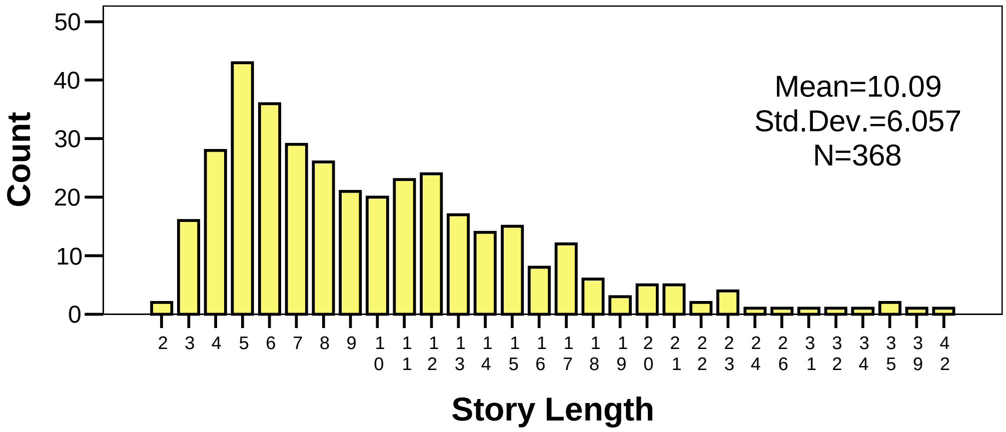


Figure 27 The Frequency of Story Length in the Gospels

The KJV Bible and NIV Bible match each other exactly in book title, chapter number and verse numbers. They only differ in vocabularies due to translation. In this study, the story segmentation results applied with the KJV Bible texts were compared with the story index extracted from the NIV Bible for measuring the systems' performance, in terms of the IR Precision, Recall and F-measure (previous section).

This corpus is quite different from Hearst's (1997) and the traditional collections used in the Topic Detection and Tracking evaluations. In our collection, all the documents, that is, all the chapters in the four Gospels together form one major theme and are in continuous narrative streams. Each chapter may have more than one story. Many of the stories are told in different books by different authors.

5.5 Results

5.5.1 Factors and Best Combinations

General Linear Model (GLM) with univariate analysis is employed to see the effectiveness of window size and cut-off levels towards the performance of each choice of words and similarity measures combination.

Table 4 GLM analysis of Between-Subjects Effects on F-measure

Source	df	F-value	Sig.
CO	6	166.916	.000
CW	2	9.950	.000
SM	2	38.576	.000
WS	6	17.104	.000
CO * WS	36	2.084	.000
SM * WS	12	7.009	.000
CW * WS	12	1.095	.359
CO * SM	12	.981	.464
CW * SM	4	.788	.533
CO * CW	12	.283	.992
CO * SM * WS	72	.253	1.000
CO * CW * SM	24	.052	1.000
CO * CW * WS	72	.090	1.000
CW * SM * WS	24	.215	1.000
CO * CW * SM * WS	144	.073	1.000

Results in Table 4 shown that the individual SM, CW, CO and WS affects the F-measure significantly ($p<.05$). The interactions between the window size and cut-off, and between the window size and similarity measures, are significant.

Thus we found evidence to support our hypothesis HC and HW, that the window size and cut-off were important factors in the TextTiling segmentation algorithm, when applied to narrative text. In the table, column one shows the factors and the interactions of the factors; column two shows the degrees of freedom (*df*); column three shows test statistics (F-value); column four shows the corresponding p-value of significance.

The best combinations for the mean of the F-measure are displayed in Table 5.

Table 5 Best Combination for F-measure

Similarity Measure	CW	WS	CO	Mean
Jensen	Normal	5	3	.604
Jensen	Double count names	5	3	.597
Jensen	Ignore names	5	3	.588
Cosine	Double count names	5	3	.584
Cosine	Normal	6	3	.580
Kullback-Leibler	Double count names	8	3	.580
Kullback-Leibler	Normal	7	3	.572
Cosine	Ignore names	6	3	.564
Kullback-Leibler	Ignore names	7	3	.561

The results show:

- Window size WS5 and Cut-off CO_3 achieves the best F-measure for the symmetric divergence measure Jensen;
- Using the asymmetric measure Kullback-Leibler, bigger window sizes performs better;
- Double counting names of characters (CW3) improved the performance of the similarity measures compared to ignoring them (CW1).

In Table 6 we present the mean value and the standard deviation (in brackets) of Precision, Recall and F-measure taken with cut-off CO3, and choice of words CW2, treating names as normal words.

Table 6 Mean Precision Recall and F-measure for CO_3, CW2, and WS5 – WS8

WS	Similarity Measure	Mean (Std. Deviation)		
		Precision	Recall	F-measure
5	Jensen	.600 (.260)	.677 (.255)	.604 (.220)
	Kullback-Leibler	.553 (.262)	.616 (.239)	.554 (.213)
	Cosine	.580 (.256)	.623 (.245)	.574 (.222)
6	Jensen	.598 (.279)	.577 (.251)	.555 (.227)
	Kullback-Leibler	.594 (.249)	.593 (.217)	.563 (.182)
	Cosine	.625 (.260)	.598 (.234)	.579 (.206)
7	Jensen	.642 (.283)	.564 (.254)	.566 (.227)
	Kullback-Leibler	.627 (.266)	.584 (.243)	.573 (.209)
	Cosine	.602 (.257)	.556 (.250)	.549 (.212)
8	Jensen	.668 (.274)	.557 (.235)	.576 (.208)
	Kullback-Leibler	.628 (.277)	.547 (.231)	.550 (.208)
	Cosine	.642 (.281)	.530 (.243)	.550 (.216)

In the following analysis we will use the F-measure as the dependent variable as it is a trade-off of the Precision and Recall.

5.5.2 Similarity Measures (SM) Performance

We examined the performance of the similarity measures by comparing means of the F-measure using a one-way ANOVA. The results shown that the difference between the three similarity measures was significant (F=37.694, p<.001).

Using Tukey HSD post hoc tests, we found that, overall, each similarity measure is significantly different than the other two (p<.001).

Table 7 Tukey HSD Post-Hoc Test of SM on the Dependent Variable F-measure

SM	SM	Mean Difference	Sig.
Jensen	Kullback-Leibler	.021736	.000
Jensen	Cosine	.010268	.000
Kullback-Leibler	Cosine	– .011468	.000

Since the symmetric similarity measure Jensen out-performs the Cosine-distance measure significantly by the F-measure, we are able to reject the null-hypothesis of H1, showing that using symmetric divergence measure is more effective than the Cosine-distance measure, when calculating the association scores of text windows for the story segmentation purpose. However, the Cosine-distance measure performs better than the asymmetric Kullback-Leibler measure significantly, and we fail to reject the null hypothesis of H2.

It should be remembered that there is an interaction between the similarity measures and the window sizes and, therefore, the overall differences between the weighting functions are not consistent over the range of different window sizes. We would come back to this point later.

5.5.3 Choice Words

We compared the choice of words options using one-way ANOVA with post-hoc Tukey tests, shown in Table 8.

Table 8 Tukey HSD Post Hoc Test of CW on the Dependent Variable F-measure

CW	CW	Mean Difference	Sig.
Ignore Names	Normal	– .006372	.030
Ignore Names	Double Count Names	– .010999	.000
Normal	Double Count Names	– .004626	.155

Statistically, we found evidence that treating character names and normal content words alike (CW2) is significantly better than ignoring the names (CW1) (p<.05) , so we are able to reject the null-hypothesis of HN2. The results also shown that, overall, double counting character names (CW3) out-performs ignoring them (CW1) (p<.001).

However, the difference between CW2 and CW3 is not statistically significant. So we fail to reject the null-hypothesis of HN1, and find no evidence that characters are an important factor in the story segmentation task. A comparison of the best combinations in Table 5 (page 94) provides a further indication that CW3 is no better than CW2. Given this somewhat counter-intuitive result, we suspect that our way of treating the characters (simply doubling the frequency of names) may not be sufficient for the task.

5.5.4 Window Size and Story Length

In the illustration of the TextTiling concept in Figure 23 (page 73), although a true topic boundary is found at the sentence gap between 6 and 7 due to a sudden vocabulary change, there is also obvious difference between block 1 – 2 and 5 – 6. We suspect that changing the window size might cause the relocation of the story boundary in this case.

Using the best combination Jensen + count character as normal words + cut-off 3, we tested the correlation between the window size and the average length of stories predicted, which is generated by the length of chapter divided by the number of estimated story boundaries. Correlation between window size and estimated story length is significant (r=.543, p<.001), with 623 records (89 chapters × 7 window sizes).

Hence, there is evidence to support the hypothesis HL, that the window size affects the estimated length of story. For a relatively long story, there might be vocabulary changes in the middle of the story. In this case, a narrow window would cause the algorithm to insert additional story boundaries, thus producing shorter stories. On the other hand, a wide window might fail to detect genuine changes in the narrative and the estimated story length would be too long. This might explain why a mid-range window size, WS5, performs well in the best combination.

5.5.5 Window Size and Cut-off

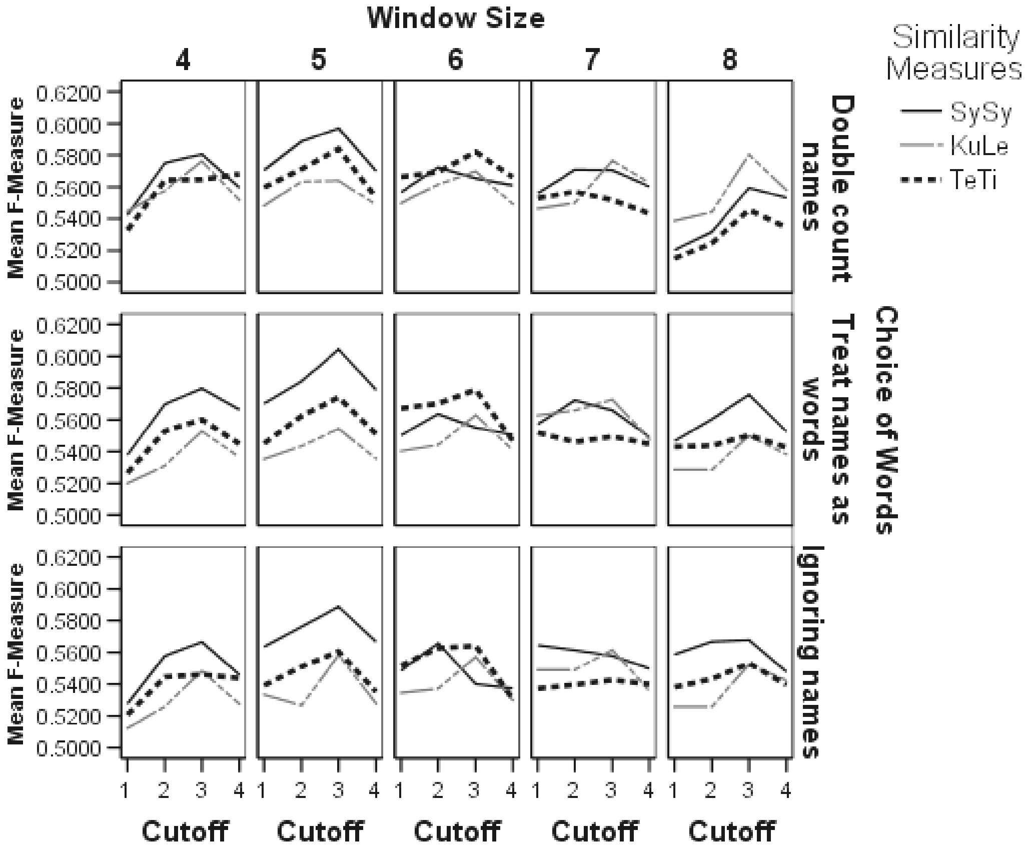


Figure 28 WS*CO*SM Combinations for CW3

Having looked at Table 5 (page 94) for the best combination and having noted that a significant interaction exists between window size and cut-off in Table 4 (page 93), we explored a wider range of window size and cut-off combinations. The diagrams shown in Figure 28 are produced with the window size ranges from WS4 to WS8 (column), and the cut-off ranges from CO_1 to CO_4 (x-axis). Different lines show the three different similarity measures. The y-axis shows the mean of the F-measure. Each diagram represents one of the choice of words options.

By inspection, the mean F-measures achieved by the Jensen difference divergence measure (Jensen) are at a higher level than the other two weighting functions in most situations. For Jensen, the window size WS5 and cut-off CO_3 achieves the peak F-measure in all three choice of words situations. This suggests that the optimal settings of window size and cut-off are less affected by the different choice of words options. When using the Kullback-Leibler measure, larger window sizes, i.e., either WS7 or WS8 combined with CO_3 achieve the peak F-measure in the three choices of words situations. When using Cosine-distance measure, the performances are similar with either WS5 or WS6 combined with CO_3.

Thus we find additional evidence to support our hypothesis HW and HC, that the window size and cut-off are important factors in the applying TextTiling to segmenting of narrative text.

5.6 Discussion

In this chapter we explored the TextTiling story segmentation algorithm for the purpose of the story segmentation in narrative texts, i.e., Biblical text. We improved the performance of original TextTiling significantly by using symmetric divergence measure to model the comparing text windows.

We conducted systematic evaluation to examine three different similarity measures that can be used in the algorithm: a symmetric Jensen difference divergence measure, an asymmetric Kullback-Leibler measure and the ad hoc Cosine-distance measure which was used in the original TextTiling algorithm. In practice we used the language modelling term probability instead of the original term frequency.

We have found statistical evidence that using the symmetric divergence measure works better than using the Kullback-Leibler measure and the Cosine-distance measure in the story segmentation algorithm. This might be explained by the fact that the symmetric divergence measure uses the available information to a greater extent than the other similarity measures.

We have also discovered that, while character names should be included, they are no more, or less important than normal content-bearing words in the story segmentation. We were unable to demonstrate that doubling the term frequencies of names is an effective way of using information about story characters. While names play an important role in story segmentation, there may be other factors, such as story length, that affect the performance of system. For example, suppose that a text contains a long story and segmentation is being performed using a small window size, the character names are introduced intensively at the beginning of the story, and are, thereafter, replaced by pronouns. In such a situation, we surmise that there is a high chance of unwanted story boundaries being placed in the middle of the story, where the names are replaced by their pronouns. To address this problem, it may be necessary to use pronoun resolution to replace pronouns with their referents.

In addition, the performance of the story segmentation algorithm is affected by the text window size and the cut-off level. Further, the length of detected stories is affected by the size of window used.

5.7 Summary

The experiments demonstrated that the TextTiling technique is effective in partitioning narrative book chapters into non-overlapping stories (narrative segments) in Biblical text, and the performance of the story segmentation task can be improved by using the symmetric divergence measure. We can then use these story segments as the basic units, to identify stories related by content, and thus link the related stories in the e-Book applications. This is an important step in semi-automatic detection of the narrative structure of e-Book.

6

NARRATIVE STRUCTURE DETECTION AND EVALUATION

6.1 Introduction

In the previous chapter, we designed and evaluated methods that sub-divide the semi-structured narrative e-Book into story segments. The key assumptions of this chapter are:

- The text collection is organized by topics or stories (we achieved this step in previous chapter).
- If two stories are topically related, or linked, they are likely to share a good proportion of common keywords.
- Presenting related stories for e-Book readers could help them discover the narrative structure of the book and understand the story and themes in the book better.

The objectives of this chapter are:

- To design algorithms to semi-automatically detect the narrative structure of an e-Book, and
- To evaluate such algorithms with narrative e-Book corpus.

In general terms, we will use information retrieval technique to match similar story segments in the corpus, link related stories with a narrative thread, and build a similarity network for all the narrative segments in the collection. We will also design evaluation methods to evaluate the performance of the algorithms and report the results in details.

6.2 Review of Previous Works

The work reported in this chapter has its roots in various other areas of work in the field:

- On automatic detection of subjects for e-Books, there is related work in book indexing (Salton 1988) and information extraction (Pazienza 1997);
- On the narrative thread detection, there is related work in topic detection and tracking;
- On linking related stories, there is related work in automatic hypertext generation.

Subject indexes were an important step forward for books because they enabled the comparison and correlations of information without extensive reading, re-reading and memorization. Chi, Hong et al. (2004) reported one e-Book system, ScentIndex that enhanced the subject index of an e-Book by conceptually reorganizing it to suit particular information needs. Users first entered information needs through keywords describing the concepts they were trying to retrieve and comprehend. ScentIndex then computed index entries that were conceptually related and displayed these index entries for the user. Chi, Hong et al. demonstrated that ScentIndex was faster and more accurate in assisting users with retrieving, comparing and comprehending information in the subject index of a book in comparison with a paper version.

Though sharing the same motivation of improving reading for comprehension, our work differs in that we are interested in creating an index of stories or topics, rather than a traditional subject index, which is mainly made of keywords and phrases. Subject indexes create relationships between subject descriptors such as words and phrases based on the text. We are trying to do something similar by building links of relationships between entire stories or passages within a text.

Topic Detection and Tracking (TDT) research has investigated finding new events and tracking existing events in a stream of textual broadcast news stories (Allan 2002). In TDT, the term topic is usually referred to that of an event. Carthy and Sherwood-Smith (2002) presented a novel technique that used lexical chains to represent document content in order to detect and track events more effectively. A lexical chain is a sequence of related words in a text, spanning short distances (adjacent words or sentences) or long distances (the entire text). In effect, a lexical chain is a list of words that captures a portion of the cohesive structure of the text. Each chain represents a sub-topical element of the text. The concept of the lexical chains was an important inspiration in our approach to narrative structure detection.

Although we did not use exactly the same methodology, both use vocabularies to determine the topic of text, and link related topics based on common vocabularies.

In 5.2.2, we briefly reviewed the TDT corpora. In TDT 2002 version, there are 20 different sources including ABC News, Associated Press, New York Times, etc. Each source might use the vocabulary differently. For example, the names of the sources, names of shows, or names of news anchors are much more frequent in their own source than in the other ones. Using this corpus, Chen, Brants et al (Chen, Brants et al. 2003) demonstrated a evaluation of story link detection and new event detection using information retrieval technique. Results shown that a symmetric clarity measure (Croft, Cronen-Townsend et al. 2001; Larvrenko, Allan et al. 2002) improved the story link detection, and optimizing story link detection was not equivalent to optimizing new event detection.

As hypertext systems became popular, various approaches were taken to converting existing documents and paper books into hypertext forms, such as TACHIR (Agosti, Crestani et al. 1995) and Hyper-TextBook (Crestani and Ntioudis 2001). In these hypertextual forms, models from information retrieval were applied to determine how to structure the information, i.e., probabilistic models of retrieval (Croft and Turtle 1989).

Our study followed a similar path, and moreover, led to the design of a tool for readers to help improve the effectiveness of linking similar and related topics. This tool was then evaluated with an e-Book corpus, real world simulated reading tasks, and a solid ‘ground truth’.

6.3 Methodologies

6.3.1 Algorithms

We designed a semi-automatic narrative structure detection algorithm for an e-Book corpus, shown in Figure 29.

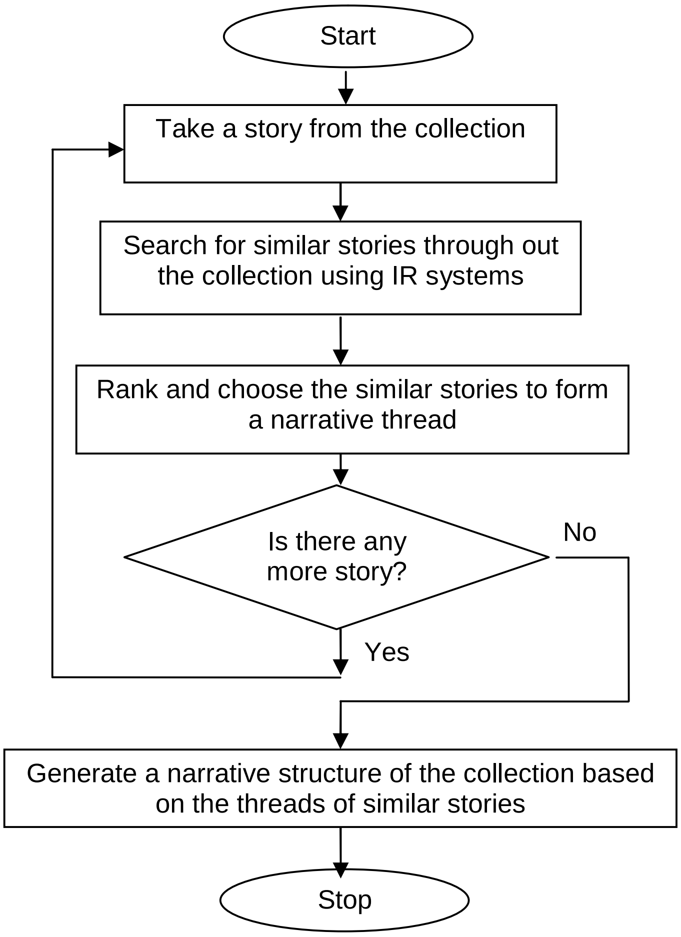

Start
Take a story from the collection
Search for similar stories through out the collection using IR systems
Rank and choose the similar stories to form a narrative thread
Is there any more story?
No
Yes
Generate a narrative structure of the collection based on the threads of similar stories
Stop

Figure 29 Flowchart of Narrative Structure Detection Algorithms

This algorithm took the assumption that if two stories are related, they should share a big proportion of common words. Thus a comparison of probability distributions of the two stories will be an indicator of the relatedness of these two stories. We can then measure the similarity of two stories to determine whether such link exists. Usually in Information Retrieval tasks the effective of system is affected by the distribution of terms both in the targeted document and in the collection as a whole, particularly when the collection is large and similar documents are many. In story link detection, we suspect that when a similarity measure is effective, whether two stories are related is not largely affected by other stories in the collection.

The processes are:

- For each story in the collection, comparing it with all other stories using a statistical language modelling approach, i.e., a simple 'bag of words' model, and generating a similarity score for each pair.
- Ranking the list of similar stories and generating a narrative thread for that story.
- Creating a narrative structure of the collection by linking each story with its most similar or related ones.

This generates a narrative thread network for each of the stories in the collection, which reveals the narrative structure of the e-Book.

Modelling the comparing stories is similar to modelling text windows using unigram word models, as described 5.3.4.1 (see next section).

A key insight is that similar or related stories often share common vocabularies. Therefore, a symmetric divergence measure would be effective in determining the relatedness between two story texts. In this chapter we present algorithms that employ the symmetric divergence measure to generate the narrative structure of an e-Book corpus, and evaluate its performance by comparing it with another IR query search system.

Our system is called Harmonizer, as it aims to 'harmonize' many narrative segments into topical narrative threads. This name is borrowed from the concept of a Gospel Harmony, which will be introduced in 6.4.3 in more detail.

6.3.2 IR Systems

6.3.2.1 Harmonizer

Modelling the stories is similar to the modelling of text windows (5.3.4.1), which uses a matching function to measure the association between two texts. In this chapter we model two stories using statistical language modelling – specifically a simple 'bag of words' model (Croft and Lafferty 2003; Fuhr 2001). We then rank related stories by computing the association score between these two stories.

Assume that two stories S_1 and S_2 are represented as probability distribution in n-space, thus, $S_1 = [p_1, p_2 \ldots p_n]$ and $S_2 = [q_1, q_2, \ldots q_n]$ where p_i is the probability of the term *i* in S_1, and q_i is the probability of the term *i* in S_2.

Ponte and Croft (1998) produced the following model (*M*) in (Equation 1, page 81):

$$P_{mle}(t_i \mid M(S_j)) = \frac{tf(t_i, S_j)}{dl(S_j)}$$

Equation 9

where $tf\ (t_i, S_j)$ is the term frequency of *i* in story S_j, and $dl\ (S_j)$ is the document length in words of story S_j.

To avoid the non-occurrence of a term causing a probability of zero for this term, we modified the formula by adding a parameter α to each term frequency. We set the parameter to 0.5, treating the non-occurrence term as if it was 'appeared' in the text *'half a time'* or *'less than once'*:

$$P_{mle}(t_i \mid M(S_j)) = \frac{tf(t_i, S_j) + \alpha}{dl(S_j)} \quad (\alpha=0.5)$$

Equation 10

There are other smoothing methods available, i.e., in Zhai and Lafferty's paper (2004).

Note that the dissimilarity between two stories is usually associated with two criteria: (1) sharing fewer common terms, and (2) differing in more terms. The dissimilarity between the two stories can be measured based on these two facts using symmetric comparison, which consider the two facts equally.

With a symmetric comparison, a dissimilarity score between two stories can be computed using the Jensen difference divergence measure (Taneja 2001). This is a symmetric divergence measure that measures the extent to which texts differ.

$$I(W1 \| W2) = \sum_{i=1}^{n} [\frac{p_i \log p_i + q_i \log q_i}{2} - \left(\frac{p_i + q_i}{2}\right) \log \left(\frac{p_i + q_i}{2}\right)]$$

(Taneja 2001)

This formula yields a score between 0 and 1. The lower the score, the higher the similarity between two stories compared.

6.3.2.2 Apache Lucene

To assess the Harmonizer approach, we used as a benchmark system a wrapper around Apache Lucene (Cutting, Bialecki et al. 2006). Lucene is a freely available open-source IR engine. We implemented Lucene in the following steps:

- Index the stories collection using Lucene Indexer class. This will produce a folder with index files for query search.

- Remove punctuation marks and parse each story as a query using Lucene QueryParser class.
- Search for related stories using Lucene Search class

Lucene generates a Hit list for each query (story) with all the stories in the collection ranked according to their similarity scores. We then chose related stories from the top ranked ones using a threshold (see next section).

Lucene performs ranked retrieval using a standard *tf.idf* model (Salton and McGill 1983; Salton and Buckley 1988), although it also supports a Boolean query language.

The *tf.idf* weight (term frequency vs. inverse document frequency) is a weight often used in information retrieval and text mining. This weight is a statistical measure used to evaluate how important a word is to a document in a collection or corpus. The importance increases proportionally to the number of times a word appears in the document but is offset by the frequency of the word in the corpus. Variations of the *tf.idf* weighting scheme are often used by search engines to score and rank a document's relevance given a user query.

By default, the similarity score of query *q* for document *d* is defined as follows:

$$score(q,d) = \sum_{t \ in \ q} \left(tf(t \ \ in \ \ d) \times idf(t)^2 \times B_q \times B_d \times L \right) \times C$$

$$\textit{where}$$

$$B_q = getBoost(t\ field\ in\ q)$$

$$B_d = getBoost(t\ field\ in\ d)$$

$$L = lengthNorm(t\ field\ in\ d)$$

$$C = coord(q,\ d) \times queryNorm(S),\ and$$

$$S = sumOfSquar\ edWeights = \sum_{t \ in \ q} (idf(t) \times B_q)^2$$

Equation 11

We generated this similarity measure from the Java API documentation of Lucene (Cutting, Bialecki et al. 2006; Gospodnetic and Hatcher 2005). Note that the formula is motivated by the *Cosine-distance* or normalized *dot-product* (Fuhr 2001) between the document and the query vector, which is an asymmetric similarity measure.

The *tf* computes a score factor based on a term or phrase’s frequency in a document. The *idf* computes a score factor based on a term’s document frequency (the number of documents which contain the term). The *tf* value is multiplied by the *idf* for each term in the query and these products are then summed to form the initial score for a document.

Terms and phrases repeated in a document indicate the topic of the document, so this method usually returns larger values when frequency is large, and smaller values when frequency is small. Terms that occur in fewer documents are better indicators of topic, so implementations of this method usually return larger values for rare terms, and smaller values for common terms.

In addition, the terms in Equation 11 can be explained as follows, also according to the API documentation (Cutting, Bialecki et al. 2006):

- *boost(t field in d)* gets the ‘boost’ for this query clause. Field boosts come in explicitly in the equation and are set at indexing time. The default value of field boost is 1.0.

- *lengthNorm(t field in d)* is the normalization value of a field, given the number of terms within the field. This value is computed during indexing and stored in the index.

- *queryNorm(q)* is the normalization value for a query, given the sum of the squared weights of each of the query terms.
- *coord(q, d)* computes a score factor based on the fraction of all query terms that a document contains.

This similarity measure yields a score between 0 and 1. The higher the score, the greater the similarity.

6.3.2.3 A Comparison of Harmonizer and Lucene

The main differences between the two systems are:

- Harmonizer used a symmetric divergence measure to compare two stories (6.3.2.1), while Lucene used an asymmetric one to compare a query and a collection of documents (6.3.2.2).
- Harmonizer used the maximum likelihood estimate (*mle*) to weight terms in the relatedness formula, while Lucene used simple term frequency (*tf*). In fact *mle* is a normalized form of *tf*, albeit interpreted as a probability.
- Lucene took into account the global distribution of terms in the entire collection (*idf*), based on the assumption that terms which occur in fewer documents are better indicators of topic, so implementations of this method usually return significant values for rare terms and insignificant values for common terms. Harmonizer does not consider this information.
- There are other parameters included in Lucene's default similarity measure which need to be examined further.

We illustrated an example of the story linking process in Harmonizer and Lucene in the following diagrams.

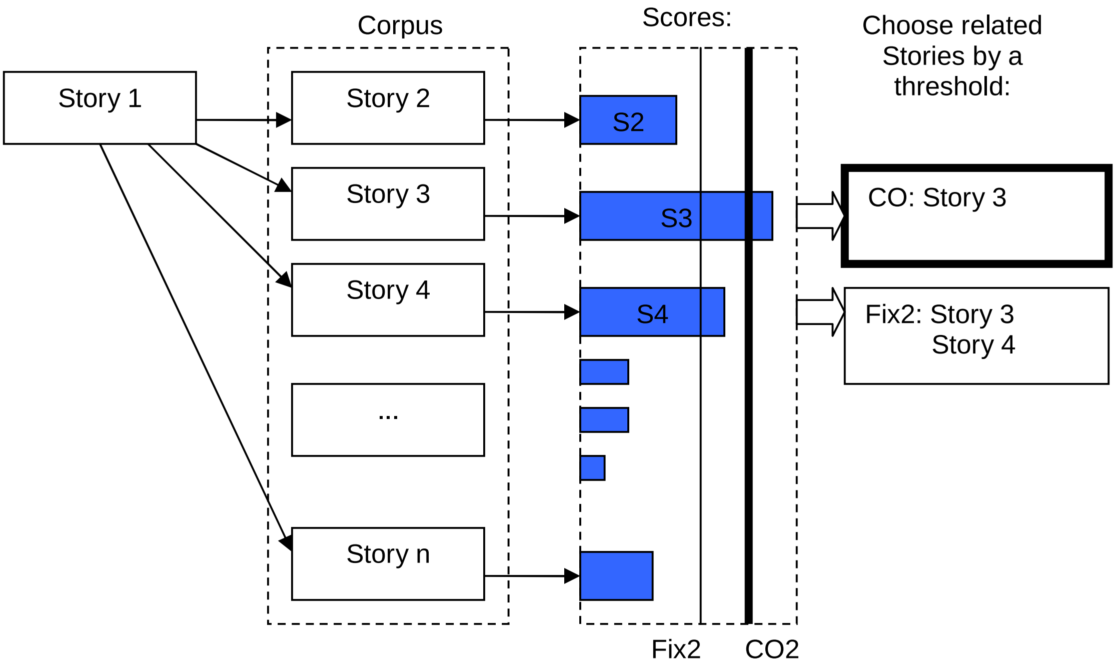


Figure 30 Harmonizer Story-Comparison and Linking Process

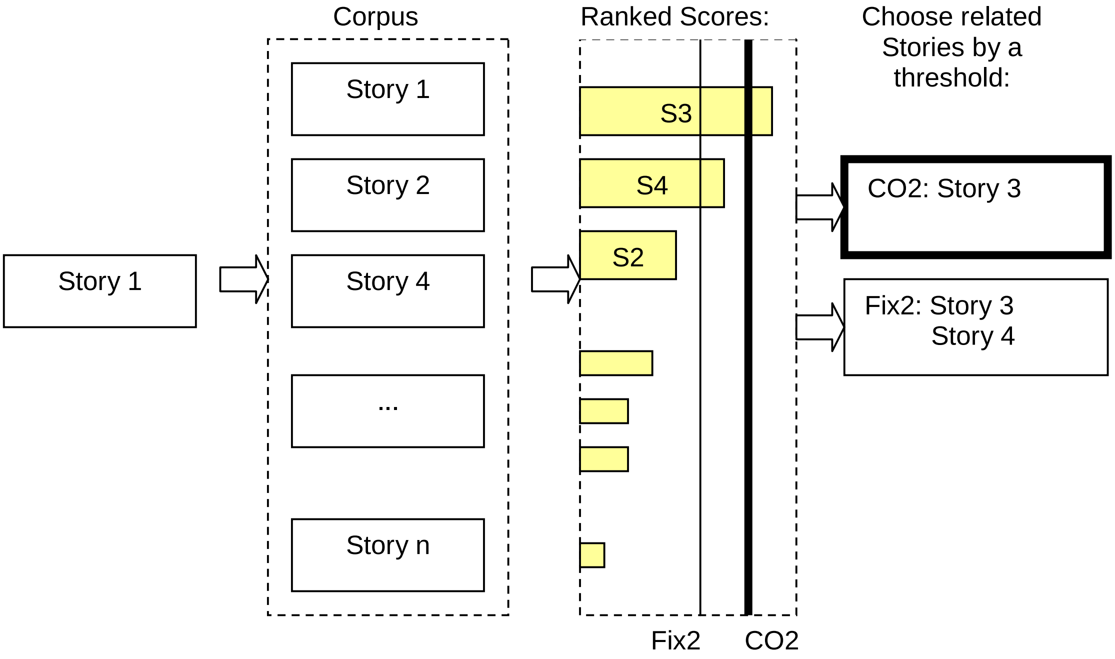


Figure 31 Lucene Story-Searching and Linking Process

Note the similarity scores computed by each system for each story are not likely or necessarily to be the same. Two thresholds CO2 and Fix2 were placed to select most related stories. More details on the thresholds will be discussed in 6.3.3. Using Lucene, the top ranked related story is always the story itself, i.e., Story 1 will be identified as most related story for Story 1 itself, in the above example. In practice we removed the self-score when select related stories for a story. Since the index of corpus is created in advance, it is not possible to use a different corpus index, i.e., one excludes the story to be searched, for each story. This might be an effect to the system performance, we suppose.

6.3.3 Story Link Detection

When a story is compared with all the other stories in the collections, there will be a similarity score computed for each pair. The identification of related stories is performed by ranking the similarity scores and selecting the stories that are top ranked. This can be implemented in two ways:

- Placing a variable cut-off threshold, see below. Or
- Placing a fixed threshold by a fixed number of top ranked stories.

See the vertical lines CO2 (a cut-off variable threshold) and Fix2 (a fixed rank threshold) placed on the scores as shown in Figure 30 and Figure 31.

A variable function can be tuned to achieve correspondingly varying Precision/ Recall trade-offs. In this study we computed a cut-off variable by combining the average (*A*) and standard deviation (*S*) of the scores for each story thread, and explored a range of thresholds using a parameter (which varied between 1 and 7) to adjust the cut-off. The range of cut-off thresholds was calculated using the following formula:

$CO_n = A + (0.5 \times n - 5.5) \times S \qquad (1 \le n \le 7)$	Equation 12

where *A* indicates the average of similarity scores and *S* indicates the standard deviation.

For easy comparison, we call this threshold calculation ‘cut-off’ and the fixed rank threshold ‘fixed numbers’ in the evaluation and data analysis.

Note that in Lucene, the top ranked story in the collection is the story used as a query. We excluded it before placing a cut-off threshold since a story is not related to itself.

6.4 Evaluation

6.4.1 Hypotheses

In general terms, we wanted to investigate whether the semi-automatic narrative structure detection system Harmonizer is more effective for identifying related or similar stories in the collection, than a competing tool based on functionality drawn from an IR query search system, Lucene. The main aim of the study is to compare the effectiveness of the two systems in detecting similar stories in a collection, in order to map out the narrative structure of the collection.

In previous chapter, we evaluated the effectiveness of three similarity measures used to compare text windows in the story segmentation tasks. Our results showed that a symmetric divergence measure is more effective than the other two asymmetric ones as measured by IR Precision, Recall and F-measure. We suspect the symmetric measure can be effective also in the narrative structure detection task, as this task also requires comparing the distributions of terms in two documents (stories).

It seems like that the both system has pros and cons, thus we conjecture that the effective of system will be comparable for Harmonizer and Lucene for identifying similar or related stories in a collection of stories, as measured by IR Precision, Recall, F-measure.

Hypothesis HP: Harmonizer has greater Precision (IR term) than Lucene for identifying similar or related stories in a collection of stories.

On the other hand, Lucene takes into account the global distribution of terms in the entire collection, based on the assumption that terms occur in fewer documents are better indicators of topic. This might create a bias in the IR Recall measure when the number of truly related stories in the collection is considered, possibly at the expense of Precision. However, it is possible that Harmonizer might achieve comparable levels of Recall. Our hypotheses are as follows:

Hypothesis HR: Lucene has greater Recall (in IR terms) than Harmonizer when identifying similar or related stories.

Hypothesis HF: Supposing that hypothesis HP and HR hold, then the effectiveness, as measured by the combined F-measure, will be comparable. This hypothesis is simply a consequence of the fact that the F-measure ‘trades off’ Precision against Recall.

Considering the way of selecting the top ranked similar stories, we conjecture that the level of threshold applied may affect the performance of both systems. In this evaluation, we explored 7 levels of threshold. In the cut-off option, the level of thresholds 1 to 7 represent cut-off variables with parameter *n* ranges from 1 to 7. In the fixed number threshold, these levels simply mean counting the top 1 – 7 stories from the ranked list.

6.4.2 Experimental Design

We used a 2 × 7 *between groups factorial design* to test the hypotheses. There are two independent variables (i.e., factors): the systems, and the level of threshold. We run the experiment twice, using the two choices of threshold. Each experiment yielded 4760 records (340 stories × 2 systems × 7 levels of thresholds).

In the evaluation, the two systems compared were:

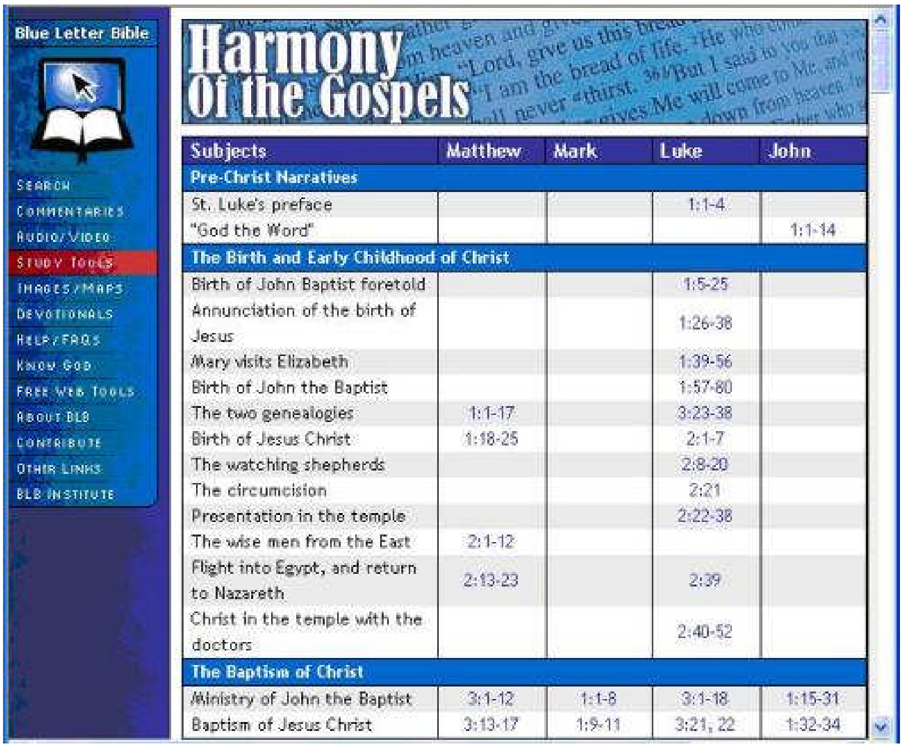
- Harmonizer, a language modelling IR system using the symmetric Jensen difference measure (Taneja 2001) that measures the association of two probability distributions.
- Lucene, an IR system, whose similarity measure is motivated by the Cosine-distance measure or dot-product between the normalized document and query vectors.

We ran the semi-automatic narrative structure detection algorithm with the Harmonizer system, found related stories for each story, computed Precision, Recall, and F-measure (see details later) based on a ‘ground truth’, and then compared these results with results achieved by using Lucene.

We also explored the effect of the level of threshold options in selecting the top ranked similar narrative segments, using a cut-off level which is calculated by the average and standard deviation of the similarity scores of each story thread, and using a cut-off level which is determined by a fixed rank number from the top. However, due to the fact that the two thresholds differ significantly in selecting methods and the levels between them are incomparable, we will report the results separately.

6.4.3 Gospel Harmony and Collection Creation

We choose stories in the four Gospels in the Holy Bible in King James Version text as our text corpus, and a Harmony of Gospels as the ground truth. Under one main theme (*the teaching and life of Jesus Christ*), the four Gospels have rich stories and characters, and were written by four different authors with different narrative styles.



Subjects	Matthew	Mark	Luke	John
Pre-Christ Narratives				
St. Luke's preface			1:1-4	
"God the Word"				1:1-14
The Birth and Early Childhood of Christ				
Birth of John Baptist foretold			1:5-25	
Annunciation of the birth of Jesus			1:26-38	
Mary visits Elizabeth			1:39-56	
Birth of John the Baptist			1:57-80	
The two genealogies	1:1-17		3:23-38	
Birth of Jesus Christ	1:18-25		2:1-7	
The watching shepherds			2:8-20	
The circumcision			2:21	
Presentation in the temple			2:22-38	
The wise men from the East	2:1-12			
Flight into Egypt, and return to Nazareth	2:13-23		2:39	
Christ in the temple with the doctors			2:40-52	
The Baptism of Christ				
Ministry of John the Baptist	3:1-12	1:1-8	3:1-18	1:15-31
Baptism of Jesus Christ	3:13-17	1:9-11	3:21, 22	1:32-34

Figure 32 Screenshot of the Gospel Harmony

All four Gospels in the New Testament of the Bible tell the story of Jesus Christ. Each book – Matthew, Mark, Luke, and John – stands alone, emphasizing a unique aspect of Jesus’ life. But when these are blended into one complete account, or harmonised, we gain new insights. There are different versions of the Gospel Harmonies available, we used the one written by Nevin and Alfred (2002) and downloaded from the Blue Letter Bible website (e-Resource 2002), see Figure 32. This harmony combines the four Gospels into a single chronological account of Christ's life on earth. It includes every chapter and verse of each Gospel, leaving nothing out.

The creation of the 'ground truth' and the text collection followed three steps:

- With permission from the author (Nevin and Alfred 2002), we copied the entire Harmony table into Microsoft Excel spreadsheet, removed the group headings, and converted the table into a Comma Separated Values (*csv*) file format.
- We wrote a Java program to parse the *csv* file and created an index of stories in the four Gospels. This index had information for each story: the story title, the book, the chapter and verse numbers, and a list of similar or related stories in other three books, etc.
- Using the information from this index, we generated the story text from the King James Version of the Bible, saved each story as an individual text file, and created a collection of stories.

This index served as a 'ground truth' for our story comparison evaluation. In the Gospel Harmony, 340 stories out of 426 are interlinked with at least one other story, thus mentioned in at least two books. These 340 stories were chosen as narrative anchors in the experiments, while the rest were also included in the corpus. We compared each story with every other to find the similar stories in the collection, then ranked the similar stories, and selected the most strongly related stories to form a narrative thread. The results were compared to the original index for IR effective measures: Precision, Recall and F-measure (see below).

We did not use all the 426 stories as anchors (or queries, for Lucene), because, for those 86 stories that did not have any related stories according to the ground truth, the Precision was zero, the Recall and the F-measure could not be computed. Arguably that there might be other ways to improve these measurements, for example, in (Sebastiani 2002). This could be an interesting topic for future development in the story linking field, since the story linking tasks are not quite equivalent to IR tasks, the related stories are not quite equivalent to relevant documents in IR. We will come back to this point later.

This index of Gospel Harmony was adapted again (see 7.1.4) to generate a ‘ground truth’ for part of the subsequent user evaluation purposes.

6.4.4 Measures

Precision and Recall measure accuracy of related stories identified for each story. F-measures is computed using *α=0.5* as a parameter, with a formula to balance effect of Precision and Recall. They are defined as follows:

Precision (P) = number of correctly identified related stories / total number of identified related stories.	Equation 13
Recall (R) = number of correctly identified related stories / number of truly related stories.	Equation 14
$F(P,R) = \frac{PR}{\alpha R + (1-\alpha)P}$	Equation 8

6.5 Results

6.5.1 Precision, Recall and F-Measure

Since we used a between group factorial design, we employed the General Linear Model (GLM) univariate tests to analyse the results. System and level of threshold were treated as fixed factors and the stories were treated as random factor. In this way the power of effect of system will be emphasized.

Table 9 – Table 11 shows the mean, the standard deviation, the F-value, the significant p-value, and the interactions between system and the level of threshold, for each of the Precision, Recall and F-measure.

Table 9 Factorial Analysis of Precision

Choice of threshold	Effects	Mean (Std. Deviation)		F-value	Sig.
		Lucene	Harmonizer		
Cut-off	System	.454 (.354)	.534 (.410)	156.75	.000
	Level			602.69	.000
	System * Level			20.37	.000
Fixed Number	System	.448 (.321)	.457 (.320)	3.25	.071
	Level			1012.161	.000
	System * Level			.522	.792

Table 10 Factorial Analysis of Recall

Choice of threshold	Effects	Mean (Std. Deviation)		F-value	Sig.
		Lucene	Harmonizer		
Cut-off	System	.784 (.348)	.738 (.376)	67.49	.000
	Level			179.47	.000
	System * Level			17.464	.000
Fixed Number	System	.732 (.364)	.746 (.354)	8.889	.003
	Level			529.824	.000
	System * Level			.62	.715

Table 11 Factorial Analysis of F-measure

Choice of threshold	Effects	Mean (Std. Deviation)		F-value	Sig.
		Lucene	Harmonizer		
Cut-off	System	.510 (.328)	.540 (.364)	28.36	.000
	Level			464.80	.000
	System * Level			23.23	.000
Fixed Number	System	.493 (.272)	.502 (.268)	4.10	.043
	Level			284.206	.000
	System * Level			.382	.891

Using the cut-off threshold, the difference of means between systems was significant for each of the three effectiveness measures. Harmonizer was significantly more effective than Lucene as measured by Precision (p<.001); Lucene was significantly more effective than Harmonizer as measured by Recall; and the Harmonizer was significantly more effective than Lucene as measured by the F-measure.

When fixed number thresholds were used, the difference between systems was not significant according to Precision. However, the Harmonizer is more effective than Lucene as measured by Recall (p<.01) and F-measure (p<.05).

Overall, we have found evidence to reject the null hypothesis of HP, showing that the Harmonizer was more effective than Lucene for detecting narrative structure, as measured by Precision.

Since the results of Recall using cut-off threshold and fixed number thresholds conflicted with each other, we could not find evidence to reject the null hypothesis of HR. Harmonizer is probably as effective as Lucene, measured by Recall.

The F-measure results provided evidence for our hypothesis HF, namely that overall effectiveness of Harmonizer and Lucene was comparable by the F-measure, for the narrative structure detection task. In spite of different performance on Precision and Recall, the Harmonizer was overall more effective than Lucene as measured by the F-measure, regardless of which cut-off option was used.

Our results indicated that the symmetric divergence measure, as implemented in the Harmonizer system, was more effective than IR search engine Lucene, using its default similarity measures, and moreover this effectiveness was achieved with significant improvements as measured by Precision and the F-measure.

6.5.2 Effect of the Cut-off Thresholds

Next, we analysed the effect of the levels of threshold (LT) on the performance of each system. Table 9 – Table 11 (see page 121 to 122) also shows that the level of threshold affected the system performance significantly (p<.001), according to Precision, Recall and F-measure, regardless of which threshold method used. The interaction between the system and the level of thresholds are significant for cut-off options (p<.001), but not so for fixed number options.

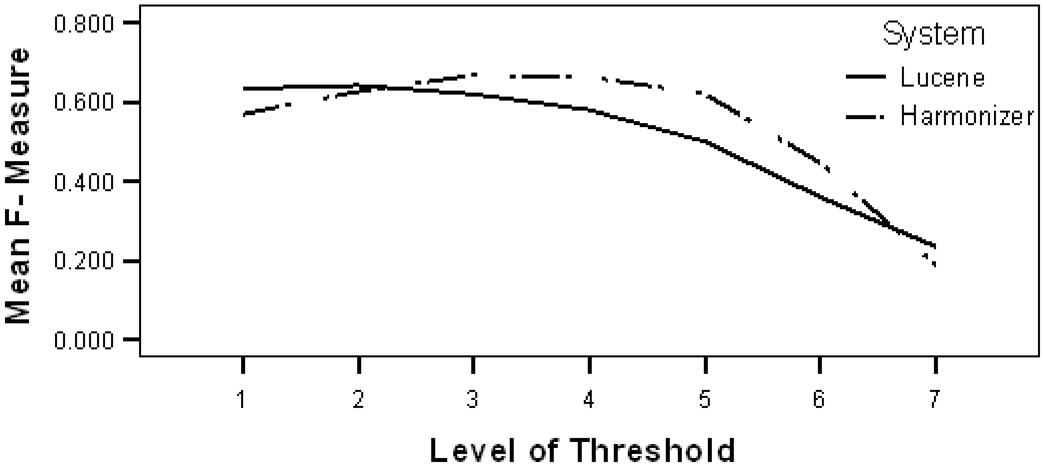


Figure 33 Line Chart of the Interaction between System and Cut-off Thresholds

Figure 33 shows the interaction effect of the system and the level of threshold in a line chart. Different lines show the two systems. The y-axis shows the mean F-measure and the x-axis shows the seven levels of threshold. With the cut-off thresholds, Lucene performed better with cut-offs level 1, but Harmonizer performed better with higher cut-offs (i.e., level 3 to 6). The two systems met at almost same value when the cut-off level is 2, and they both dropped to bottom at level 7.

Note that there is a platform trend in Harmonizer using cut-off 2-5. This might be explained that the difference between each level of the cut-off variables in Harmonizer was tiny; and/or the similarity scores of related stories were obvious distinguish from other stories, that even after changed the cut-off threshold level, the Harmonizer still selected the same stories. See Figure 30, page 113 for the diagram of scores in the middle of the figure.

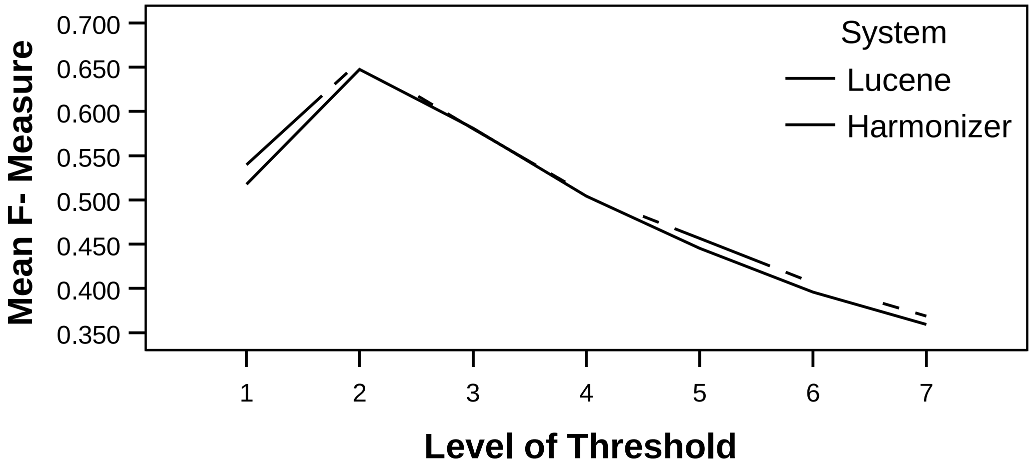


Figure 34 Line Chart of Interaction between System and Fixed-number Thresholds

With the fixed number thresholds, there is no difference between Lucene and Harmonizer. However, both systems reached their best performance with the rank number 2, and their worst at rank number 7.

The following figure shows the frequency of the number of related stories in the ground truth table.

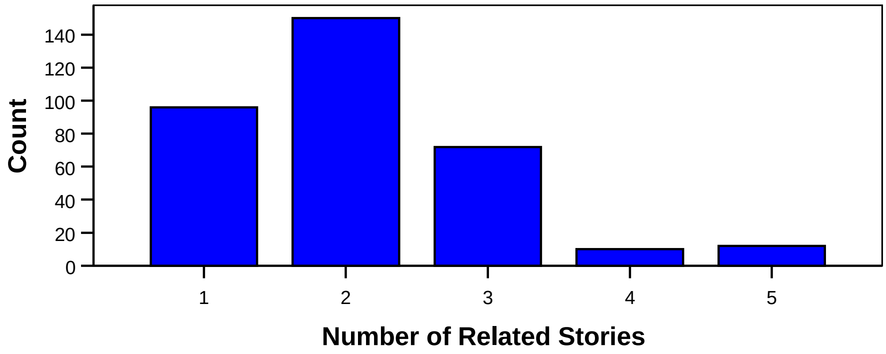


Figure 35 Number of Related Stories in the Ground Truth

A good amount of stories in the collection have only one or two related stories. This might provide an explanation of why the fixed thresholds reached its best at level 2 - when chose 2 top ranked stories.

6.6 Discussion

We examined the performance of the semi-automatic narrative structure detection algorithms with two IR implementations: Harmonizer and Lucene.

The evaluation results showed that Harmonizer was more effective at identifying similar or related stories than Lucene, as measured by F-measure.

On Precision measure, using cut-off thresholds Harmonizer achieved better Precision than Lucene; using fixed number thresholds, Harmonizer was as good as Lucene. This might be explained by the fact that Harmonizer used a symmetric divergence measure when comparing two texts, while Lucene used an asymmetric one.

On Recall measure, Lucene was more effective than Harmonizer using cut-off thresholds; but Harmonizer (mean=.732) was better than Lucene (mean=.746) using the fixed number thresholds, although the difference was tiny (F=8.889, p<.01). Although we suspect the cut-off threshold is more reliable than the fixed numbers, we could not demonstrate that Lucene is Recall-oriented overall.

On average the two systems' performance on Precision (mean≈.47) is not as good as its Recall (mean≈.75). This might be explained by that the 'ground truth' we used, the Gospel Harmony, being strictly accurate to events. For example, the two events of *Jesus Feeds the Five Thousand* (Matthew 14:13 – 21, Mark 6:30 – 44, Luke 9:10 – 17 and John 6:1 – 15) and *Jesus Feeds the Four Thousand* (Matthew 15:29 – 39 and Mark 8:1 – 9) are considered as 'unrelated' or 'dissimilar' stories in the 'ground truth', while both system are term-centred and would not precisely detect such differences.

We also found that there is a strong interaction between the system and level of threshold in the cut-off formula option, but not in the fixed number option.

However, since there are many parameters in Lucene, we are limited in this study to find the affects of each of the parameters, i.e., query boost, etc. Also, we are not sure about the impact of long queries (whole story) on the performance of Lucene, since Lucene was designed for query search, which is normally in the form of a few keywords, not for story comparison. It will be interesting to explore further why Lucene's '*idf*' model fails to achieve a more effective Precision and Recall in this experiment. It might be affected by the corpus and ground truth we used. Further experiment will be needed to explore these factors in system performance, i.e., use the entire Bible as corpus.

We also observed that normal IR effective measures like Precision, Recall and F-measure might not be ‘effective’ enough in the evaluation of the story linking algorithms. We should find effective ways to evaluate system’s performance even on those stories that did not have apparent related stories in the ground truth.

6.7 Summary

In this chapter we described two general algorithms for semi-automatic detection of narrative structure in e-Books, and evaluated them with a real narrative e-Book collection. We designed the evaluation methods and developed a ‘ground truth’ to measure the performance of the algorithms. The evaluation results shown that the semi-automatic narrative structure detection algorithms identified related stories effectively. This algorithm is particular powerful when combined with a symmetric divergence measure, as compared with a baseline IR search system.

Although we discovered a few issues that needed further studies, we have met our objectives of this chapter for the purposes of semi-automatic detection of narrative structure of e-Books. The evaluation results provided a good foundation for using the narrative structures as scaffolding for e-Book readers. In next chapter, we will describe the user evaluation results of the e-Book user interfaces separately powered by Harmonizer and Lucene, and discuss the effect and influence of these systems on readers’ comprehension activities.

III EVALUATION, DISCUSSION AND CONCLUSIONS

7

EXPERIMENTAL DESIGN AND USER EVALUATION

7.1 Introduction

In previous chapters, we discussed the importance of information seeking, concept mapping, and contextual cues in people's reading for comprehension, and presented a design for an e-Book user interface that integrates searching, linking and browsing. Further, we reported in detail technologies for dividing semi-structured text into topical segments, semi-automatically detect the narrative structure of an e-Book, and automatically detect and link similar stories and passages. Would such an e-Book user interface help the reader's comprehension? If yes, to what extent would it be helpful?

In this chapter, we will describe an evaluation of the e-Book's interface under two conditions, one with story linking features that are powered by the Harmonizer (6.3.2.1) and the other with query search features, powered by Lucene (6.3.2.2), while other navigation functions are held the same. We then report the methodologies and results of the experiments.

7.1.1 Previous Work

Researchers in information seeking and information retrieval have investigated people's information-seeking behaviour to a great extent using different evaluation methods, for example, using system logs of navigation nodes (Heinstrèom 2002; Park and Kim 2000), reading time (Kelly and Belkin 2001), queries length (Belkin, Kelly et al. 2003), using questionnaire (2004; Voorhees and Harman 2005), using questions (Crestani and Ntioudis 2001) and essays writing (Halttunen and Järvelin 2005), using simulated tasks (Harper, Koychev et al. 2003b), and using eye-tracking capture facilities (Granka, Joachims et al. 2004), etc. Although most of this research focused on improving search engines, it also produced incidental results that aided reading and scanning. To assist people's reading with information-seeking tools is an important challenge.

Crestani and Melucci (2003) demonstrated an automatic constructed hypertext system, Hyper-Textbook, for self-referencing. They used both closed and open questions in the system evaluation. 30 participants used three systems in a between-groups experiment where the effectiveness of Hyper-Textbook was compared with an online version and a paper version of the same book. Closed questions were supplemented by a list of four answers of which only one was correct. Open questions required finding pages containing relevant information and writing a few paragraphs of text. Questions covered terms occurring in the subject index, the table of contents, or in paragraph titles. Each participant had a time limit to answer each of seven questions.

Similar experiments were carried out by Crestani and Ntioudis (2001), in which they compared two alternatives, Hyper-Textbook and the paper book. They used the accuracy of the answers, the speed of task completion, and the users' opinion on the assistance of each system as evaluation measures. However, in both papers the method used to evaluate the accuracy of questions was omitted. It is generally much easier to evaluate closed questions than open questions.

In the evaluation of ProfileSkim (Harper, Koychev et al. 2003), we used a back of book index to create simulated information seeking tasks. Participants were given two user interface conditions and a limited time to find page numbers in a hypertext book for given topics, which were semi-randomly chosen from the index. These tasks were typical for book authors and editors when they create a subject index, and for readers when they need to use an index to find relevant pages. They were effective in demonstrating the systems' functionalities. The evaluation of task performance was based on the original book index, which played the role of 'ground truth', and the time to complete a task, which was recorded by participants themselves on the task sheet. The performance of these tasks demonstrated the effectiveness of the system in assisting users to achieve their goals.

7.1.2 Systems

7.1.2.1 iSee and iSearch

We designed the e-Book user interface with two searching conditions, one with the Story Linking feature (iSee, Figure 21, page 68), the other one with query search function (iSearch, Figure 22, page 69). For system evaluation purposes, we described the iSee system to the participants as ‘Blue’, and the iSearch system as ‘Orange’, to prevent participants from guessing the functions and features of the system.

The iSee user interface has a mini-table of contents which lists story titles in the chapter; it provides a story searching function by clicking on the story title; related stories are pre-retrieved using language modelling techniques.

The iSearch user interface has a traditional query box and a search button; stories are retrieved using queries based on the Lucene free-text search technology.

7.1.2.2 Comparison of iSee and iSearch

The main similarities between the two systems are:

- Both e-Book user interfaces display the NIV Bible (4.2.2) in HTML format and supported same function of browsing through the interactive visualization tools;
- Stories were indexed in advance to obtain information of story titles and breaks;
- Both used ranking and cut-off thresholds to reduce or filter the number of related stories retrieved.

At first sight the two user interfaces are very similar. The reading and browsing functions were made equivalent in iSee and iSearch. In spite of the similarities in appearance, there are important dissimilarities lying under the surface, and few key differences on the user interface:

- The iSee user interface has a mini-table of contents which lists story titles in the chapter (See Figure 21, page 68 for the Story Panel of the iSee user interface). It provides story searching function by clicking on the story title; related stories were pre-retrieved using Harmonizer. In contrast, the iSearch user interface has a traditional query box and a search button; queries are retrieved using Lucene technology.

- The interaction for searching is different. To search for related stories of a story displayed in the reading panel, one step is needed in iSee: clicking on the story title in the mini-table of contents; two to four steps are needed in iSearch; they are, clear the query box if it is not empty, type or copy and paste query terms into the query box, then press the Search button.
- The text search engines are different. iSee is powered by Harmonizer using language modelling technology with a symmetric divergence measure. Narrative structure of the e-Book was discovered in advance, and each story was matched in advance for a list of related stories (chapter 6). On the other hand, iSearch is powered by Lucene. Users need to write their own queries. As a result, the result list is different. iSearch (Lucene) returns more results (Hits) than iSee.
- These systems differ in flexibility: iSee does not accept query search. iSearch accepts any number of words query search.
- iSee is closer to hypertext books; it integrates browsing stories and searching for stories with navigation of the organizational structure of a book. iSearch is closer to query search tools.

7.1.3 Hypotheses

A controlled, within-subjects laboratory user experiment was conducted to compare two e-Book user interfaces (see section 7.3 for details in experimental design). System and task orders were rotated and counterbalanced using a Latin square design to ensure every possible ordering of conditions is occurred equally in the experiments.

The main difference between the two interfaces is that the one provides a story linking tool to support readers browsing for similar stories, and users will not need to generate and formulate queries themselves if they are looking for similar stories or passages; the other provides a query search box for readers to search for similar stories manually.

In this study we define ‘effectiveness’ as ‘how well a task is done’, and ‘efficiency’ as ‘how quickly a task is done’. ‘Comprehension’ and ‘understanding’ are used in their commonsense interpretations, not as scientific or psychological terms.

We proposed the following hypotheses as the basis for the comparative study at the core of the evaluation.

Hypothesis HEI: an e-Book user interface with a story linking tool is more effective than one without, to supporting a user’s information seeking task. Effectiveness will be tested by measuring the number of navigation actions, the number of searches, and the number of pages viewed, as recorded by the system logs. In addition, the number of attempted answers and the proportion of completed answers will be calculated to support the test; the number of correct answers, and their precision, will be assessed by a comparison of users’ answers to a ‘ground truth’ baseline.

Hypothesis HFI: Based on the same assumption, we also argue that within a given time, an e-Book user interface with a story linking tool will be faster than one without, to supporting a user's information seeking task. This hypothesis will be tested using the same measures of effectiveness, divided by the time taken.

Hypothesis HEC: Within a given time, an e-Book user interface with a story linking tool is more effective than one without, to supporting a user's reading for comprehension task. This hypothesis will be tested using the measures of the number of navigation actions, number of searches and number of pages viewed, as recorded by system logs; and the number of similar stories and number of themes identified from the task answer sheet. In addition, we developed a set of qualitative measures of understanding to evaluate the open-ended questions in this task: scores indicate the relevance of themes identified by the readers will be marked by three judges together to ensure consistency, and will be used to test the hypothesis. Details of the reading task and the corresponding measure are discussed later.

We conjecture that iSearch is easier to learn to use, because it has a simple query search box, which is familiar to users; iSee is easier to use once learnt, because it requires less interactions and provides more structural information of the book, meanwhile supporting more accurate location of related stories. We also conjecture that a user is likely to be satisfied with their task when the system is easy to use, enjoyable and helpful; and their satisfaction of the two systems is comparable over their ratings on satisfaction, enjoyment and helpfulness.

We proposed the following hypothesis on the usability measures of the systems:

Hypothesis HS: The users will have more satisfaction in using the story linking tool, because they find the system more enjoyable and helpful.

Following this, we propose a hypothesis relating to the users' preference between the systems.

Hypothesis HP: users will prefer the interface with story linking tool to the search-based one as measured by their answer to the preference in an exit questionnaire.

In addition, we also proposed the following hypotheses:

Research in text comprehension found that structural information like headings in text could improve the understanding of the text, for example, in (Chung 2000). We believe such information would help readers with different background knowledge in different ways. For novice readers, the structural information will help them to understand the text. For experienced readers, since they already have more or less understandings of the text, such information could help them to quickly recall what they knew, and therefore improve their speed.

Hypothesis HKN: iSee would be more helpful than iSearch for novice readers in their information seeking and comprehension, because it supported the readers with narrative structure navigation, and did not require the readers to define queries themselves.

Hypothesis HKE: iSee would also be more helpful than iSearch for experienced readers to achieve a faster speed in the information seeking tasks.

Different narrative stories might cause an effect of the system performance. For example, the story of '*Moses and the Burning Bush*' from Exodus 3: 1 – 22, in the

Old Testament section is an important event in Israel's history; it has been repeated and recalled many times in the Bible. Therefore, there are more chances to find similar stories of it from different books and section. The story '*In Athens*' from Acts 17:16 – 34, in the New Testament section happened late in the Bible timeline, and was not in the main geographical area of Israel, therefore there is less chance of finding similar stories from different books and section. The use of narrative in iSee could lead to greater use of this narrative structure in comprehension tasks. Therefore:

Conjecture CS: the story linking tool might find more similar stories from the same book and section than the search based one, because it takes the narrative structure information into account.

7.1.4 Corpus and Ground Truth

We used the organizational and narrative structural information of New International Version (NIV) of the Bible for the e-Book user interface prototype, to enable browsing and reading in the user interface.

We also used the NIV Bible as the e-Book corpus. The content of the book was displayed in the user interface with the chapter being the basic unit. With permission from the International Bible Society, we downloaded the NIV Bible from (Bible Gateway 1995) in HTML format.

As we described in chapter 5, to generate the ground truth for the story segmentation, we wrote a Perl program to parse the HTML files of the Bible, i.e., to identify the book title and the chapter number (<h4> tags), the story title (<h5> tags), and the verse number (<sup> tags) in the HTML files. In each book we also extracted the chapter number, the story title, the number of the starting verse, and the number of the ending verse for each story, and wrote this into an index file. All together 2128 stories were identified in the whole Bible. This provided an index of all the stories with their locations.

We used this index in three ways:

- First, we were able to display a list of stories in each chapter when the chapter was displayed in the iSee e-Book user interface.
- Secondly, we were able to create a collection of stories by writing each story as an individual text file.
- Thirdly, using the collection of stories we were able to search for similar stories with IR search engines, e.g., Harmonizer and Lucene.

The third step is done offline for iSee user interface – the lists of related stories are generated in advance using Harmonizer (6.2.3.1). We were able to choose a small number of top ranked similar stories to generate a narrative thread for each story, and return this thread to user when it was requested.

For the iSearch user interface, a Lucene search engine is built-in to provide query search within the collection of stories.

In addition, we acquired the Gospel Harmony table from the Blue Letter Bible website (Nevin and Alfred 2002), and used randomly selected events/topics from it to design and assess of the story searching tasks (for the Gospel Harmony task, see more details in section 7.2.4) in the user experiments. More details of the Gospel Harmony and how it was converted were covered in section 6.4.3 for the narrative structure detection.

7.2 Materials and Resources

The study required two systems (7.1.2), twenty four participants, one experimenter and three judges, and two reading task sets (each with an information seeking task and a comprehension task).

7.2.1 Participants

7.2.1.1 Participants

People who were interested in reading the Bible and who could use computers were recruited by an email invitation with suggested dates. Participants were assigned to an identifier, and compensated £10 – 20 book tokens for participation (where university staff received less than students and guests participants).

In total, we conducted 6 pilot studies to identify problems and to finalize the experimental procedure; these were followed by studies with 24 participants, among whom 12 are experienced Bible readers, and 12 were novice Bible readers. Participants were divided to experienced and novice reader of Bible according to their self-rating. The experienced readers had been reading Bible daily for at least three years. The novice readers were recruited from a wider audience, who were interested in Biblical story but rarely or never read it. Two novice participants among the twelve could not finish the experiments due to their reading speed, and thus were excluded from the study. Two further candidates were recruited to replace them in the study.

Considering that novice readers would need extra time to learn the characteristic of Bible, we gave them 10 minutes longer for the comprehension tasks. Since our experimental design was a repeated-measures one rather than between groups, such a change is probably acceptable (more detail on the design is in the section 7.3). But this might have affected the results to an extent, so we were careful to analyze the performance of novice readers and experienced readers separately for their performance on the reading task.

The career or occupation of participants includes undergraduate and postgraduate students, lecturers, medical and health care professionals, engineers, IT professionals, and a retired person. Nine participants' first language was not English, and seven belong to the novice reader's group. All participants were educated to at least undergraduate level and have lived in the UK for at least two years; therefore we believed their English is sufficient for the reading tasks.

We used the entry questionnaire (see Appendix A3.1) to find out participants' background knowledge and experience, both in using computers and in reading the Bible. A brief overview on these data will be presented below.

7.2.1.2 Computer Experience

Participants were asked to indicate their experience of reading and writing using computers (E4) and using Internet search engines (E6) using a five-point rating scale, where scale 1 is 'None', 3 is 'Some', and 5 is 'A lot'. All participants claimed to have at least some experience of computers and search engines, and seventeen participants claimed to have a lot of experience with both.

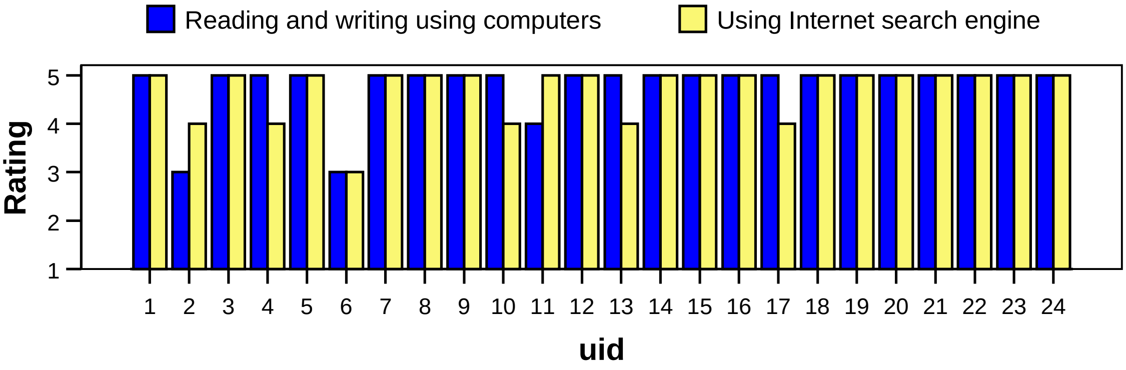


Figure 36 Participants' Computer & Search Engine Experience

Because the participants' experiences of reading and writing using computers and using Internet search engines were quite high, we suspected that there should be no significant difference between individuals on their learning to use the evaluation systems.

7.2.1.3 Knowledge of the Bible

Participants were asked about how often they read the Bible (daily, some times, and rarely or never: E1).

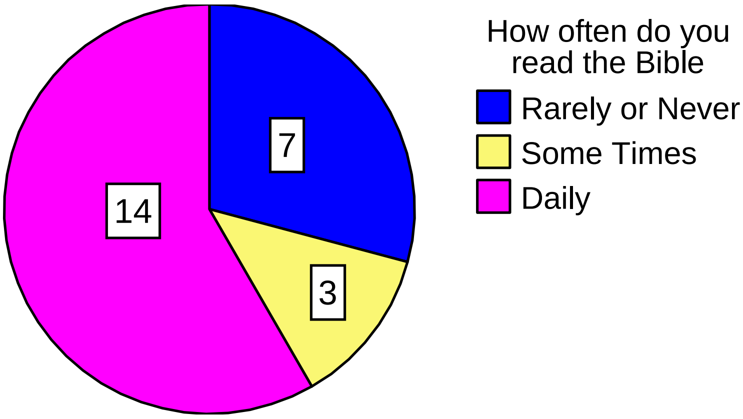


Figure 37 Participants' Frequency of Reading the Bible

14 participants indicated that they read Bible daily. Figure 37 shows the distribution of the three choices. This information is important to predict participants' background knowledge of the reading tasks, and their expected performance on the tasks. Experienced Bible readers would have a better background knowledge of Bible stories and themes.

Using a 5 point rating scale, where scale 1 is 'None', 3 is 'Some', and 5 is 'A lot', more than half of the participants indicated that they have at least some experience of using reference of the Bible (E2). 66% of participants said they have at least some experience of reading Bible Study materials (E3). Only six indicated that they have some experience in reading an e-Bible or an online Bible (E5). It is notable that among the 12 novice Bible readers, three indicated having some experience of using references of the Bible and five indicated having some experience of reading Bible Study material. Figure below shows the mean rating of participants over knowledge groups.

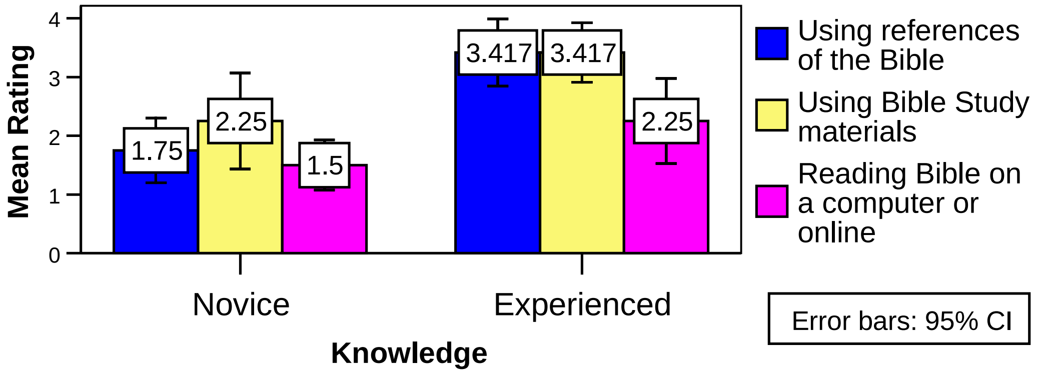


Figure 38 Bible Reading Experience (Rating 1=None, 3 = Some, 5 = A lot)

7.2.1.4 Points of Attraction while Reading

Participants were asked to describe how often they were attracted by information while reading using a 5 point rating scale, where 1 is 'Never', 3 is 'Sometimes', and

5 is ‘Often’. This shows their interestedness or motivation in reading. The following table shows the mean and mode of these rating, and percentage of participants who at least sometimes focused on the interesting points of information.

Table 12 Analysis of Points of Focusing

When you read the Bible or a Book, how often do you focus on:	Knowledge	Mean	Mode	Percentage of Ratings>=3
E7A. Word(s)	Experienced	3.08	3	83.3%
	Novice	2.5	3	50%
E7B. Phrase	Experienced	3.33	3	91.7%
	Novice	2.92	3	66.7%
E7C. Verse (Sentence)	Experienced	3.83	3	100%
	Novice	2.75	3	58.3%
E7D. Character (e.g., Jacob)	Experienced	3.92	4	91.7%
	Novice	2.67	2	41.7%
E7E. Passage/Topic/Story	Experienced	4	5	91.7%
	Novice	3.58	5	75%
E7F. Theme	Experienced	3.58	4	83.3%
	Novice	2.92	2	58.3%
E7G. Reference	Experienced	2.75	2	50%
	Novice	2.25	2	25%

The table showed that both experienced and novice readers are often attracted often by the topic or story in a book, while experienced readers are also attracted by characters and themes of a book. Information like words, phrases, and verses sometimes attract readers’ attention. References are, for some reason, not attractive.

The discovery of readers’ interests in stories is interesting and encouraging. This confirmed our motivation, described in the scenario (1.4.2) and system design (4.1), that readers need user interface support to explore the stories in a complex book. The current references in books, including back of book indexes, footnotes, cross-references, etc., are not as helpful as they should be.

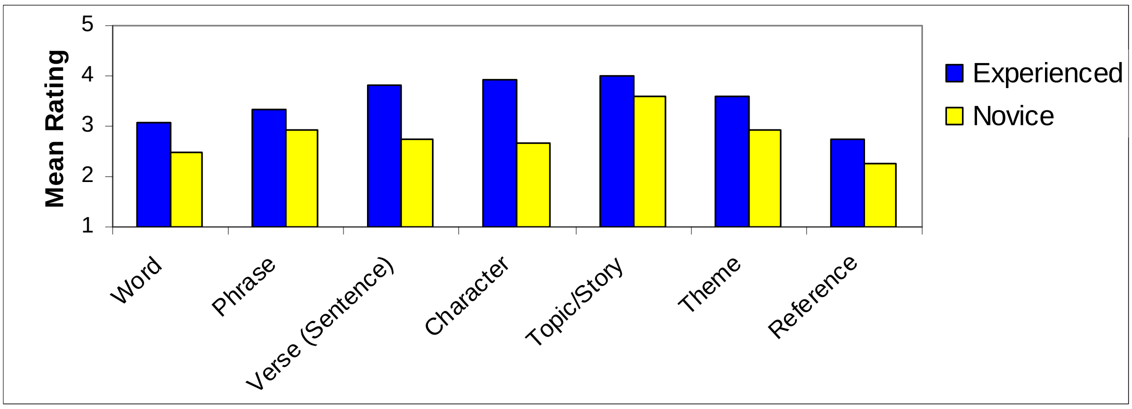
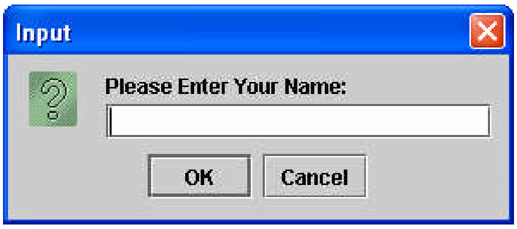
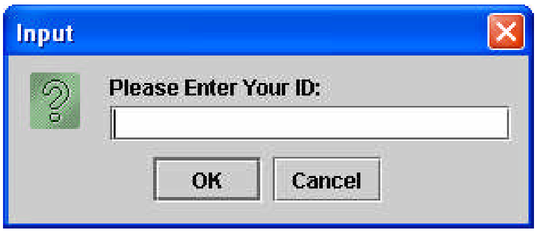


Figure 39 Average Rating for Points of Interests in Reading

7.2.2 Control Panel and System Logs

The participants were required to use the e-Book system on a PC and complete their reading tasks and questionnaires with a pen on printed paper, making it hard to monitor their progress on the tasks. To support this, we designed a control panel to assist recording of the system logging information. Upon starting a system, each participant was prompted to enter their name and identifier to start a control panel.





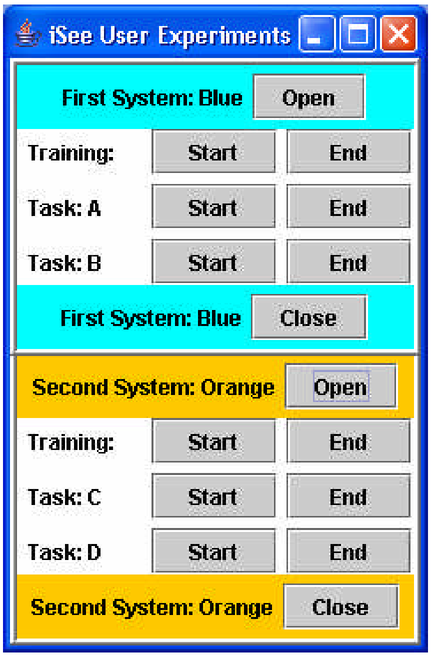


Figure 40 Experiments control panel

The order of the systems and the order of tasks in the control panel were determined by their identifier. Once launched, the participant would use the buttons on the control panel to open each system user interface, and to control and record the starting and ending time for each task. Their interactions on the user interface and the control panel were recorded in system logs.

Since the systems were written in Java, we used the built-in Logging APIs in Java to create log files and record information from user's interactions. The participant's identifier, current system, current task, the time and the number of navigation actions for each task, the number of pages viewed, the number of queries (when necessary) searched and the length of each query, etc., were recorded. These logging files were uploaded to Excel and SPSS for further data analysis.

7.2.3 Experimenter and Judges

The experimenter was present during each experiment to train participants with the systems and tasks, and to assist them with the experimental procedure. Two independent judges and the experimenter rated the responses on the task sheet, and made final assessment of the answers. All the judges had good knowledge of Biblical stories and were familiar with the themes in the Bible.

7.2.4 Reading Tasks and Questionnaire

7.2.4.1 Overview

Two kinds of reading tasks, information seeking tasks and reading for comprehension tasks, were designed for the experiments. Each participant was assigned two tasks of each type, inspired by the structure of Bloom and Anderson’s Taxonomy (2.3.2). The four tasks were arranged in two task sets for the study, with an information seeking task (tasks A and C, Appendix A3.4 and A3.12) leading a reading for comprehension task (tasks B and D, Appendix A3.7 and A3.14), in order to help the reader to get to know the system through progression from easier to harder tasks.

Task set 1 contained task A and B, task set 2 contained task C and D. Due to time and number of participants available, the other possible combinations of tasks were ignored in the study. In the final results we analysed each type of task independently.

These tasks were merged with a set of questionnaires to gather participants’ opinion and responses before and after their experiments and between each task and system. Details of these questionnaires can be found at Appendix 3.1, A3.5, A3.9 and A3.16.

We designed a real world simulated information seeking task, the Gospel Harmony task, to test the systems' effectiveness in assisting readers to rapidly browse and search for passages. The second type of task, a semi-directed reading task (was presented to participants as 'Link and Think task'. We also call it 'Story-Theme Map' task in this thesis) was used to test the systems' effectiveness in assisting readers in understanding and identifying related stories and their wider themes.

7.2.4.2 Information Seeking Task: Gospel Harmony

A Gospel Harmony is a list of events that combines the four Gospels into a single account of *Jesus Christ's life on earth*. It sorts all the events in the Gospels in a roughly chronological order, and presents the Gospel narratives of the same event (or topic) side by side. For example, the event of *Baptism of Jesus Christ* appeared in Matthew 3:1 – 12, Mark 1:9 – 11, Luke 3:21 – 22, and John 1:32 – 34. Readers can easily compare the passages and thus view each Gospel writer's own perspective on the event or topic.

Table 13 Example of Gospel Harmony Tasks

Title	Matthew	Mark	Luke	John
E.g., Baptism of Jesus Christ	3: 1 – 12	1: 9 – 11	3:21 – 22	1:32 – 34
Q1.				6:1 – 15

The Gospel Harmony tasks (task A and C, which is also the first tasks for each system, were created based on the Gospel Harmony. Each task contained three randomly selected events from the harmony, with one passage revealed for each event. Following an example, participants were given 7 minutes to find all the passages in the four Gospels that described the same event, i.e., *Jesus Feeds the Five Thousands*.

Table 14 Example of Suggested Correct Answers in the Gospel Harmony

Title	Matthew	Mark	Luke	John
E.g., Baptism of Jesus Christ	3: 1 – 12	1: 9 – 11	3:21 – 22	1:32 – 34
Q1. Jesus Feeds the Five Thousand	14:13 – 21	6:30 – 44	9:10 – 17	6:1 – 15

Because there is a ground truth, the Gospel Harmony, behind each event selected, participants' performance of this task was measured by the effectiveness and efficiency of their answers, i.e., the time taken to complete the task, the proportion of responses completed, the number of correct responses, precision and speed.

7.2.4.3 Reading for Comprehension Tasks: Story-Theme Map

The Link and Think tasks (task B and D, which is the second tasks for each system) were designed based on the Concept Mapping and Directed Reading and Thinking concepts (2.2.3 and 2.2.4, more on this will be discussed in chapter 8). The main part of the task is to illustrate a Story-Theme Map following an example (see 2.4 and Appendix A3.6). Given a story, participants were asked to understand the core message, find related stories in the Old and New Testament, and generalize themes. Participants were asked to present their findings in a Story-Theme Map, which would be explained in the next section.

The experienced participants were given 28 minutes and the novice readers were given 38 minutes to read and browse the passages, and complete the map. There was no right or wrong answers; and participants were encouraged to explore the text through various routes. Their performance was measured by the effectiveness and relevance of their answers. Three judges independently judged the relevance of a theme identified by participants, with the thread of stories associated with the theme, and assign the theme a ranked score from 0 to 3. We will discuss the data coding in 7.4.2. A coding sheet for the judges is included in Appendix A4.

7.3 Experimental Design

7.3.1 Experimental Design

As we have described, a within-subject, repeated measures laboratory experiment was conducted to compare two different user interfaces for an e-Bible. The system and tasks were rotated to ensure that there were no task-specific effects in the experiments. This was achieved as follows.

All participants completed two reading task sets, one with iSee (Blue) system, and the other with iSearch (Orange) system. Participants were randomly assigned into four groups, two groups started with the iSee system first, and the other two started with the iSearch system first. Table 15 shows the task sets and system (iSee/iSearch) combinations (where task set 1 refers to task A and B, task set 2 refers to task C and D, see Appendix A3.4, A3.7, A3.12, and A3.14).

Table 15 System-Task Groups

System-Task Group	System & Task Order	
	First System/Tasks	Second System/Tasks
1	iSee/Task set 1	iSearch/Task set 2
2	iSee/Task set 2	iSearch/Task set 1
3	iSearch/Task set 1	iSee/Task set 2
4	iSearch/Task set 2	iSee/Task set 1

General Linear Model and a number of mixed ANOVA tests were used to investigate the difference between systems' performance within each participant and between two knowledge groups.

Details of data coding and measures are reported in 7.4, and the data analysis methods in 7.5.1.

7.3.2 Procedure

The time allocated for the experiments was 130 minutes (for experienced readers) and 150 minutes (for inexperienced readers), in both cases including a 15 minute break.

Table 16 Experiment Protocol

(The order of tasks may vary, see Table 15)

1. Entry Questionnaire & Oral Briefing	10 minutes	
2. First System Training and Training Task	10 minutes	40/50 minutes
3. Gospel Harmony Task & Post-Task Questionnaire	7 minutes	
Read the Link and Think Example	5 minutes	
4. Link and Think Task & Post-Task Questionnaire	28/38 minutes	
5. Post-System Questionnaire	5 minutes	
Break	15 minutes	
6. Second System Training and Training Task	5 minutes	35/45 minutes
7. Gospel Harmony Task & Post-Task Questionnaire	7 minutes	
8. Link and Think Task & Post-Task Questionnaire	28/38 minutes	
9. Post-System Questionnaire	5 minutes	
10. Exit Questionnaire	5 minutes	
Total	130/150 minutes	

The overall procedure was:

- Upon arrival, participants were met and briefed about the evaluation, and were asked to fill in a short entry questionnaire. They were given an identifier with the entry questionnaire (a set of questionnaire and task sheet are included in Appendix A3). The identifier number determined which task/system group they would be in. They were permitted to ask any questions during the experiments, and to finish their tasks earlier if desired.

- Once ready, participants were presented with the first experimental user interface and a user manual, and were given a training task. They were to teach themselves using the user manual and complete the training task. The instructor checked their answer for the training task to ensure that they had learnt how to use the system.
- Participants were then given a work sheet with tasks and a post-task questionnaire for each task. They had 7 minutes to complete the Gospel Harmony task, 5 minutes to learn the Story-Theme Map concept, and 28 or 38 minutes to complete the Link and Think task, depending on their level of background knowledge of the Bible. After finishing all the tasks for the first system, participants completed a post-system questionnaire.
- After a 15 minute break, participants performed a training task for the second system, and completed a Gospel Harmony task followed by a Link and Think task. Again, after completing the tasks, participants completed a post-system questionnaire.
- Finally, participants rated the two systems for general preference by completing an exit questionnaire after they completed the tasks using both systems.

7.4 Data Coding and Measures

7.4.1 Data Coding: Overview

We evaluated the usability of the two interfaces by using measures of effectiveness, satisfaction, and efficiency, as recommended in (Frøkjær, Hertzum et al. 2000). The effectiveness of the interaction with the two interfaces was measured through the score for the answers to the tasks, depending on the accuracy, relevance and completeness. Satisfaction was measured by the post-system questionnaire, exit questionnaire and comments. The main efficiency measure, the time used, was derived from the system logs.

Four forms of data were collected during the evaluation:

- System log data: number of navigation actions, number of searches, number of pages viewed, and time (minute) spent on each task
- Questionnaire responses: questions include 5-point rating scale questions, multiple choice questions and comments. Numeric data were then entered into SPSS and analyzed; comments were typed into Excel spreadsheet for further qualitative analysis. See Appendix A3 for the set of questionnaires.
- Correctness and completeness ratings for the close-ended questions in the Gospel Harmony tasks, and
- Relevance and completeness ratings for the open-ended questions in the Link and Think tasks.

The analysis of quantitative data from the closed questions, the Gospel Harmony task, was fairly straightforward. The answers of participants were compared with the original Gospel Harmony table and for each passage (including the story titles, book, chapter and verse), a code is given to indicate a right answer (code = 1), a wrong answer (code = –1), and an omit answer (code = 0). This allowed further measurements to be calculated, i.e., proportion of completion (ignoring right or wrong answers), number of correct answers, and precision, etc.

The analysis of answers from open-ended questions, the Link and Think task, was significantly more difficult. A lot of effort was put into converting the qualitative data to quantitative data by a coding the responses. This is worth discussing in some detail.

Due to the different nature of the Gospel Harmony tasks and the Think and Link tasks, the records for these tasks were coded into separate data sheets and analyzed separately.

7.4.2 Data Coding: Comprehension Tasks

During the task, participants were asked to read a story, search for and explore related stories, identify common themes of them, and draw a Story-Theme map (7.2.4.3 and 2.4) that linked the themes with a thread of related stories, where the theme is a unifying subject or idea behind these stories. The way a reader identifies themes of a story shows how she understood and interpreted the story. In order to evaluate the readers' responses effectively, we developed a two stage coding scheme: firstly counting, and second scoring the maps.

At the counting stage, we counted the number of themes identified in a Story-Theme Map, the number of stories linked to each theme, the number of stories from the same book as the original story, the number of stories from different books, the number of stories from the same section (Old or New Testament), and the number of stories from different sections. We believed that a bigger span of stories viewed by a reader would help her to get a better overview of the stories, and thus make it easier to find a theme.

We then used a content analysis of the Story-Theme map drawn by the participants, to code participants' responses and allocate scores that could be analyzed. Three judges include the experimenter graded the themes separately and independently. Before coding, the judges met to discuss the process and to agree a coding scheme.

We developed a coding scheme to transfer the qualitative data to quantitative measures. For example, giving a limited time, how many themes has a reader identified? How many stories has she read? How many stories she read were from the same book? And how many stories are from different books? For each of the themes identified: is the theme relevant to all the stories linked to it? Is the theme accurately representing all the stories linked to it? A relevance score from 0 to 3 is assign to a theme, where 0 is 'irrelevant', 1 is 'weak relevant', 2 is 'mostly relevant' and 3 is 'highly relevant'. An example of the coding scheme that we gave to each judge is included in Appendix A4.

After coding the Story-Theme maps separately, judges met to compare results and discuss difference. In general, there was very low agreement about the relevance of themes between three judges: Judge1 and Judge2 has 58% agreement (Kappa =

0.316), Judge1 and Judge3 has 46% agreement (Kappa = 0.358), and Judge2 and Judge3 has 67% agreement (Kappa = 0.40). Because of the low agreement, judges engaged in a consensus method to resolve disagreements and produce a final listing of scores for the themes identified.

7.4.3 Independent Variables and Measures

The main independent variables of this study are the system (iSee and iSearch) and participant's background knowledge of Bible (experienced and novice).

The experimental design also includes the following factors:

- Order (first system/task group and second system/task group).
- System task group (see Table 15).
- Tasks (A, B, C and D)

We collected information from system logs, questionnaire and task sheets. They were used as dependent variables for both reading tasks respectively in the data analysis. The main dependent variables were:

- The time each participant spent on each task.
- The number of navigation actions performed in a task.
- The number of searches performed in a task.
- The number of pages viewed when performing a task.
- The proportion of completed questions.
- The correct answers for the close-end questions.

- The relevance score of the open-ended questions.
- Ratings in the post-task, post-system and exit questionnaires.

We also collected and used a few other variables during the data analysis, as we felt it was worth exploring the effect of these additional variables. They will be reported in later discussion.

7.5 Results

7.5.1 Data Analysis Methods

The analysis of the user evaluation data is a big challenge.

- The two kinds of tasks, information seeking and comprehension task, should be analysed separately.
- The performance of the two reader groups should first be analysed separately, and then compared together.
- The quantitative data from system logs, task sheet and questionnaires should be analysed differently.
- The effect of Order and Task should be considered.

When comparing means of a dependent variable which is used to measure the system performance, we employed a General Linear Model (GLM) univariate analyses, using system, order and task as fixed factors, users as random factor. The dependent variables are the measures we used for the system performance, e.g., number of searches, minutes, proportion of completed task, etc. This method is more powerful and accurate than the GLM repeated measures analysis because, using this method, the difference between two systems will be emphasized, with the effect of order or task estimated.

Other methods are also adapted for different type of data, for example, using the Wilcoxon tests to compare the ratings in questionnaire data. These will be specified when used.

Below we present most interesting and important results from the evaluation.

7.5.2 Hypothesis HEI – IR Task Effectiveness

The simulated information seeking tasks on the Gospel Harmony (Task A and C, Appendix A3.4 and A3.12) were designed to test the systems' effectiveness in assisting readers in rapidly browsing and searching for passages.

7.5.2.1 Overview: Time and Interactions

The task was limited to 7 minutes, and an average of 5.64 minutes was obtained, with a minimum of 3.1 minutes and a maximum of 12.5 minutes. The extreme 12.5 minutes was due to an operational failure (the participant forgot to press the End button in the control panel when they finished the task, and the experimenter did not notice it when this had happened). Note the time recorded by system logs for each task included both reading time and time spent on answering the questions. As we used paper and pen mode questionnaire and task sheet (Kelly, Harper et al. 2007), we could not record precisely when did readers switched from reading on the screen to their task sheet to write down their findings.

On this task, 31.3% of tasks were completed within 5 minutes (n=15); of which 66.7% were done with iSee and 33.3% were done with iSearch.

64.6% of tasks were fully completed (n=31); of which 58.1% were completed using iSee and 41.9% were completed using iSearch.

31.25% of responses achieved 100% in precision – i.e., were wholly correct (n=15); of which 60% were done with iSee and 40% were done with iSearch.

The following table shows the comparison of the two systems by a number of General Linear Model univariate tests. The 'Sig.' column presents the significant p-value, this is accompanied with the f-value and the significant p-value for order effect.

Table 17 Factorial Analysis of Minute, Navigation, Searches and Pages

Measure	Knowledge	Mean (Std. Deviation)		Order Effect	F-Value	Sig.
		iSee	iSearch			
Minutes	Novice	5.75(1.50)	6.34(2.32)	.039	1.024	.338
	Experienced	4.89(.90)	5.56(.79)	.425	7.329	.024*
Navigation Actions	Novice	30.5(10.9)	25.92(9.64)	.088	3.186	.108
	Experienced	30.58(11.13)	23.08(7.09)	.663	8.387	.018*
Searches	Novice	4.17(1.69)	3.83(1.64)	.611	.277	.611
	Experienced	4.75(1.81)	3.75(1.28)	.083	2.382	.157
Pages	Novice	13.25(6.03)	12(5.9)	.046	.430	.528
	Experienced	11.92(6.45)	9.25(3.79)	.394	3.209	.207

Note the searching in iSee system is not explicit: we counted the number of times a reader clicked on a story title to get the list of related stories as the number of searches. In iSearch, the number of searches is simply the number of times a reader clicked on the Search button to submit a query.

The above table shows that for novice readers, the two systems were almost equivalent in terms of time (minutes), number of navigation actions, number of searches, and number of pages viewed. However, the experienced readers spent significant less time in iSee than iSearch for their task and performed more actions using iSee than iSearch. Further ANOVA tests showed that the difference between experienced readers and novice readers was not significant using either system.

7.5.2.2 Task Effectiveness

We counted the number of responses by each participant and calculated the proportion of completion, assessed the corrected answers by a comparison with the original Gospel Harmony, and computed precision (Equation 13, page 120). Precision was the number of correct answers as a proportion of the number of responses; and completion was the proportion of questions completed out of total number of questions. See Appendix A3.4 and A3.12 for Task A and C. Each blank cell in the table of Gospel Harmony task is treated as a question.

Table 18 Factorial Analysis of Completion, Correct Answers and Precision

Measure	Knowledge	Mean (Std. Deviation)		Order Effect	F-Value	Sig.
		iSee	iSearch			
Completion	Novice	.88(.18)	.89(.15)	.578	.083	.779
	Experienced	.98(.03)	.90(.13)	.057	4.765	.057
Correct Answers	Novice	9.67(2.22)	9.58(2.31)	.126	.035	.856
	Experienced	10.17(1.46)	9.83(2.08)	.172	.353	.567
Precision	Novice	.91(.11)	.88(.11)	.507	.446	.521
	Experienced	.85(.11)	.89(.08)	.907	2.373	.158

There was no significant difference between the two systems. iSee was at least as good as iSearch in terms of the proportion of completion, correct answers and precision. Experienced readers’ completion showed a trend towards being higher with iSee compared to iSearch (at p<.06 level).

Interestingly, the novice readers achieved better precision than experienced readers on average (.91 vs. .85). However, a follow up ANOVA test shows that the difference was not significant (F=1.639, p>.21).

We also tested the difference between novice readers and experienced readers' when they used each system. The results are not significant, showing that novice readers did as good as experienced readers using both systems, during these information seeking tasks.

There were also two tasks completed, one with each system. One way ANOVA tests showed that participants' performance on task A is significant better than their performance on task C (p<.05) in terms of minute, number corrects, precision, completion and correctness. This result indicated that task A might be easier than task C.

To summarize, we were unable to reject the hypothesis HEI. In information seeking task, iSee with a story linking tool is as effective as iSearch with a query search tool. This is measured widely by the number of searches, the number of page viewed, the proportion of completion, the correct answers and precision. However, for the experienced readers, the iSee story linking tool did show evidence of greater performance than iSearch, according the time spent and the navigation actions performed in their tasks.

Using these effectiveness measures and above analysis, we could not find evidence for our hypothesis HKN, that iSee was more effective to help novice readers in their information seeking task, than iSearch.

However, are these measures really 'effective' in measuring the systems' performance? We will discuss on this in 8.5.2.

7.5.3 Hypothesis HFI – IR Task Efficiency

We investigated efficiency by computing five new variables: number of navigation actions per minute, the number of searches per minute, the number of pages viewed per minute, and the number of correct answers per minute:

Table 19 Factorial Analysis of Navigation, Searches, Pages, Completion and Correct Answers Per Minute

Per-Minute	Knowledge	Mean (Std. Deviation)		Order Effect	F-Value	Sig.
		iSee	iSearch			
Navigation Actions	Novice	5.44(1.77)	4.24(1.13)	.706	5.678	.041*
	Experienced	6.4(2.56)	4.17(1.28)	.881	12.434	.006*
Searches	Novice	.77(.39)	.65(.28)	.097	1.151	.311
	Experienced	.99(.37)	.66(.16)	.581	4.884	.054
Pages	Novice	2.29(.8)	1.9(.66)	.316	1.653	.231
	Experienced	2.53(1.6)	1.68(.73)	.633	5.072	.051
Completion	Novice	.16(.07)	.15(.06)	.022	.787	.398
	Experienced	.20(.03)	.16(.03)	.074	11.164	.009*
Correct Answers	Novice	1.87(.91)	1.71(.79)	.032	.637	.445
	Experienced	2.13(.48)	1.8(.49)	.105	5.486	.044*

For experienced readers, iSee was significant faster than iSearch in the number of actions, completion and correct answers per minute. On average, iSee was faster than iSearch in the number of searches and pages per minute, the significant p-values were marginal.

Novice readers use iSee also achieved significant navigation actions per minute using iSee than iSearch.

Further ANOVA tests showed that the difference between experienced readers and novice readers was not significant using either system, according to these five measures.

In no occasion iSee was less efficient than iSearch. Therefore, we were able to reject null Hypothesis HFI, in that within a given time, the iSee user interface was more efficient than iSearch in supporting users' information retrieval tasks.

This also provided evidence for our hypothesis HKE, that iSee helped experienced readers to achieve a faster speed in their information seeking tasks.

7.5.4 Hypothesis HEC – Comprehension

We designed a semi-directed reading task, the Link and Think task (Appendix A3.7 and A3.14 for Task B and D) to test the systems' effectiveness in assisting readers in understanding and identifying the similar stories and their corresponding themes. These tasks required the participants to read a story, identify related stories, identify common themes, and complete a Story-Theme map that linked the themes with related stories. In the task, a Story-Theme Map consists of an original story as a starting point, maximum of three themes, and ten related stories.

Degrees of understanding were evaluated using the number of related stories and themes identified by participants, the location of identified stories, and the relevance scores of the themes (7.4.2).

7.5.4.1 Overview: Time and Actions

Participants were given 28 minutes (if they were experienced Bible readers) or 38 minutes (if they were novice Bible readers) for the task. Experienced readers took a mean 20.17 minutes, with a minimum of 12.6 minutes and a maximum of 23.6 minutes. Novice readers took a mean of 30.7 minutes, with a minimum of 20 minutes and a maximum of 40.1 minutes.

The following table describes the quantitative data from the system log and task log, and the results from the GLM univariate tests:

Table 20 Factorial Analysis of Minutes, Searches, and Pages

Measure	Knowledge	Mean (Std. Deviation)		Order Effect	F-Value	Sig.
		iSee	iSearch			
Minute	Novice	28.83(4.76)	32.56(5.26)	.002	9.156	.014*
	Experienced	20.09(3.24)	20.25(2.70)	.936	.027	.872
Searches	Novice	11.92(15.77)	7.92(4.54)	.38	.786	.398
	Experienced	6.33(3.70)	5.17(1.69)	.869	1.409	.266
Pages	Novice	31.42(18.58)	25.67(16.46)	.053	3.123	.111
	Experienced	18.67(8.25)	17.58(3.605)	.613	.160	.698

For the experienced readers, there was no significant difference between the two systems in all the three measures.

For the novice readers, there was no significant difference between the two systems in terms of the number of pages viewed, and the number of searches performed during the Link and Think task; but iSee was significantly shorter than iSearch in terms of the time taken (in minutes).

We did not compare the difference between the novice readers and experienced readers on these system logs measures, since they were given different length of time for their tasks in the experiments. However, we will compare their performance on the qualitative aspect of their tasks later.

7.5.4.2 Number of Stories and Themes

We counted the number of stories and the number of themes each participant identified in each Story-Theme Map. A total 113 themes (including repeated or similar themes, e.g., ‘God’s power’ and ‘God is powerful’) were identified by participants in the experiments. 58 themes were identified using iSee, and 55 themes using iSearch.

Table 21 Frequency of Number of Themes Identified in Each Task

Knowledge	Number themes	iSee	iSearch	In number tasks
Novice	0	1	1	2
	1	1	4	5
	2	4	0	4
	3	6	7	13
	Total	27	25	24
Experienced	0	0	0	0
	1	1	2	3
	2	3	2	5
	3	8	8	16
	Total	31	30	24

Table 22 shows the results of GLM univariate tests on the number of stories and the number of themes:

Table 22 Factorial Analysis of Number of Stories and Number of Themes

Dependant Variable	Knowledge	Mean (Std. Deviation)		Order Effect	F-Value	Sig.
		iSee	iSearch			
Number Stories	Novice	7.67(2.06)	7.58(2.53)	.623	.029	.869
	Experienced	8.67(1.61)	7.92(2.71)	.132	.989	.346
Number Themes	Novice	2.25(.96)	2.08(1.16)	1.000	.474	.506
	Experienced	2.58(.66)	2.50(.79)	.061	.184	.678

There was no significant difference between the two systems in terms of number of stories retrieved and the number of themes identified. However, when used as the second system, participants identified significantly more stories with iSee than iSearch (F=5.91, p<.05). This perhaps can be explained that in the first half of the experiment, participants have to learn the tasks and the system; once they learnt the tasks, iSee is more helpful for their tasks.

On average, experienced readers found more related stories and identified more themes than the novice readers, but further ANOVA tests showed the differences were not statistically significant, for both systems. Note that novice readers had 10 minutes longer than the experienced readers in these Link and Think tasks.

Summary: these results show that iSee was at least as good as iSearch in assisting readers' comprehension, as measured by the number of related stories and the number of themes identified in a given time.

7.5.4.3 Relevance Score of Themes

Two Link and Think type of tasks: B and D, were used with each of the two systems. The results of these tasks were evaluated by three judges to give a relevance score for each theme identified. One way ANOVA test showed that participants' performance on task B and D has no significant difference in terms of time, the number searches, the number of pages, the number of stories, the number themes, etc. These results showed that task B and D were comparable in the degree of difficulty.

Table 23 shows that the number of top-scoring themes identified by novice readers using iSee was double that of iSearch (15 compared to 7):

Table 23 Frequency of Score of Themes

Score of Theme	Novice		Experienced	
	iSee	iSearch	iSee	iSearch
0		1		
1		2		1
2	12	15	7	8
3	15	7	24	21
Total number themes	27	25	31	30
Total score of themes	69	53	86	80

Table 24 shows that the themes identified by novice readers using iSee have significant higher scores than iSearch (F=9.54, p<.01). Experienced readers' score of themes were not affected by which system they used.

Table 24 GLM Univariate Factorial Analysis on the Score of Themes

Knowledge	iSee	iSearch	Order	Task	F-value	Sig.
Novice	2.56(.51)	2.12(.73)	.129	.654	9.54	.004*
Experienced	2.77(.43)	2.67(.55)	.951	.787	.458	.502
All	2.67(.47)	2.42(.69)	.281	.936	8.357	.005*

When included all the participants in one test (the bottom row in the above table), we found that the themes identified using iSee have significant higher scores than using iSearch (F=8.357, p<.01). The task and order effects were not significant.

The above table also showed that score differences were greater between the novice readers and the experienced readers when they used iSearch. A further ANOVA test confirmed that novice readers' average score of themes were as good as those of experienced readers when they used iSee; and they were significantly lower when they used iSearch (F=10.137, p<.05).

7.5.4.4 Distribution of Stories

We examined the stories identified by each participant to find the distribution of the related stories they found in the Bible. The table below presents the results from GLM univariate tests on the distribution of number of stories.

Table 25 Factorial Analysis of the Distribution of the Number of Stories

Measure	Knowledge	Mean (Std. Deviation)		Task Effect	F-Value	Sig.
		iSee	iSearch			
Same Book	Novice	5.08(2.42)	3.58(2.39)	.029*	2.782	.130
	Experienced	5.08(3.11)	3.08(2.27)	.010*	10.623	.010*
Different Books	Novice	2.58(2.27)	4.0(2.92)	.002*	2.979	.118
	Experienced	3.75(3.49)	4.92(2.42)	.002*	2.722	.133
Same Section	Novice	6.83(1.99)	6.08(2.50)	.257	1.467	.257
	Experienced	7.42(1.78)	5.42(2.81)	.121	6.612	.030*
Different Section	Novice	1.0(.85)	1.5(1.38)	.756	.920	.362
	Experienced	1.42(1.56)	2.5(1.56)	.907	2.434	.153

For experienced readers, iSee helped to identify significant more stories from the same book (p<.05) and same section (p<.05), than iSearch. There was no significant difference, however, on the number of stories in the different book and section.

For novice readers, the two systems were almost identical according to the distribution of stories.

Further ANOVA tests showed that the difference between experienced readers and novice readers was not statistically significant for either system, according to these measures.

Note there were significant task effects on the number of stories found in the same book and different books, see the fifth column above for p values.

Figure 41 and Figure 42 shows bar charts of the distribution of stories in each task for each system on average. As predicted, for story ‘*Moses and the Burning Bush*’, both systems identified more stories from different books than from the same book (Exodus), and for story ‘*In Athens*’, both systems identified more stories in the same book (Acts) than different books.

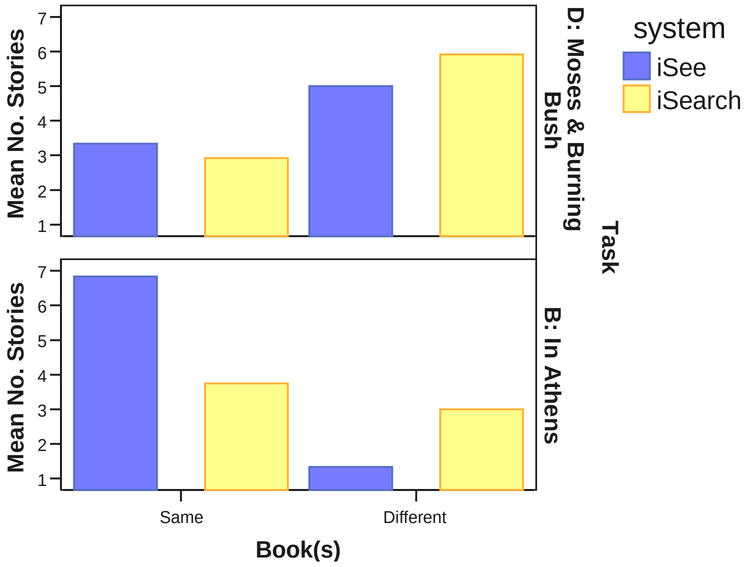


Figure 41 Bar chart of number of stories in same or different books

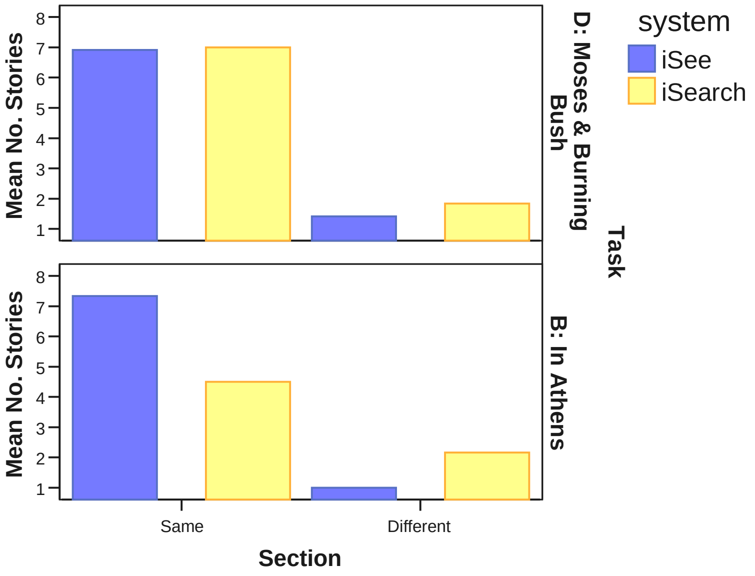


Figure 42 Bar chart of number of stories in same or different section

Figure 42 also shows, both systems identified more stories in the same section than different section.

Despite of the task effect, data analysis showed that in iSee, related stories from different books tended to be ranked lower than those from the same book or section. This might discourage a reader to look into those stories, yet it does reflect the use of narrative structure by iSee. To improve the iSee techniques further, we could provide additional features, for example, allowing the search results to be filtered showing similar stories from a certain book, section, or different books. We have found evidence to support our conjecture CS, the use of narrative in iSee led to greater use of the narrative structure in responses to the comprehension task.

7.5.4.5 Summary

Using these results, we found evidence to reject the null hypothesis HEC. Participants using iSee did have better understanding of the stories and themes than those using iSearch, when compared in time, the number of searches done and the number of pages viewed being almost equal. iSee is also as good as iSearch in assisting readers' comprehension, as measured by the number of related stories and number of themes found in a given time.

Experienced readers using iSee found significant more stories from the same book and the same section than iSearch. Meanwhile, iSee was as good as iSearch by providing similar stories from different books and difference section, since the difference between the two systems on these measures was not significant.

The number of stories and number of themes identified using iSee were not significantly higher than iSearch. However, the overall quality of themes found using iSee was better than those found using iSeasrch. For novice readers, themes found using iSee was significantly better than those found using iSearch, as measured by the score of themes. Also, the total score of participants' themes found using iSee was significant higher than those found using iSearch.

Most importantly, novice readers' score of themes were as good as experienced readers' ones when they used iSee; and they were significantly worse when they used iSearch. This provide good evidence that iSee is helpful to novice readers trying to understand and make thematic links. This is a good evidence to support our hypothesis HKN. We will discuss this point further in 8.5.4.

7.5.5 Hypothesis HS: Satisfaction

7.5.5.1 Usability Measures

The post-task and post-system questionnaire consisted of a number of 5-point rating scale type of questions, where 1 meant 'Not at all', 3 meant 'Somewhat', and 5 meant 'Extremely' as anchors. The mean, standard deviation (in brackets), and the significant p-value of the Wilcoxon (exact test with two tails) tests are reported in the following tables. We will discuss these results in next section.

Table 26 Post-Task Satisfaction of the Gospel Harmony Task

Question	Knowledge	iSee	iSearch	Sig.
PT1. Was it easy to do this portion of the Gospel Harmony?	Novice	3.92(.90)	4.25(.75)	.359
	Experienced	4.25(.75)	3.83(.93)	.375
PT2. Are you satisfied with your performance?	Novice	3.42(1.31)	3.67(1.63)	.677
	Experienced	4.00(.60)	3.83(1.33)	.836
PT3. Did your previous knowledge help you with your answering?	Novice	2.25(1.21)	2.50(1.56)	.563
	Experienced	3.25(1.35)	3.08(.90)	.730

Table 27 Post-Task Satisfaction of the Story-Theme Map Task

Question	Knowledge	iSee	iSearch	Sig.
PT1. Was it easy to do this reading task?	Novice	3.08(1.16)	3.17(.937)	.617
	Experienced	3.00(.85)	3.42(.90)	.344
PT2. Are you satisfied with your performance?	Novice	3.00(.95)	3.00(1.20)	1.00
	Experienced	3.08(.90)	3.08(.99)	1.00
PT3. Did your previous knowledge help you with your answering?	Novice	1.83(1.03)	2.33(1.43)	.156
	Experienced	3.67(.98)	3.75(1.21)	.796

Table 28 Post-System Satisfaction

Question	Knowledge	iSee	iSearch	Sig.
PS1. How easy was it to learn to use this user interface?	Novice	4.08(.51)	4.42(.66)	.289
	Experienced	4.08(.66)	4.33(.49)	.500
PS2. How easy was it to use this user interface to find stories in the Bible?	Novice	4.42(.99)	4.25(1.05)	.531
	Experienced	4.17(.71)	4.25(.75)	1.00
PS3. How well did you understand how to use the user interface?	Novice	4.08(1.16)	4.50(.90)	.063
	Experienced	3.92(.51)	4.25(.62)	.125
PS4. Did you enjoy using the system?	Novice	3.83(1.33)	4.00(.95)	.750
	Experienced	3.92(.66)	4.08(.90)	.750
PS5. Did the system help you to complete the tasks?	Novice	3.83(1.26)	3.83(.83)	1.00
	Experienced	3.92(.90)	4.17(.83)	.625
PS6. Did the system help you to understand the stories?	Novice	3.08(1.31)	2.92(.90)	.750
	Experienced	3.58(.99)	3.67(.49)	1.00

Upon completion the experiments, in the exit questionnaire, over half the participants (n=14) said that it was easier searching for related stories with iSee than iSearch (Ex7), because there was a list of stories in the chapter. On the other hand, less than half the participants (n=10) responded that it was easier searching for related stories with iSearch (Ex8), because it provided query searching.

Table 29 shows the participants' opinions of the two systems on their ease of use:

Table 29 Counts of Participants' Opinions on Ease of Use

Question	Knowledge	iSee	No Difference	iSearch
Easier to learn (Ex9)	Novice	3	4	5
	Experienced	4	7	1
	All	7	11	6
Easier to use (Ex10)	Novice	5	3	4
	Experienced	4	3	5
	All	9	6	9

7.5.5.2 Satisfaction

Overall, the difference between the two systems was not significant according to the mean ratings of the post-task, post-system, and exit questionnaires. The statistical results were presented in Table 26 – Table 28 in previous section.

Both novice and experienced readers' ratings on the easiness of task were not affected by which system they used (PT1); they all thought that both interfaces were easy to learn (PS1, mean rating>4.0), and easy to use (PS2, mean rating > 4.2) and there was no significant difference between the two systems. Results in Table 29 confirmed this by showing the participants' opinions on exit questionnaire.

All the participants understood the user interface (PS3) of iSee (mean rating > 4.0) as good as iSearch (mean rating = 4.3), and they thought that both systems were more or less helpful in completing the tasks (PS5, mean rating > 3.8), and in understandings of the stories in the tasks (PS6, mean rating > 3.2), but the differences between systems were not significant.

Interestingly in PS6, novice readers' mean rating of iSee (3.08) was slightly higher than iSearch (2.92) on average, although not statistical significant. As '3' is the middle value of the 5 point rating scale we used, this result shows that iSearch is slightly less helpful as judged by novice readers.

Participants were more or less satisfied with their answer to both tasks (PT2, mean rating >3.0).

As predicted, novice readers did not have much previous knowledge of Biblical stories to help them completing the tasks (PT3, mean rating <2.5 for both Gospel Harmony and Story-Theme Map task). Experienced readers rated this question slightly higher (mean rating > 3.0) on average.

7.5.5.3 Summary

As far as the software usability measures were concerned, the two systems were found to be equally easy to learn, easy to use, helpful with the task, enjoyable to use and left their users satisfied. We were unable to reject the null-hypothesis HS. Our results showed that the iSee was at least as good as iSearch in these usability metrics, if there was little evidence of any significant differences.

7.5.6 Hypothesis HP – Preference

Upon the completion of tasks using both systems, participants were asked to compare the system and give further comments. On being asked to choose their preferred system, the participants' opinions were divided equally between the two systems. Table 30 shows the results:

Table 30 Counts of Participants' Preference

Knowledge	iSee	No Difference	iSearch
Novice	6	1	5
Experienced	5	1	6
All	11	2	11

In their additional comments, eighteen participants commended they liked the query search tool in the iSearch user interface, while fourteen said they liked the story linking in iSee. Eight participants commented the lack of flexibility in the searching aspect in iSee; four participants commented on the lack of accuracy of search engine in iSearch. Seven said that entering the query in iSearch was too much work, and there might be easier ways input long texts for the query, e.g., by selecting text then click on the area. We will consider these in future developments.

We were unable to find good evidence to support our Hypothesis HP, that participants preferred iSee over iSearch for completing their reading and comprehension tasks.

7.6 Further Findings

We have discussed the findings of the study in 7.5 using quantitative and qualitative data analysis relating to the hypotheses. We were also interested in the way the participants used the systems and the strategies that they adopted in completing their tasks. This data might provide further insights into the study.

7.6.1 Query Length

We first looked into the system log data relating to the queries that participants used for their task. For iSee, searches were done through the list of story titles built into the user interface. Users could search for related stories by clicking on one of the story titles, when a list of ranked related stories was retrieved and displayed, using the story as a query. For iSearch, users were provided with a traditional query search box. Users could search for related stories by entering text in the query box and then clicking on the Search button. Unlike traditional keyword searching tools, which only accept a limited number of words, iSearch would accept unlimited text in its queries. These long text query searches were demonstrated during the training session in the experiments.

Table 31 Frequency of Query Word Count

Query Length	iSee	iSearch
Key words (up to 5 words)	N/A	38
Sentence (5 – 20 words)	N/A	2
Paragraph (20 – 50 words)	N/A	10
Whole story (over 50 words)	326	198
Total	326	248

A total of 326 searches were carried out using iSee, and 248 using iSearch. Table 31 shows the frequencies for the different systems, showing the number of searches for different levels of query word count. This showed evidence of two distinct strategies for using iSearch – using short keyword queries, and copying in the whole story.

Figure 43 shows the distribution of query types by the participants. Participants 1 – 12 were experienced readers, 13 – 24 were novice readers. Words, sentence and paragraph queries are combined to simplify the data. This shows that fourteen participants used keyword queries and five of them tried it only once or twice. All but one participant used whole-story queries; and ten participants only used whole-story queries. Thirteen participants used at least two query modes (shown by both the blue and the yellow bars).

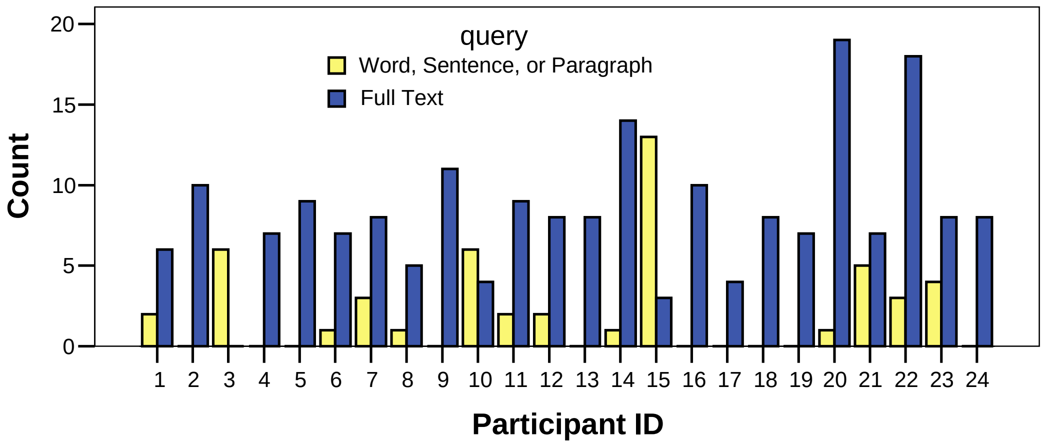


Figure 43 Distribution of Query Mode Frequency

Chi test showed that the number of whole-story queries used in iSearch was significantly larger than the number of keyword, sentence, and even paragraph queries (p<.01), showing the demand for and popularity of long query searching in story searching and comprehension tasks. Compared to the functionality of iSee, it is clear that most participants used iSearch as if it was iSee, taking advantage of searching for whole stories. However, only a few participants explicitly saw this advantage of iSearch; they thought iSearch was just a traditional keyword-searching tool, as its appearance implied. We will discuss this further in 8.6.

7.6.2 Strategies

Participants were also asked for any strategies they used in the post-system questionnaire. Eighteen participants indicated that they adopted strategies in completing their tasks using iSearch.

One participant said this after his experiment with iSearch, which was the first system he used:

> *"Yes, I used the passage as a query to find related stories. I used the titles to get an indication of how related the stories were and what the common theme might be."*

This was a common response – six participants mentioned that they used the whole story as a query. The same participant mentioned the following strategy after using iSee:

> *"I used the titles of the stories as clues to main themes. I found the related stories by double clicking on links in Stories Panel, and then choosing the most related stories using the title and some previous knowledge of the Bible."*

Another participant also mentioned clicking on the title of stories. Other participants paid more attention to the book bars and the reference list in the user interface. One participant said this for iSee:

> *"Click as much as I want, and it will not go wrong."*

The following strategies were used commonly for both iSee and iSearch:

- Using the ranked reference list.
- Looking for key words in the passages.
- Following the example of training tasks.
- Using the book bars and chapter bars to browse the book.
- Using the History panel to return to previously viewed pages.

These showed that users of iSearch understood the potential for whole-story searching, and used it as a strategy in completing the tasks, but users of iSee did not pay much explicit attention to the built-in story linking, which was the key characteristic of the system.

This reminds us the quote we used earlier in 3.3: *A well designed user interface is an ‘invisible’ user interface, it disappears to enable the user concentrates on their work, exploration or pleasure* (Baeza-Yates and Ribeiro-Neto 1999; Shneiderman 1987).

7.7 Summary

In this chapter, we have reported the results of a user-centred evaluation of two e-Book reading tools in simulated tasks of information seeking and reading for comprehension activities. Two tools were compared, one based on story listing and linking, and one based on query search. The major findings of our investigation were these:

In the information seeking task, the e-Book with a story linking tool, iSee, was more helpful than iSearch for experienced readers to achieve faster speed in navigation, completion and correct answers. For novice readers, iSee was no better, or worse than iSearch in most of the aspects, although it was also significantly faster than iSearch according to the navigation actions per minute measure. This showed that iSee was significant more efficient than iSearch, especially for experienced readers (7.5.3).

While the results were not statistically significant, the general trend was that the information seeking task effectiveness, as measured by the proportion of completion, the number of correct answers, and precision, etc., was on average better when using iSee compared with iSearch.

Note that the mean value of completion was quite high, which indicated a 'ceiling effect', especially with experienced readers. In statistics, 'ceiling effect' refers an effect whereby data cannot take on a value higher than some 'ceiling'. In the case of a ceiling effect, the majority of scores are at or near the maximum possible for the test. The ceiling effect on the completion showed that either the task was too easy, or the time given was too long. This could be resolved by increasing the number of tasks and reducing the time allowed.

In the reading for comprehension tasks, an e-Book user interface with a story linking tool was more effective than one without, for supporting users' comprehension task, as measured by the accuracy and relevance of the themes identified by readers. Participants using iSee have better understanding of the stories and themes than using iSearch. Moreover, novice readers found iSee more helpful than iSearch in that the themes they found were as good as those of experienced readers when they used iSee, and were significantly worse when they used iSearch.

Although the iSee had a novel story linking tool, and iSearch had a traditional query search tool, the two systems were identical in usability measures, i.e., they were both easy to learn, easy to use and as helpful, as ranked by users. Participants' satisfaction of their performance was not effected by which system they used. Overall participants liked both systems equally and enjoyed the experiments.

In addition, query analysis revealed that significant number of participants used whole story queries in iSearch. This reduced the difference between the two systems and weakened the ability to test these differences. On the other hand, this also provided evidence that the concept of iSee, with features like searching and linking stories in e-Book browsing too, was well understood, quickly adopted, and used smoothly in the problem solving activities.

In addition, from observation of the participants during the experiments, we suspect a more structured training on the iSee before their tasks might help improve the performance of iSee even better – it will help participants to understand the new functions of story linking in the iSee user interface, and the scaffoldings of narrative structure detection of the e-Book.

8

DISCUSSION

8.1 Introduction

In chapter 1 we raised a number of research questions after a brief analysis of the problem in computer-aided reading and learning, and outlined the work described in this thesis, aimed at addressing these questions. This study was novel in that it created a user-centred, theory-based, search engine powered e-Book user interface design framework, and carried out a set of evaluations on the effectiveness of this framework.

This chapter begins by revisiting the research questions outlined in chapter 1, and describing how this study has gathered evidence relating to each question. It then briefly touches on the design goals which follow from these questions, and then concludes with a discussion of some of the issues raised by iSee, and suggests some future directions for addressing those issues.

This thesis investigated five primary research questions. These research questions were proposed to examine how to aid comprehension and navigation in an e-Book. We will revisit them and discuss our findings in the following sections.

8.2 Design Framework

Research question: *How can a framework be designed that incorporates reading for comprehension theory, information retrieval innovations, and software user interface design principles in the design of an e-Book software application?*

This is a key research question. We tackled this question in 4.1, and then expanded on the issues with more details and proposed solutions in the chapter 4 – 6.

Most e-Book and hypertext design guidelines concentrate on issues such as how much text should be contained in a node, what hypertext features to use, and how the information should be structured, yet do not provide concrete design rules based on reading theory (see section 3.1). Moreover, in the past the researchers and designers of e-Book user interfaces focused on the display, navigating and querying functions, and paid little attention to supporting understanding of the book contents (see section 3.3).

Having looked at design framework examples in software engineering (Deutsch, 1989, Welsh et al. 2000), user interface design (Marchionini, 2000; Lee, 2002) and evaluation (Ingwersen and Jarvelin 2005), we constructed a simple yet innovative design framework for an e-Book user interface (see section 4.1). This framework takes into consideration reading tasks, comprehension theory, and topic detection and tracking techniques, and draws these components together to support implementation and future reuse. It also guides user interface design and evaluation. This design framework allows us to design and provide many alternative types of scaffolding for different reading tasks, based on comprehension theory, story segmentation and story linking.

We will discuss each aspect of this design framework in detail in the following sections, discussing specific research questions on these aspects in turn.

8.3 Detection of Multiple Structures in an e-Book

8.3.1 Overview

Research question: *How can the organizational and narrative structures of e-Book be semi-automatically discovered?*

This question was addressed in three steps:

- Reviewing the state of art of e-Book, hypertext book and other reading tools in the field (chapter 3)
- Analyzing and discovering the organizational structure of an e-Book (4.2 and 4.3)
- Designing algorithms to semi-automatically discover the narrative structure of e-Book (chapters 5 and 6)

8.3.2 Organizational Structure of an e-Book

We applied text processing technique to detect the organizational structures of two corpora of electronic Bible: the King James Version in plain text format and the New International Version in HTML format, as described in section (see 4.2.2 and 4.3.1).

Since the two versions have different format, we wrote a Java program to process the text format corpus and a Perl program to process the HTML format corpus, as the latter has better pattern recognition operators. Structural headings, paragraphs, and sentence breaks have special tags assigned and thus could be identified. Due to the special characteristics of the collection, we identified within the original electronic text the book, chapter and verse information, which established the organizational structure. Both corpora were organized in the same way: each chapter was saved in a single file and stored in a folder identified by the book.

We then computed the overall number of books, the number of chapters in each book, and the number of verses in each chapter, and used this information to support visualization of the organizational structure. This was described in section 4.3.

This demonstrated methods that could detect the organizational structure of an e-Book in both plain text and HTML formats in an electronic Bible. This organizational structure is common for many types of books (apart from the verse information, which could well be replaced with natural sentences).

8.3.3 Semi-Automatic Detection of the Narrative Structure

We then used a more complex and more innovative process to discover the narrative structure of an e-Book.

In chapter 5 we reported an algorithm that could be used to subdivide long text into short topic segments. In previous work (TextTiling, see Hearst 1993), a subject boundary could be detected by comparing neighbouring text windows on both sides of each sentence. We were able to improve the original TextTiling algorithm significantly by using a symmetric divergence measure to evaluate the probability distributions of terms in the two text windows. We demonstrated the improvement in this approach by comparing it with two baseline systems on a complex narrative text corpus. One system used an ad hoc Cosine-distance measure, as did TextTiling, and the other used an asymmetric Kullback-Leibler model.

In chapter 6 we continued to discuss techniques for identifying similar stories in a large story collection, developed from our main text corpus. We achieved this by incorporating the symmetric divergence measure in a narrative engine based on ideas from information retrieval. The Harmonizer (as we called the engine) compares each story in the collection with all the others and computes a similarity score for each pair. It then ranks the similar stories and selects the most related stories to form a narrative thread. We demonstrated the advance of this approach by comparing it with a baseline system, powered by Lucene (section 6.3.2.3).

Both of these approaches relied on recent development in language modelling (Croft and Lafferty 2003) and Topic Detection and Tracking (Stokes 2004) . The results of the evaluations provided evidence that:

- The distribution of terms is important indicator of topic or subject of a narrative text.
- A symmetric divergence measure is effective in modelling the comparison narrative texts.
- The narrative structure of an e-Book can be detected by partitioning the text into narrative segments by its topic, and then creating a narrative thread of similar or related segments for each segment in the book.

However, in the evaluation of our story linking algorithms, as part of the narrative structure detection process, there were other factors of the comparison systems need to be clarified and explored with more details. For example, we do not have enough evidence that the different performance between the systems was achieved by the similarity measure alone. We would also like to explore the factor of inverted document frequency (*idf*) in the story linking detection task with a larger corpus, for example, within using the whole Bible as corpus, instead of the Gospels.

8.4 Presenting the Multiple Structure Information of an e-Book

Research question: *How should multiple structures be presented in an e-Book user interface? And to support it, how the navigation of narrative structure with the navigation of organizational structure of e-Books be integrated?*

We addressed these questions in chapter 4, by adapting information visualization technique and common user interface design guidelines (section 3.1).

We used storyboard designs on paper to form our earlier user interface prototypes, and asked readers of e-Books for their feedbacks in informal discussion and brain-storming. We did not report this process in detail in this thesis, but showed the storyboards we used in Appendix A1.

We used this approach to design a unique information visualization tool to provide an overview of the organizational structure of e-Book and support navigation, inspired by a range of systems described in section 3.2, including SuperBook (Landauer, Egan et al. 1993), TileBars (Hearst 1995), and ProfileSkim (Harper, Coulthard et al. 2002). We also included navigation buttons to aid browsing, inspired by the BibleGateway user interface (Figure 12, page 43).

To avoid a navigation overload problem, which is common in hypertext books, we provided various representations of the narrative structure of the e-Book in simple ways, instead of using sophisticated graphical presentations. These design approaches are presented in section 4.4. For the subsequent user evaluation we did not use all the representations developed. Instead we focused on the methods that revealed the narrative structure of e-Book by presenting a ‘mini’ table of contents for each chapter, and linking similar stories in a ranked reference list. The narrative structure of the e-Book was semi-automatically detected by Harmonizer, as reported in chapter 6.

8.5 User Evaluation of an e-Book with Simulated Reading Task

8.5.1 Overview

Research Question: *How can an e-Book application be evaluated in real world simulated reading and comprehension tasks? Two secondary questions within this are: How can meaningful and purposeful real world simulated reading tasks be designed? How can the users’ comprehension performance with applicable measures be evaluated?*

These research questions were central to this study, and were the most challenging to accomplish. We addressed these questions in three steps:

- Reviewing research in text comprehension, cognition, and problem solving theory, on the state of art tools designed to support reading and information seeking, and on appropriate evaluation approaches (in chapter 2 and 3).
- Designing specific information-seeking tasks within the e-Book context, and corresponding evaluation measures, inspired by other information-seeking evaluations, for example, Hyper-TextBook (Crestani and Ntioudis 2001) and ProfileSkim (Harper, Koychev et al. 2003).

- Designing comprehension tasks within the e-Book context and corresponding evaluation measures, inspired by the concept of concept mapping (Novak 1998), the dynamic memory model (Schank 1999), Bloom and Anderson’s taxonomy (Anderson and Krathwohl 2001; Bloom 1956), and directed reading and thinking activities (Bear and McIntosh 1990).

These steps were accompanied by the development of ‘ground truth’, which was adapted from literature resources (7.1.3). We will discuss in details below the findings from the user evaluation.

8.5.2 Information Seeking

How people search and select what to read demonstrates their understanding of the subject. Providing effective information seeking tools can help people to understand what they read (Chi, Hong et al. 2004; Egan, Remde et al. 1989; Pirolli and Card 1999; Puntambekar, Stylianou et al. 2003). One of the major findings of this study was that iSee, an e-Book user interface with a story linking tool, speeded up experienced readers in their information seeking. Using iSee, novice readers also achieved faster navigation speed. However, iSee was no more, or less effective than iSearch, as measured by the effectiveness measures: i.e., time, navigation actions, searching actions, pages viewed, and reading time are important measures used to evaluate information seeking system’s effectiveness. Apart from what we have reported in chapter 7, there are a few issues of these measures we would like to discuss.

For navigation actions and visited pages, is the more always the better? In this study, we assumed that the more navigation actions, the better understanding, because according the comprehension theories (Schank 1991; 1995; 1999), the more you read, the better back ground knowledge you have (Novak and Gowin 1984), the more cases you collect, the better reasoning skills you have(Aamodt and Plaza 1994; Schank 1982a). However, it has also been observed that more navigations choices would create ‘navigation overloading’, which confuses users and readers rather than help them (Edward and Hardman 1989). Therefore, how to distinguish those ‘purposeful’ navigations from those ‘overloading’ navigations became an interesting question. Here, ‘purposeful’ navigation means when a user made a right choice of what to read or visit next; ‘overloading’ navigation means when a user made a regrettable choice of what to do next. One way to distinguish them is to check if the user immediately changed their mind after a navigation action, i.e., when they went back to previous link.

For searching actions, how to distinguish a local searching and a global searching? Here we use ‘local’ to mean the within the document search; and ‘global’ to mean the search within the collection. In this study, we used chapters as individual pages in the e-Book reading window. A chapter could be long or short. When a chapter is long, not all the stories can be displayed in one screen. User has to scroll the chapter to find the story. In our user interface, we have provided loading tool to help readers locate the story they were searching for. In iSee, when a reader double click on a story title to get a list a related stories, the chapter will be scrolled to where the story was, and the related stories will be displayed in the reference panel. However, we observed that often a user clicked on the story title to locate the story in the chapter, rather than ‘search’ for the related stories over the collection. One way to resolve this is to use single click for a local search, and double click for a global search. There might be other ways of doing it.

For reading time: how to distinguish real reading time from navigation and question answering time. Since we used the paper and pen mode for questionnaire and task sheet answering, reading on the screen was somehow mixed with writing on the paper.

Reading time has been used as a measure for IR system performance. For example, Morita (Morita 1996) suggested to use reading time as an implicate feedback for document relevance in the information filtering system. She found a strong tendency for users to spend a greater length of time reading those articles rated as interesting, as opposed to those rated as not interesting. However, Kelly and Belkin (Kelly and Belkin 2001) argued that the reading time might be effected by tasks, document collection and searching environment. In their evaluation with IR systems, they found that the difference between reading relevance document and irrelevance document was not significant, and suggested that when given a complex task (in which readers were required to construct queries, evaluate, save and label document all within a specific time period), readers might be compelled to spend time on reading. In their study the reading time on both relevant and irrelevant documents was so low (about 25 second per document).

In this study, we assumed that a shorter reading time is more effective than a longer one. We faced a similar situation as Kelly when we gave readers limited time to construct a Gospel Harmony table. Users may have felt stressed to perform as quickly as possible because of the current experimental protocol (7.3.2). It will be interesting to find out that with particular e-Book systems and tasks, whether a shorter or a longer reading time will improve readers' understanding. Such test will need further evaluation and discussions.

Most of the participants rated the easiness of the task positively (a mean of 4 out of 5), and their performance on the correct answers and precision were quite high, implying a possible ceiling effect. This indicated that the task was too easy or the time given is too long. As a result, we were not able to demonstrate that iSee was more effective than iSearch, although we did find that the task efficiency was improved with iSee compared to iSearch.

Overall, the tasks we designed for information seeking and the corresponding evaluation of the system demonstrated that providing narrative navigation aid could help readers in their information seeking process and thus improve their understanding of the subject.

8.5.3 Reading for Comprehension

Research in text comprehension has investigated people's reading and cognitive behaviours, and developed a number of different models of explanation (Hoover and Gough 1990; Kintsch 1998; Schank 1995; 1999) (see sections 2.1 and 2.2). Although these findings have impacted on the development of hypertext books, neither hypertext books nor e-Book have established a theory or comprehension model of their own. In spite of the increasing popularity of these tools being used in reading and learning, most of the evaluations provide no evidence that these tools are more effective than paper books, and most of the design guidelines only advise on display and formatting issues, often based on common sense rather than on content organization based on theory. This could be because most practitioners in the field are computer scientists rather than psychologists or learning technologists. There is some evidence of a change nowadays, though.

One of the main aims of our study was to find the relationship between the narrative structure of an e-Book and the effectiveness of readers' comprehension. We wanted to understand whether the in-depth narrative structure helped readers in their reading, and whether navigating through narrative linking helped readers to acquire a richer understanding of the stories in a book. We reviewed related work in sections 2.1.3 and 3.2.4. We were interested not only in how hypertext or e-Book systems were designed, but also in how these systems were evaluated with real users. As comprehension is a complex and rather mysterious process, it is an important challenge to find a simple yet sufficient way of measuring the outcome of comprehension, so that we could use these measures to evaluate a system's effectiveness in assisting reading.

The evaluation tasks for comprehension outcome reported so far included:

- Writing up an essay about what have been found or learnt (Halttunen and Järvelin 2005).
- Answering open-ended and/or close-ended questions, for example, a quiz" (Crestani and Ntioudis 2001; Puntambekar, Stylianou et al. 2003).
- Completing simulated tasks, for example, creating a subject index for an e-Book (Harper, Koychev et al. 2003b).
- Filling in a set of questionnaires (Kelly, Harper et al. 2007).

The closed questions, for example, single- or multiple- choice questions, are easier to evaluate (if hard to create), and meanwhile they can carry less information than open questions. Open-ended questions such as essays and free comments, however, are difficult to evaluate, as the evaluator may have different views to those of the participants.

To assess this research question, we needed a task that aims not only to find how a reader understands a particular story, but also how she interprets the relationship among related stories. Our comprehension task – the Story-Theme Map was inspired by the dynamic memory model, concept mapping, directed reading activity, and Bloom and Anderson's taxonomies (chapter 2).

With a view to evaluating readers' comprehension of the narrative structure of an e-Book, rather than its' language, we chose to focus on the middle levels of the taxonomy structure (which are: understanding, applying and analysing) and developed evaluation methods that applied the suggested verbs: 'describe', 'identify', 'choose', 'demonstrate', 'sketch', 'compare', 'differentiate', 'distinguish', etc. (see section 2.3.2).

We explored a few alternatives before the design was finalized. For example, we invited an author to write a Bible study note for our participants. The questions in this note were similar to those widely used for Bible study and discussion. Questions normally started with: *"Read…, what do you think about…?"; "What is the main idea of …..": "How did something happen and what impacts did it bring to our culture or history?"*; and *"In your personal experience, can you give an example of….?".*

These kinds of question, also typically used in literacy education, are suitable for encouraging people to relate to what they have just read. However, these questions are more on a one-to-one basis (relating to a story/passage with personal experience and knowledge) rather than on a many-to-one basis, (relating to many stories/passages and generating a higher level theme or theory). We might call the one-to-one questions ‘static’ questions, and the many-to-one questions ‘dynamic’ questions. In fact the latter is usually accompanied with dynamic critical thinking, and answers to such questions require strong reading experience – either through many years reading to establish many one-to-one based links, or through intensive training on research and critical thinking.

According to Anderson’s taxonomy, these ‘static’ questions fall into the two categories at the bottom of the learning structure, which are remembering and understanding. These are the starting points of further- and higher-level comprehension: applying, analysing, evaluating and creating.

Results of two pilot studies demonstrated that these ‘static’ questions were not suitable for evaluating reading systems, since the reading of one passage and thinking what it might bring requires minimal assistance from a system. This might explain why so many experiments failed to find evidence that a technology-based system is more effective than a paper book in assisting reading (section 2.1.3).

Halttunen and Jarvelin (2005) used essay and concept mapping to evaluate students’ learning outcomes from two information retrieval learning environments, an experimental information retrieval game, and an operational information retrieval system, in a university teaching semester. 120 students attended the course and 47 completed the experiments. This evaluation and its task are natural and dynamic, but the span of the study, the participants, and the experimental settings were too specific to replicate. However, using concept mapping to score students' essays was an interesting way of assessing the learning outcome. It is not clear, though, why did not they ask the students to construct a concept map themselves rather than writing an essay. Although the inter-rater's reliability between two raters was statistically high, the amount of effort put in by the raters was also high.

We designed the Story-Theme Map task to evaluate readers' comprehension. To complete this task, readers had to first read a given story carefully, then identify the core message of it, and find related or similar stories in the e-Book. They were then asked *to* illustrate their findings on an empty Story-Theme Map, to group similar stories by lines and write down the main themes based on a group of stories.

There were no right or wrong answers. Readers were encouraged to read as many as stories as they could, and identify up to three themes in a set amount of time. More than half the participants identified three themes (section 7.5.4.2); however the difference between the two systems was not statistically significant.

The evaluation of the relevance of themes was also a big challenge. "If there are a thousand readers, there are a thousand Shakespeare(s)" (Anonymous). Stories and themes could easily be interpreted in various ways by different readers, and for different purposes. In this study, the results of these Story-Maps were coded and evaluated by three judges to judge the rank of relevance of each theme (section 7.4.2 and Appendix A4). At first, the inter-rater's reliability was low, so a consensus session was carried out. The three judges met to discuss every theme that was ranked differently, and agreed a final rating for the theme. The consensus session is often used when inter-rater reliability is low, e.g., in Kelly, Harper and Landau (2007). This enabled us to carry out further data analysis on the Story-Theme Map tasks. However, this also left us with questions about how to establish more reliable measures for comprehension evaluation. More studies are needed in the comprehension and the evaluation of comprehension theories.

These results provided evidence that iSee was more effective than iSearch for assisting readers' comprehension. This showed that iSee was effective at providing narrative structure information to readers, which improved their dynamic thinking in a limited time task. Moreover, providing the narrative structure of an e-Book in addition to the organizational structure of the e-Book could fill the gap in poor readers' knowledge and experience, and help them to gain the same level of insight as experienced readers in a short time.

8.5.4 Background Knowledge

The role of background knowledge towards comprehension is an interesting and complicated one. Many studies confirmed that readers' background knowledge played an important role in their comprehension (Britton and Gulgoz 1991; van Dijk and Kintsch 1983), and an ill-designed system will easily confuse its users rather than help them (Edward and Hardman 1989; Puntambekar, Stylianou et al. 2003).

Our user evaluation results showed that, for the information seeking tasks, when given same amount of time, the novice readers performance was as good as experienced readers' using either iSee or iSearch. Meanwhile, with a story linking tool, iSee helped experienced readers achieve a faster speed than iSearch. This also showed that story titles, as listed in iSee user interface might have helped experienced readers to recall their previous knowledge. However we failed to find evidence that the structural information, like story titles listed in iSee, would be more helpful to novice readers, since their performance on both systems was identical. When searching for reasons of this, we found that users have been intensively use whole stories as queries in their information seeking task. We will discuss this further later.

For the comprehension tasks, we found that when given slightly longer time, iSee was significant more effective to help novice readers understand the related stories and themes. The themes they found were as good as experienced readers' ones when they used iSee; and they were significantly worse when they used iSearch. This provide good evidence that iSee was helpful to novice readers trying to understand and make thematic links. Note that those comprehension tasks request both information seeking skills and analysing skills as readers had to first find related stories, secondly identify the relevant themes.

Using Anderson and Bloom's taxonomy structure (Anderson and Krathwohl 2001), we could find that the information seeking tasks belonged to a lower category than the comprehension ones. Information seeking tasks request basic text understanding skills, while the comprehension tasks request understanding, comparing, applying and analysing. This showed that when the tasks are complex and intensive, novice readers need more support from the reading software. Our iSee met their demands by not only providing structural information, but also providing the story linking tool to reveal narrative structures of e-Books. Such tools provided scaffolding for readers to read, explore, understand and analyse the text when they browse the e-Book.

8.6 More on Query Length

Our discovery of query length for iSearch (7.6.1) was an important result for this study. On one hand, it weakened the contrast between the two systems that we aimed to show. Our original intention was to find differences between iSee, a story-linking tool that supported narrative structure navigation, and iSearch, a traditional keyword-searching tool, that assisted readers browsing and searching for related stories or passages in situ, but it turned out that we were in fact comparing iSee and ‘iSee minus’, where iSearch was used extensively as a whole-story search tool (essentially the same as iSee). This might be the reason why some of the differences between the two systems were not significant as we had predicted, i.e., the number of related stories and the number of themes identified in the Story-Theme Map task.

On the other hand, this does provide indirect evidence that the concept of iSee, with features like searching and linking stories in an e-Book browsing tool, was well understood, quickly adopted, and smoothly used in problem solving activities.

Query length in best-match information retrieval (IR) systems is well known to be positively related to effectiveness in the IR task, when measured in experimental, non-interactive environments. However, in operational, interactive IR systems, query length is typically very short, just a few keywords (Jansen, Spink et al. 2000). Researchers in interactive information retrieval endeavoured to find way to increase query length and improve the IR search effectiveness. For example, Belkin, Kelly et al. (Belkin, Kelly et al. 2003) investigated a query elicitation technique to increase initial searcher query length. Results showed that the technique resulted in increased user satisfaction with the search, compared to a baseline system, and that query length was positively correlated with user satisfaction with the search.

In our user evaluation, however, users automatically chose to use unusual long queries – the entire story, when they felt it was more effective for their tasks, and when the system accepted it.

8.7 Narrative Text Corpora and Ground Truth

Research question: *How can the performance of an application in a previously unexplored domain for lexical cohesion analysis, such as the narrative literature, be evaluated and measured?*

We developed novel narrative text corpora and used them in system experiments and user evaluation, as described in chapters 4, 5, 6 and 7. The results of these studies provided evidence that the corpora were sufficient, adaptable and reusable for different purposes.

As we mentioned in section 4.2 that we used an Internet version of the King James Version Bible, and we later obtained permission to download an Internet version of the New International Version Bible (4.2.2). We then partitioned these texts into the desired corpora for our evaluations. All in all, we created the following data sets:

- A KJV Bible that was organized in folders identified by book, files identified by chapter number, and lines identified by verse number, in text format.

- An NIV Bible that was organized in folders identified by book and files identified by chapter, in HTML format.
- An index and a collection of 2128 stories from the whole Bible, identified by the editors of the NIV Bible, in HTML format, for story searching and indexing (chapters 4 and 7.1.3)
- An index and a collection of 426 stories from the four Gospels of the Bible, identified by the Gospel Harmony editors, in text format, for evaluating the tools used to detect narrative structure.

We used these corpora in each of the system prototype (chapter 4) and evaluation stages: story segmentation (chapter 5), story linking (chapter 6), and user evaluation (chapter 7). Such corpora and ground truth development proved to be reliable, effective, and purposeful in all these tasks.

Initially we wanted to use 11 books from Bible for the user evaluation: the first five books in the Old Testament and the first six books in the New Testament. But two pilot studies revealed that this corpus was too narrow for narrative structure browsing. Although these 11 books are rich in stories, character names and events, the stories are locally distributed – similar stories are usually distributed within the same book or in neighbouring books. Through two pilot studies we found that this set of 807 stories was not sufficient to encourage global browsing. Users were guided to closely-related stories rather than globally-related ones – which we believed to be better indicators of themes.

To address these concerns, we then sought permission to use the whole Bible in NIV and generated an index of 2128 stories; this resulted in a much wider distribution of related stories.

In section 7.5.4.4 we presented the results of the studies on the distribution of the identified related stories. Participants using iSee identified significantly more related stories within the same book or section. Participants using iSearch identified slightly more related stories in other books or sections on average, but the difference was not statistically significant. This might have been caused by the special characteristic of the corpus we used, since the different books and section in the Bible spanned a great period of time in history. To explore the effect of corpus on the reading tasks, we need choose more stories and design more tasks, and evaluate with a wider range of readers.

9

CONCLUSIONS AND FUTURE WORK

9.1 Conclusions

Our research on e-Book reading tasks left us with two important insights:

- Readers need support to help them understand the organizational and narrative threads structure in the book for better comprehension.
- This support can be achieved through innovative information retrieval and visualization techniques based on the development in comprehension theories.

In this research, we have devised two compelling scenarios for using e-Book software to perform reading and studying tasks, showing the importance of: (1) access to the overview of the organizational structure in the book; (2) the ability to generalize the themes based on the narrative threads structure in the Bible.

On reviewing of our research questions:

- We described a novel way – a design framework – to design and evaluate e-Book user interfaces. We used theories of text comprehension to design iSee, an innovative e-Book environment that provides users with scaffolding to help them understand and navigate the organizational structure of an e-Book.

- We employed story segmentation and story linking techniques to semi-automatic detect the narrative structure of e-Books.
- We designed user interface prototypes of e-Book to present organizational and narrative structural information of e-Books in various ways.
- We designed evaluation methodologies and conducted evaluations of our system development in every stage. We reported in detail the evaluation of our story segmentation algorithms, the story linking algorithms, and the user evaluation for e-Book interfaces.
- We developed a number of text corpora and ground truth to support the evaluation.

The most significant findings of this thesis were:

- Scaffolding of the multiple structural information of an e-Book helped readers to understand stories and themes of the book.
- Scaffolding of the multiple structural information of an e-Book improved readers' speed in their information seeking tasks.
- Using a symmetric divergence measure improved the performance of story segmentation technique.
- Using a symmetric divergence measure also improved the story linking algorithms, in detection of the narrative structure of e-Books.

The results of this research demonstrate beyond doubt that computing technology can provide effective scaffolding support for comprehension, in the context of an e-Book.

This technology has the potential to considerably improve the understanding and use of complex texts in an online learning environment.

9.2 Future Work

The opportunities for the future applications of these technologies are also quite promising. The e-Book prototypes reported in chapter 4 could be developed as a stand-alone e-Book software, like Microsoft Reader; or an online document-reading tool; or a plug-in for digital libraries, etc. This study is by no means over. There are many issues that remain open to further research, including:

- The effectiveness of the design framework needs to be evaluated with more systems and tasks in the field.
- The methodologies need to be evaluated with other types of text corpus, for example, textbooks or academic journals, to find out to what extent these methodologies can be applied and used to improve reading for comprehension in different fields.
- The method of evaluating comprehension is an interesting field for future development. Although we developed some effective methodologies, there is plenty more that can be done to improve it. We need more choice of tasks and reliable measures to ensure the effects of outside influences (such as the evaluator) can be kept to minimum, and the differences between systems could be measured accurately. We also need to explore of the usefulness of these effective measures not only towards improving e-Book systems, but also towards improving comprehension.

- It will be interesting to do a natural, longitudinal study on usage of iSee. We could give the software to real book readers and computer users and ask them to send back their system logs after a period of time. In this case, there would be no specific tasks. We would be more interested in seeing how the user would use it without being watched over.

- The impact of characters in narrative text deserves further investigation. In chapter 5 we were able to demonstrate that treating character names as normal words was significantly more effective than ignoring them, but we were unable to show that adding additional weighting to character terms would improve the story segmentation performance.

- The efficiency and effectiveness of long query searches might be explored with other information retrieval applications and corpora. In addition, a good way to pass long text queries to the search engine, besides typing and copying/pasting, is needed.

- We would like also to improve the evaluation methods for story linking algorithms. For example, explore the factor of inverted document frequency (*idf*) in the story linking detection tasks, either by introducing a third system that simply uses an asymmetric measure, to distinguish from Lucene and Harmonizer, and/or to develop a larger corpus.

- We are in a process to analyse the data from participants' task sheet, for example, for a particular story, what themes have been identified by different readers, and how many of them were worded in a similar way, for example 'forgive' and 'forgiveness'. We believe such data would help us to know further how the readers identified related stories and themes, and how we could develop automatically constructed Story-Theme Maps for the e-Book readers.

- In addition, the measures of effectiveness of story-linking needs to be investigated. As the IR relevance measures, Precision, Recall and F-measure might not serve the need for evaluating the relatedness of stories.

We currently plan to publish iSee as open source software and make it available online to assist readers browsing, reading and searching for online books. This will widen the usage to a larger community and help both disseminate this work and gather more evidence about reading behaviour in future.

9.3 Goals Revisit

In chapter 1 we proposed three design goals of an e-Book user interface, namely that we aim to help readers:

- To know the organizational structure of e-book quickly via a well designed interactive visualization tool.
- To discover the narrative structure and identify the main themes of the book.
- To gain better understanding of the book structure and the contents.

We achieved the first goal by two steps:

- Using text processing approach to semi-automatically discover the organizational structure of an e-Book.
- Using information visualization to present the organizational structure.

Meanwhile we also provided traditional navigation buttons to aid browsing in addition to the visualization tool. This is reported in section 4.3.

We achieved the second goal by three steps:

- Semi-automatically detect the narrative segments of an e-Book, as reported in chapter 5.
- Semi-automatically detect the narrative structure of an e-Book by linking similar narrative segments together, as reported in chapter 6. And
- Providing various scaffoldings to support readers to discover the narrative structure of an e-Book, as reported in section 4.4.

The third goal was on a higher level than the other two, and was achieved in three steps:

- Analysing the reading for comprehension theories and practices, in chapter 2
- Reviewing the state of art design of e-Book user interface, in chapter 3, and
- Constructing a design framework that draws the above aspects in the design of an e-Book user interface to support reading tasks, in section 4.1.

We explored different ways of scaffolding of the narrative structural information, for example, providing a list of main character names and a highlight function for the names in text, providing an index of stories in each chapter, and providing links of similar stories for each story.

Reading and comprehension were the original terms of engagement for this research. We have achieved our goals of identifying and developing techniques that make reading e-Books more efficient and more enjoyable. The user evaluation study produced important and constructive insights and possibilities for future e-Books development, which also means that there is a lot of work yet to be done. These are important challenges in the field of information retrieval, digital book and libraries, and human-computer interaction.

REFERENCES

Aamodt, A. and E. Plaza (1994). Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. AI Communications, IOS Press **7**(1): 39-59.

Agirre, E., O. Ansa, et al. (2000). Enriching very large ontologies using the WWW. In proceedings of the Workshop on Ontology Learning, 14th European Conference on Artificial Intelligence (ECAI-00). 2000

Agosti, M., F. Crestani, et al. (1995). Automatic Authoring and Constraction of Hypertext for Information Retrieval. ACM Multimedia Systems **3**(1): 15-24.

Altun, A. (2000). Patterns in Cognitive Processes and Strategies in Hypertext Reading: A Case Study of Two Experieced Computer Users. Journal of Educational Multimedia and Hypermedia **9**(1): 33-55.

Altun, A. (2003). Understanding Hypertext in the Context of Reading on the Web: Language Learners' Experience. Current Issues in Education (Online) **6**(2).

Anderson, L. W. and D. R. Krathwohl, Eds. (2001). A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom's Taxonomy of Educational Objectives. New York, Longman.

Ausubel, D. (1968). Educational Psychology: A Cognitive View. New York, Holt, Rinehart & Winston.

Baeza-Yates, R. and B. Ribeiro-Neto (1999). Modern information retrieval. New York and Harlow, England, Addison-Wesley.

Bear, D. R. and M. E. McIntosh (1990). Directed Reading-Thinking Activities: Four Activities to Promote Thinking and Study Habits in Social Studies. Social Education **54**(6): 385-388.

Belkin, N. J., D. Kelly, et al. (2003). Human interaction: Query length in interactive information retrieval. In proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval SIGIR '03, ACM Press. July, 2003

Black, J. B. and G. H. Bower (1979). Episodes as Chunks in Narrative Memory. Journal of Verbal Learning and Verbal Behavior **18**: 309-318.

Bloom, B. S., Ed. (1956). Taxonomy of Educational Objectives, the classification of educational goals – Handbook I: Cognitive Domain. New York, McKay.

Booth, W. C. (1961). The Rhetoric of Fiction. Chicago, USA, University Of Chicago Press.

Borlund, P. (2003). The IIR evaluation model: a framework for evaluation of interactive information retrieval systems. Information Research **8**(3).

Britton, B. K. and Z. Gulgoz (1991). Using Kintsch's Computational Model to Improve Instructional Text: Effects of Repairing Inference Calls on Recall and Cognitive Structures. Journal of Educational Psychology **83**: 329-345.

Byrd, D. (1999). A Scrollbar-based Visualization for Document Navigation. In proceedings of The Fourth ACM International Conference on Digital Libraries, Berkeley, CA, USA. 1999

Chen, F. R., T. Brants, et al. (2003). Optimizing story link detection is not equivalent to optimizing new event detection. In proceedings of The 41st Annual Meeting of the Association for Computational Linguistics, Sapporo; Japan, Morristown NJ. June 12, 2003

Chi, E. H., L. Hong, et al. (2004). eBooks with Indexes that Reorganize Conceptually. In proceedings of ACM CHI 2004, Vienna, Austria, ACM Press. 24-29 April 2004, 2004

Chignell, M. H., G. Golovchinsky, et al. (1993). Information visualization and interactive querying for online documentation and electronic books. In proceedings of IBM Centre for Advanced Studies on Collaborative research: distributed computing, Toronto, Ontario, Canada, IBM Press. 1993

Chung, J. S. L. (2000). Signals and reading comprehension -- theory and practice. System **28**(2): 247-259.

Crestani, F. and S. P. Ntioudis (2001). User Centreed Evaluation of an Automatically Constructed Hyper-TextBook. Educational Multimedia and Hypermedia **11**(1): 3-19.

Croft, W. B., S. Cronen-Townsend, et al. (2001). Relevance feedback and personalization: a language modelling perspective. In proceedings of DELOS Workshop: Personalizsation and Recommender Systems in Digital Libraries. 2001

Croft, W. B. and J. Lafferty (2003). Language modeling for information retrieval. Dordrecht ; Boston, Kluwer Academic Publishers.

Croft, W. B. and H. Turtle (1989). A retrieval model incorporating hypertext links. In proceedings of Hypertext '89, Pittsburgh, ACM Press. 1989

Cutting, D., A. Bialecki, et al. (2006). Apache Lucene Release 1.9.1. Accessed on 24/03/2006, 2006. http://lucene.apache.org/

Edward, D. M. and L. Hardman (1989). Lost in Hyperspace: Cognitive mapping and Navigation in a Hypertext Environment. Oxford, England, Intellect Books.

Egan, D. E., J. R. Remde, et al. (1989). Formative design-evaluation of SuperBook. ACM Transactions on Information Systems **7**(1): 30-57.

Eick, S. G., J. L. Steffen, et al. (1992). SeeSoft - a tool for visualizing line oriented software statistics. IEEE Transactions on Software Engineering **18**(11): 957-968.

Emmott, C. (1997). Narrative Comprehension: A Discourse Perspective. Oxford, Oxford University Press.

e-Resource (1995). Bible Gateway. Accessed on 27th January, 2003. http://www.biblegateway.com

e-Resource (1997). The Unbound Bible Download Website. Accessed on 12th April, 2004. http://www.unboundbible.org

e-Resource (2000). Name and Word Lists in the Web Bible Encyclopaedia. Accessed on 01/05, 2003. http://www.christiananswers.net/dictionary/home.html

e-Resource (2002). Blue Letter Bible Website. Accessed on 24 Mar, 2006. http://blueletterbible.org/

e-Resource (2004). TREC Interactive Track. Accessed on 12th April, 2004. http://www-nlpir.nist.gov/projects/t10i/t10i.html

e-Resource (2007). CmapTools. Accessed on April, 2007. http://cmap.ihmc.us/

Foltz, P. W. (1996). Comprehension, Coherence and Strategies in Hypertext and Linear Text. Hypertext and Cognition. J.-F. Rouet, J. J. Levonen, A. P. Dillon and R. J. Spiro. Hillsdale, NJ, Lawrence Erlbaum Associates.

Frakes, W. B. and R. Baeza-Yates (1992). Information retrieval: data structures & algorithms. Englewood Cliffs, N.J., Prentice Hall.

Fuhr, N. (2001). Models in Information Retrieval. Lectures on information retrieval. Springer-Verlag New York, Inc.

Galvin, J. C. (1986). Harmony of the Gospels. Life Application Study Bible (NIV), Kingsway Publications.

Girill, T. R. (1991). Information Chunking as an Interface Design Issue for Full-Text Databases. Interface for Information Retrieval and Online Systems: The State of the Art. M. Dillon. New York, Greenwood Press**:** 149-158.

Graham, J. (1999). The Reader's Helper: A Personalized Document Reading Environment. In proceedings of SIGCHI'99 conference on Human factors in computing systems, Pittsburgh, Pennsylvania, United States. 1999

Granka, L. A., T. Joachims, et al. (2004). Eye-tracking analysis of user behavior in WWW search. In proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval, Sheffield, UK, ACM Press. 2004

Gunter, B. (2005). Electronic books: a survey of users in the UK. Aslib Proceedings: New Information Perspectives **57**(6): 513-522.

Halttunen, K. and K. Järvelin (2005). Assessing learning outcomes in two information retrieval learning environments. Information Processing & Management **41**(4): 949-972.

Harper, D. J., S. Coulthard, et al. (2002). A Language Modelling Approach to Relevance Profiling for Document Browsing. In proceedings of The Joint Conference on Digital Libraries, Oregon, USA. July 2002, 2002

Harper, D. J., I. Koychev, et al. (2003a). Query-Based Document Skimming: A User-Centred Evaluation of Relevance Profiling. In proceedings of The European Conference on Information Retrieval Research, Pisa, Italy. April 2003, 2003a

Harper, D. J., I. Koychev, et al. (2003b). Within Document Retrieval: A User-Centred Evaluation of Relevance Profiling. Information Retrieval(7): 265-290.

Hart, M. (1992). Gutenberg free e-Books library.

Hearst, M. A. (1994). Multi-paragraph segmentation of expository text. In proceedings of the 32nd Meeting of Association for Computational Linguistics, Las Cruces, NM. June, 1994

Hearst, M. A. (1995). TileBars: Visualization of Term Distribution Information in Full Text Information Access. In proceedings of The ACM SIGCHI Conference on Human Factors in Computing Systems, Denver, Colorado, USA, ACM. 7 - 11 May 1995, 1995

Hearst, M. A. (1997). TextTiling: Segmenting Text into Multi-paragraph Subtopic Passages. Computational Linguistics **23**(1): 33-64.

Hearst, M. A. (1999). User Interfaces and Visualization. Modern Information Retrieval. R. Baeza-Yates and B. Ribeiro-Neto, Addison Wesley Longman**:** 257-323.

Heinstrèom, J. (2002). Fast surfers, broad scanners and deep divers : personality and information-seeking behaviour. êAbo, êAbo akademi.

Hoover, W. A. and P. B. Gough (1990). The Simple View of Reading. Reading & Writing **2**(2): 127-160.

Hornbaek, K. and E. Frokjaer (2001). Reading of electronic documents: the usability of linear, fisheye, and overview +detail interfaces. In proceedings of The ACM SIGCHI Conference on Computer Human Interaction, Seattle, United States, ACM. 2001

Jansen, B. J., A. Spink, et al. (2000). Real life, real users and real needs: A study and analysis of users' queries on the Web. Information Processing and Management **36**(2): 207-227.

Kaszkiel, M. (2000). Indexing and Retrieval of Passages in Full-Text Databases. Computer Science. Melbourne, Australia, RMIT University.

Kelly, D. and N. J. Belkin (2001). Reading time, scrolling and interaction: exploring implicit sources of user preferences for relevance feedback. In proceedings of ACM SIGIR, New Orleans, Louisiana, United States, ACM Press. 2001

Kelly, D., D. J. Harper, et al. (2007). (to appear) Questionnaire mode effects in interactive information retrieval experiments. Information Processing & Management.

Kintsch, W. (1998). Comprehension: A Paradigms for Cognition. New York, Cambridge University Press.

Landauer, T., D. Egan, et al. (1993). Enhancing the usability of text through computer delivery and formative evaluation: The SuperBook project. Hypertext: A Psychological Perspective. C. McKnight, A. Dillon and J. Richardson, Ellis Horwood**:** 71-136.

Larvrenko, V., J. Allan, et al. (2002). Relevance models for topic detection and tracking. In proceedings of HLT-2002, San Diego, CA, USA. 2002

Marchionini, G. (1995). Information seeking in electronic environments. Cambridge ; New York, Cambridge University Press.

Martin, J. (1990). Hyperdocuments and How to Create Them. Englewood Cliffs, NJ, Prentice-Hall.

Meyer, B. J. F., D. M. Brandt, et al. (1980). Use of Top-Level Structure in Text: Key for Reading Comprehension in Ninth-Grade Students. Reading Research Quarterly **16**: 72-103.

Morita, M., & Shinoda, Y. (1996). Information filtering based on user behavior analysis and best match text retrieval. In proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. 1996

Nevin and Alfred (2002). "Harmony of the Gospels". Accessed on 24 Mar, 2006. http://blueletterbible.org/study/harmony/index.html

Newell, A. (1990). Unified theories of cognition. Cambridge, Mass., Harvard University Press.

Nielsen, J. (1990). Hypertext and Hypermedia. San Diego, CA, Academic Press.

Novak, J. D. (1998). Learning, Creating, and Using Knowledge: Concept Maps as Facilitative Tools in Schools and Corporations. Maqhwah, NJ, Lawrence Erlbaum.

Novak, J. D. and D. B. Gowin (1984). Learning how to learn. Cambridge [Cambridgeshire] ; New York, Cambridge University Press.

Park, J. and J. Kim (2000). Contextual Navigation Aids for Two World Wide Web Systems. International Journal of Human-Computer Interactions **12**(2): 193-217.

Perfetti, C. A. and S. Roth (1981). Some of the Interactive Processes in Reading and Their Role in Reading Skill. Interactive Processes in Reading. A. Lesgold and C. A. Perfetti. Hillsdale, NJ, Lawrence Erlbaum.

Pirolli, P. and S. K. Card (1999). Information Foraging. Psychological Review **106**(4): 643-675.

Ponte, J. and W. B. Croft (1998). A Language Modeling Approach To Information Retrieval. In proceedings of The 21st ACM SIGIR Conference on Research and Development in Information Retrieval, Melbourne, Australia, ACM Press. August 1998, 1998

Porter, M. F. (1980). An algorithm for suffix stripping. Program **14**(3): 130-137.

Potelle, H. and J.-F. Rouet (2003). Effects of content representation and readers' prior knowledge on the comprehension of hypertext. International Journal of Human-Computer Studies **58**(3): 327-345.

Protopsaltis, A. and V. Bouki (2005). Towards a hypertext reading/comprehension model. In proceedings of The 23rd annual international conference on design of communication: documenting & designing for pervasive information, Coventry, United Kingdom, ACM Press. 2005

Puntambekar, S. (2005). CoMPASS (Concept Mapped Project-based Activity Scaffolding System). Accessed on http://www.compassproject.net/info/

Puntambekar, S., A. Stylianou, et al. (2003). Improving navigation and learning in hypertext environments with navigable concept maps. Human Computer Interaction **18**(4): 395-426.

Resnik, P. (1999). Semantic Similarity in a Taxonomy: An Information-Based Measure and its Application to Problems of Ambiguity in Natural Language. Journal of Artificial Intelligence Research **11**: 95-130.

Rosson, M. B. and J. M. Carroll (2001). Usability Engineering: Scenario-Based Development of Human Computer Interaction (Interactive Technologies), Morgan Kaufmann.

Saint-Exupery, A. D. (1943). The Little Prince. Accessed on 27/02, 2004. http://www.angelfire.com/hi/littleprince/

Salton, G., J. Allan, et al. (1993). Approaches to passage retrieval in full text information systems. In proceedings of The 16th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval, Pittsburgh, Pennsylvania, USA, ACM. June 1993, 1993

Salton, G. and C. Buckley (1998). Term-Weighting Approaches in Automatic Text Retrieval. Information Processing and Management **24**(5): 513-523.

Sanchez, R. P., E. P. Lorch, et al. (2001). Effects of Headings on Text Processing Strategies. Comtemporary Educational Psychology **26**(3): 418-428.

Schank, R. C. (1982a). Dynamic Memory: A Theory of Reminding and Learning in Computers and People, Cambridge University Press.

Schank, R. C. (1982b). Reading and Understanding: Teaching Perspective Artificial Intelligence, Lawrence Erlbaum Associates Inc,US.

Schank, R. C. (1986). Explanation Patterns: Understanding Mechanically and Creatively. Hillsdale, NJ, Lawrence Erlbaum Associates Inc,US.

Schank, R. C. (1991). Tell Me a Story: A New Look at Real and Artificial Intelligence. New York, Simon and Schuster.

Schank, R. C. (1995). Tell Me a Story: Narrative and Artificial Intelligence, Northwestern University Press.

Schank, R. C. (1999). Dynamic Memory Revisited, Cambridge University Press.

Schank, R. C. and R. Abelson (1977). Scripts, Plans, Goals, and Understanding: Inquiry into Human Knowledge Structures (Artificial Intelligence). Hillsdale, NJ, Lawrence Earlbaum Associates Inc, US.

Schank, R. C. and C. Cleary (1995). Making Machines Creative. The Creative Cognition Approach. R. A. Finke, MIT Press**:** 229-247.

Schultz, L. (2005). Bloom's Taxonomy. Accessed on Semptember, 2006. http://www.odu.edu/educ/llschult/blooms_taxonomy.htm

Sebastiani, F. (2002). Machine learning in automated text categorization. ACM Computing Surveys (CSUR) **34**(1): 1 - 47.

Shneiderman, B. (1987). Designing the user interface : strategies for effective human-computer interaction. Reading, Mass., Addison-Wesley.

Simon, H. A. (1955). A behavioural model of rational choice. Quarterly Journal of Economics **69**: 99-118.

Stauffer, R. G. (1969). Directing Reading Maturity as a Cognitive Process. New York, Harper and Row.

Stevenson, M. (2002). Combining Disambiguation Techniques to Enrich an Ontology. In proceedings of the Workshop on Machine Learning and Natural Language Processing for Ontology Engineering, the 5th European Conference on Artificial Intelligence (ECAI-02). 2002

Stokes, N. (2004). Applications of Lexical Cohesion Analysis in the Topic Detection and Tracking Domain. Computer Science. Dublin, University College Dublin**:** 261.

Sun, Y. (2004). Discovering and Representing the Organizational and Narrative Structures of e-Books to Support Comprehension. In proceedings of the 27th annual international ACM SIGIR Conference on Research and Development in Information Retrieval, Sheffield, UK, ACM Press. 24th - 28th July, 2004

Sun, Y., D. J. Harper, et al. (2004). Design of an e-Book User Interface and Visualizations to Support Reading for Comprehension. In proceedings of the 27th annual international ACM SIGIR Conference on Research and Development in Information Retrieval, Sheffield, UK, ACM Press. 24th -28th July, 2004

Sun, Y., D. J. Harper, et al. (2005). Aiding Comprehension in Electronic Books Using Contextual Information. In proceedings of The 9th European Conference on Research and Advanced Technology for Digital Libraries (ECDL), Vienna, Austria, Springer-Verlag. 19-23 September, 2005

Tombaugh, J., A. Lickorish, et al. (1987). Multi-window displays for readers of lengthy texts. International Journal of Man-Machine Studies **26**(5): 597-615.

van Dijk, T. A. and W. Kintsch (1983). Strategies of Discourse Comprehension. New York, Academic Press.

Voorhees, E. M. and D. K. Harman (2005). TREC: Experiment and Evaluation in Information Retrieval (Digital Libraries & Electronic Publishing), The MIT Press.

Whittaker, S., J. Hirschberg, et al. (1999). SCAN: designing and evaluating user interfaces to support retrieval from speech archives. In proceedings of The 22nd annual international ACM SIGIR conference on Research and development in information retrieval, Berkeley, California, United States, ACM. 15-19 August, 1999

Wilson, R. and M. Landoni (2000). EBONI: Electronic Textbook Design Guidelines. Accessed on 12 April, 2004. http://ebooks.strath.ac.uk/eboni/guidelines/

Woodruff, A., R. Gossweiler, et al. (2000). Enhancing a digital book with a reading recommender. In proceedings of The Conference on Human Factors and Computing Systems, The Hague, The Netherlands, ACM Press. 2000

Wren, S., B. Litke, et al. (2000). The Cognitive Foundations of Learning To Read: A Framework. Accessed on October, 2005.

APPENDICES

Appendix 1 Story Board of User Interface Design

A1.1 Browsing the Organizational Structure

Readers need tool to rapidly browse the book content. We propose to visualize the table of contents in the top of the e-Book user interface. Interactive bars represent sections (books) and chapters of the book. The height of the bars indicates the length of that section or chapter. Choose a book will update the visualization from book-level view to chapter level view.

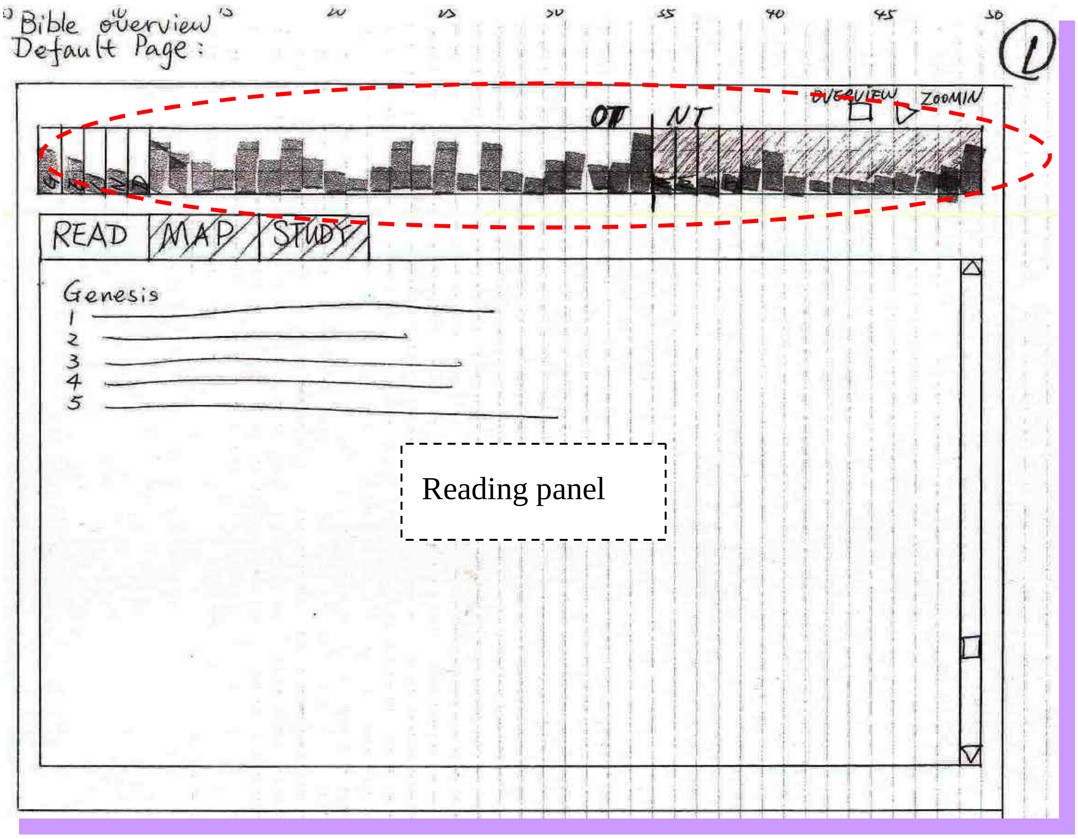


Chart 1 User Interface – Book View

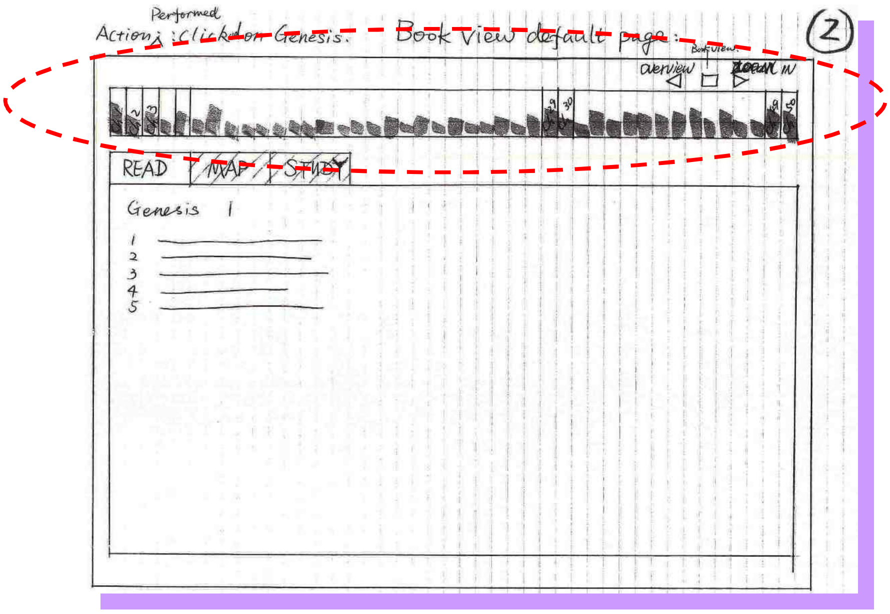


Chart 2 User Interface –Chapter View

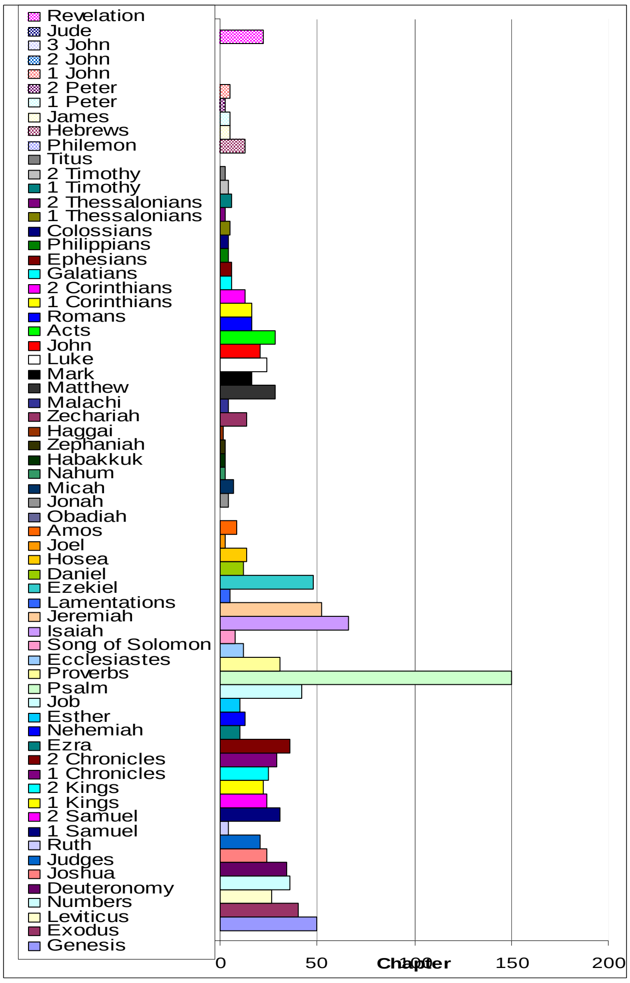


Chart 3 Visualization of the Length of Books

A1.2 Discovering Narrative Contexts based on Main Characters

In reading mood, tools that identify and highlight the key words and main characters could be helpful. See Chart 1 for the reading panel and Chart 6 for Character Map.

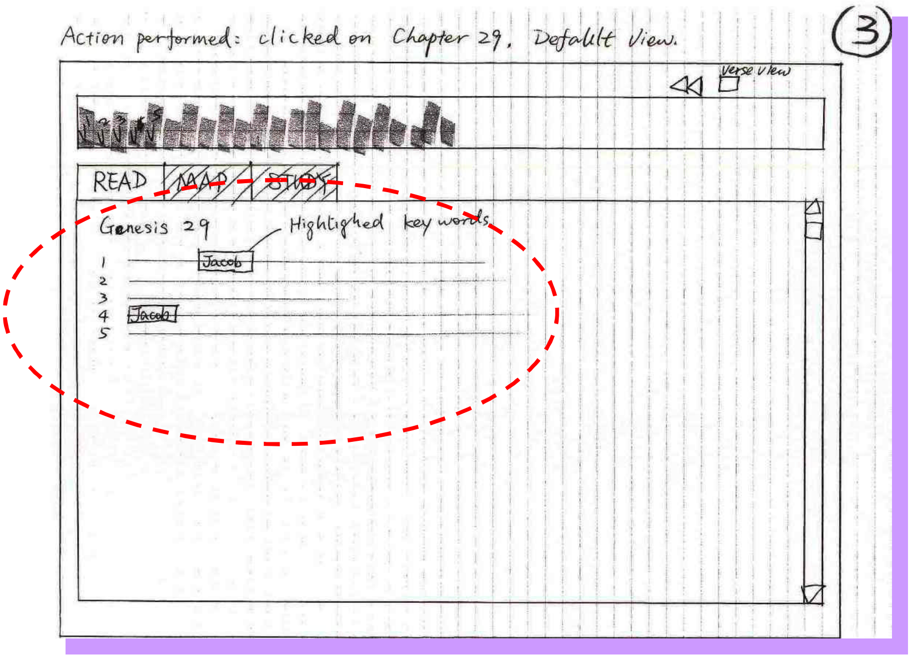


Chart 4 Highlighting of the Main Characters

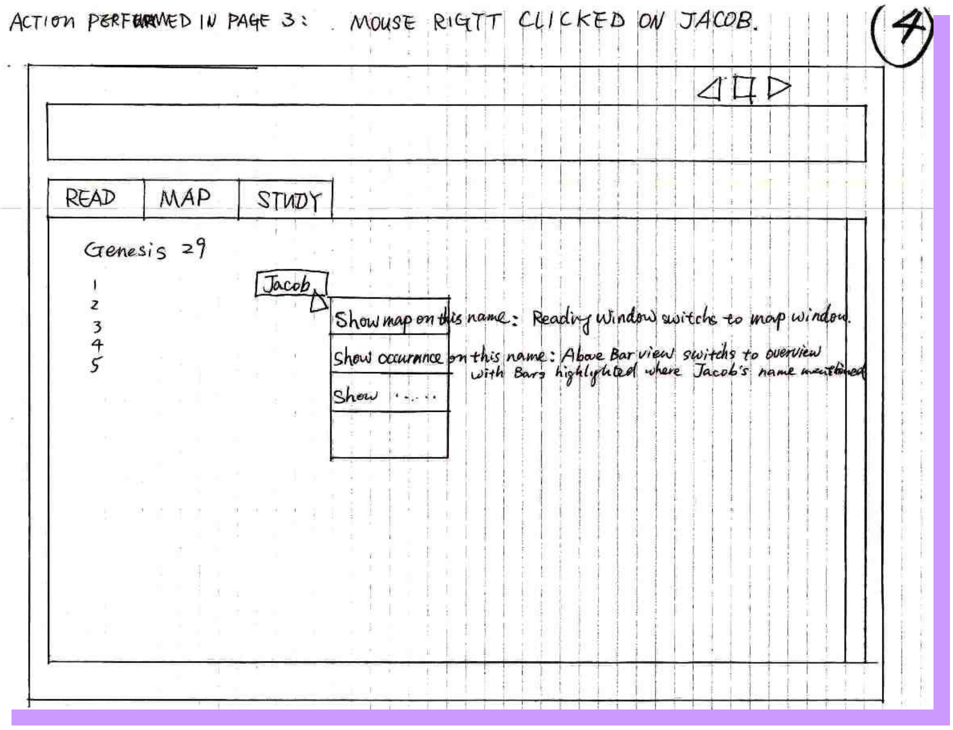

ACTION PERFORMED IN PAGE 3: MOUSE RIGTT CLICKED ON JACOB.
4
READ
MAP
STUDY
Genesis 29
1
2
3
4
5
Jacob
Show map on this name: Reading Window switchs to map window.
Show occurance on this name: Above Bar view switchs to overview with Bars highlighted where Jacob's name mentioned
Show . . . . .

Chart 5 Searching the Narrative Context by Main Characters

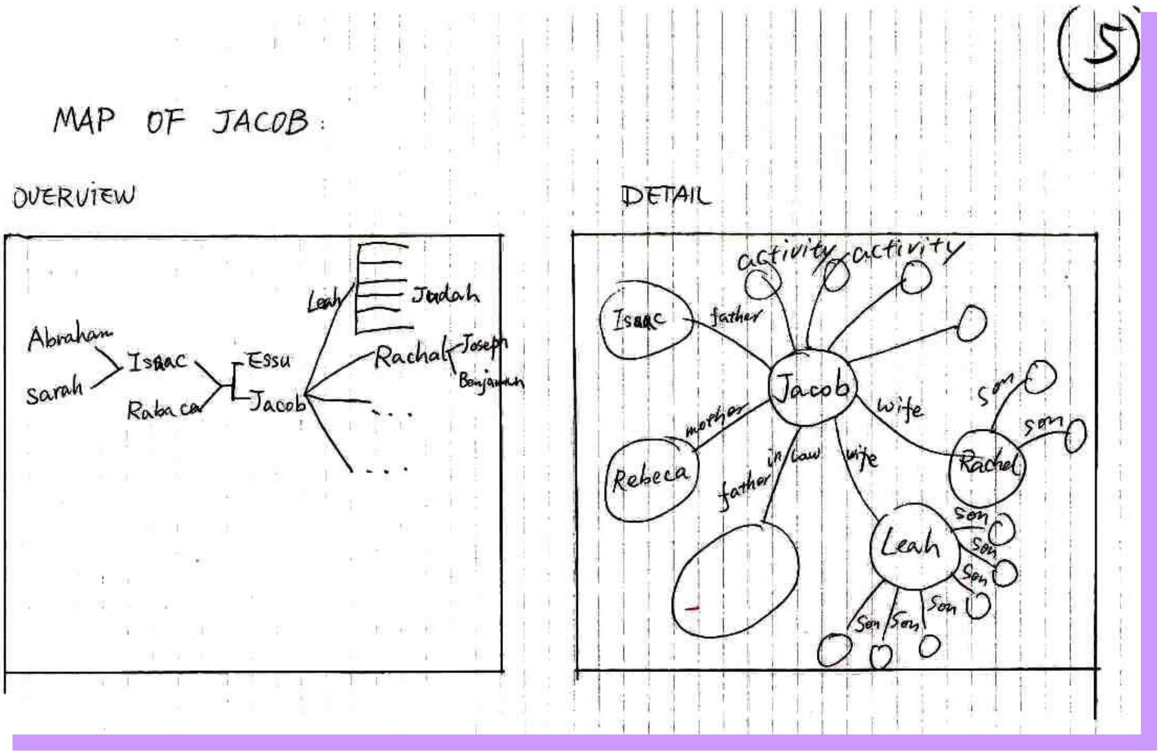


Chart 6 Browsing the Narrative Context by Main Characters

A1.3 Discovering Narrative Context based on Themes

Tools that identify and explore the thematical structure would be useful. Chart 8 shows the example of Theme Map in Overview and Detail mood.

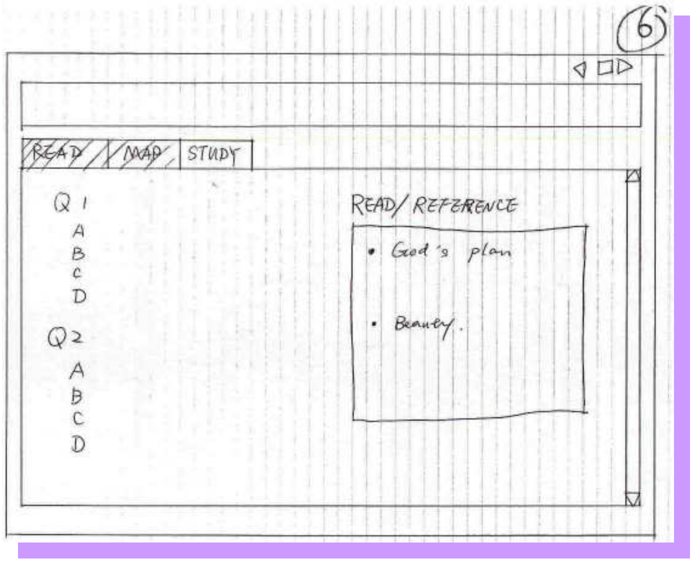


Chart 7 Assisting Reader with Reference and Study Notes

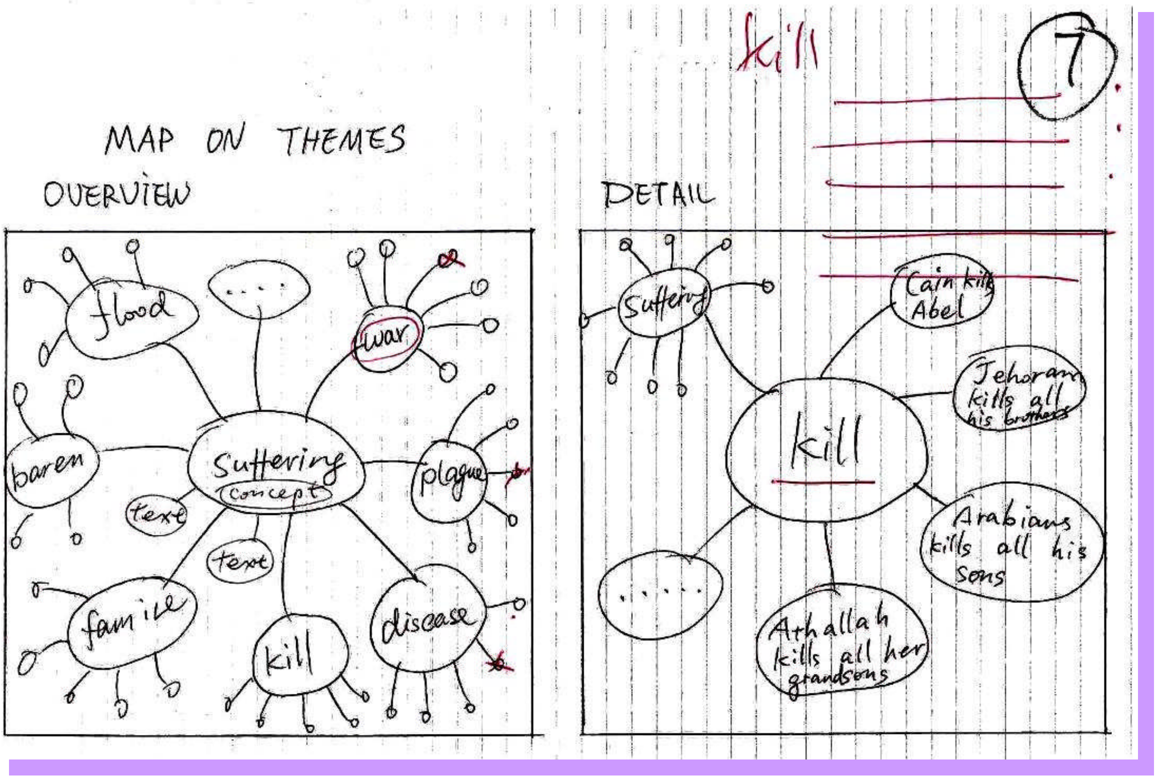


Chart 8 Assisting Readers with Theme Map

Appendix 2 Mle vs. TF in Cosine Distance Measure

Cosine Distance is a popular distance measure for comparing documents in the information retrieval literature. The concept of the Cosine Distance Measure is explained in (Salton and Buckley 1998). Hearst used it for scoring the text blocks in the TextTiling algorithm based on term frequencies (*tf*). In this study, we used a maximum likelihood estimate (*mle*), which is a normalized *tf,* when calculating the Cosine Distance Measure. Although the *mle* is effective when adapted to the symmetric divergence measure (5.3.4.2), it is powerless when used in the Cosine Distance measure. This is because, as shown below, the document length (*dl*) factors cancel each other.

The Cosine Distance Measure using *mle* is defined as:

$$Cos\ (W1, W2) = \frac{\sum_{i=1}^{n} p_i q_i}{\sqrt{\sum_{i=1}^{n} p_i^2 \sum_{i=1}^{n} q_i^2}} \qquad (1)$$

where $p_i = \frac{tf\,(t_i, W_1)}{dl\,(W_1)}$ and $q_i = \frac{tf\,(t_i, W_2)}{dl\,(W_2)}$.

Rewriting (1):

$$Cos\ (W1, W2) = \frac{\sum_{i=1}^{n} \left(\frac{tf\,(t_i, W_1)}{dl\,(W_1)} * \frac{tf\,(t_i, W_2)}{dl\,(W_2)} \right)}{\sqrt{\sum_{i=1}^{n} \left(\frac{tf\,(t_i, W_1)}{dl\,(W_1)} \right)^2 \sum_{i=1}^{n} \left(\frac{tf\,(t_i, W_2)}{dl\,(W_2)} \right)^2}} =$$

$$\frac{\dfrac{\sum_{i=1}^{n}(tf(t_i,W_1)*tf(t_i,W_2))}{dl(W_1)*dl(W_2)}}{\sqrt{\dfrac{\sum_{i=1}^{n}tf(t_i,W_1)^2}{dl(W_1)^2}*\dfrac{\sum_{i=1}^{n}tf(t_i,W_2)^2}{dl(W_2)^2}}}=$$

$$\frac{\sum_{i=1}^{n}(tf(t_i,W_1)*tf(t_i,W_2))}{\sqrt{\sum_{i=1}^{n}tf(t_i,W_1)^2\sum_{i=1}^{n}tf(t_i,W_2)^2}}$$

Therefore, for Cosine Distance Measure, using *mle* is the same as using *tf* alone.

Appendix 3 Questionnaire and Task Sheet (See 7.3.2 for Procedure)

A3.1 Entry Questionnaire (5 Minutes)

Name:

User ID:

Please circle your answer.

How much experience have you had…	None		Some		A lot
E1. reading The Holy Bible	1	2	3	4	5
E2. using references of the Bible (e.g., footnote, cross reference)?	1	2	3	4	5
E3. using Bible Study materials?	1	2	3	4	5
E4. reading and writing using computer?	1	2	3	4	5
E5. reading Bible on a computer or online?	1	2	3	4	5
E6. using Internet search engine (e.g., Google)?	1	2	3	4	5
When you read the Bible, how often do you focus on:	Never		Some -times		Often
E7A. Word(s)	1	2	3	4	5
E7B. Phrase	1	2	3	4	5
E7C. Verse	1	2	3	4	5
E7D. Character (e.g., Jacob)	1	2	3	4	5
E7E. Passage/Topic/Story	1	2	3	4	5
E7F. Theme	1	2	3	4	5
E7G. Reference	1	2	3	4	5

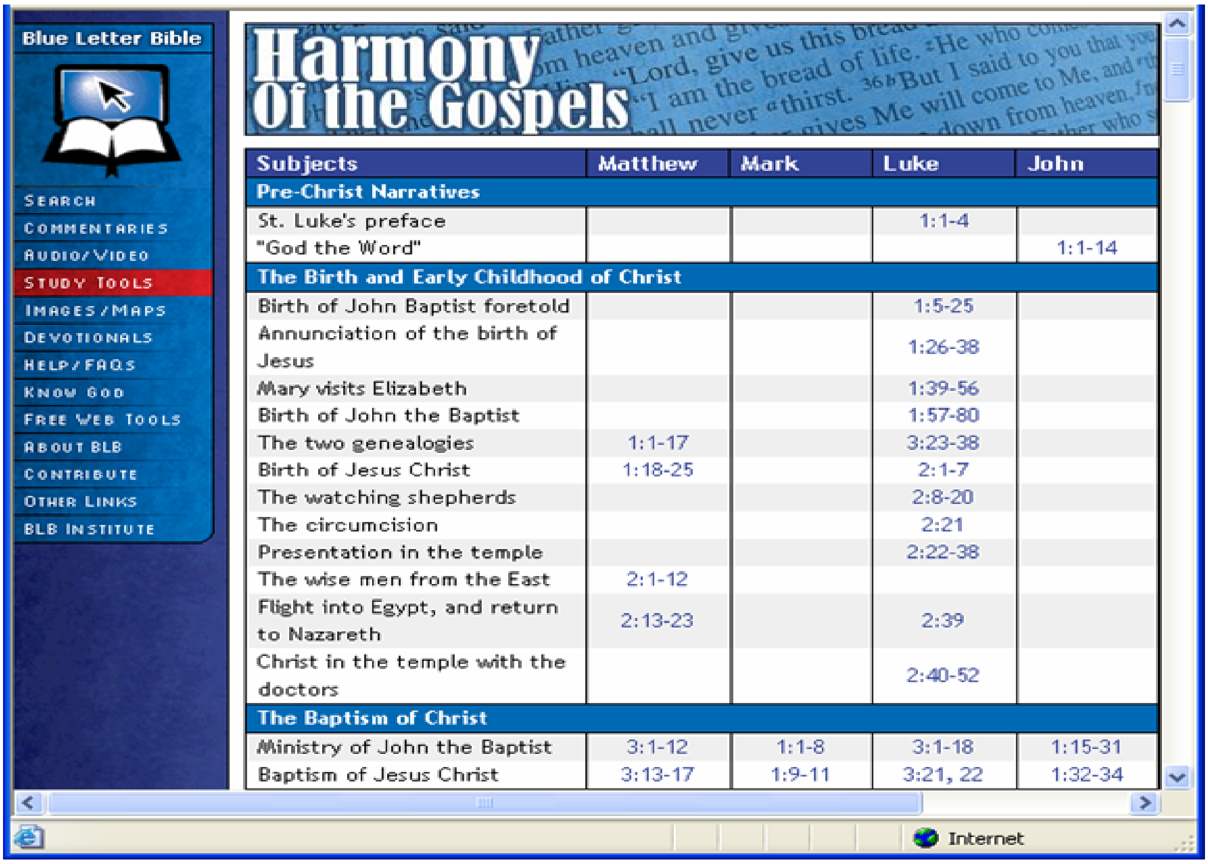
……………..Please wait for the instructions…………..

A3.2 First System Training

See Appendix 5 for training notes on both systems.

A3.3 Gospel Harmony Example

All four Gospel (Matthew, Mark, Luke, and John) in the New Testament of The Holy Bible tell the story of Jesus Christ. They were written to four separate audiences. Each Gospel narrative answers a different question about Jesus Christ, and, all together provides a clear portrait of Christ. A Harmony of the Gospels sorts all the events in the Gospels in a roughly chronological order, and presents the Gospel narratives of the same event (or topic) side by side. This Harmony combines the four Gospels into a single account of Christ's life on earth. Readers can easily compare the passages and thus view each Gospel writer's own perspective on the event (or topic). For example:



Subjects	Matthew	Mark	Luke	John
Pre-Christ Narratives				
St. Luke's preface			1:1-4	
"God the Word"				1:1-14
The Birth and Early Childhood of Christ				
Birth of John Baptist foretold			1:5-25	
Annunciation of the birth of Jesus			1:26-38	
Mary visits Elizabeth			1:39-56	
Birth of John the Baptist			1:57-80	
The two genealogies	1:1-17		3:23-38	
Birth of Jesus Christ	1:18-25		2:1-7	
The watching shepherds			2:8-20	
The circumcision			2:21	
Presentation in the temple			2:22-38	
The wise men from the East	2:1-12			
Flight into Egypt, and return to Nazareth	2:13-23		2:39	
Christ in the temple with the doctors			2:40-52	
The Baptism of Christ				
Ministry of John the Baptist	3:1-12	1:1-8	3:1-18	1:15-31
Baptism of Jesus Christ	3:13-17	1:9-11	3:21, 22	1:32-34

Chart 9 Gospel Harmony Table

Over the history, biblical scholars have to edit the Harmony of Gospels manually. Today we can use computer software to do this job. Using the given user interface, how could you find all the records of same event in the Gospels?

A3.4 Task A (5 Minutes)

Following the example, complete the portion of Gospel Harmony:

Hint*: read the given passage and*

- *use it as a query, and then search for relevant passages in the Gospels*
- *put the title of the event in the first column, and*
- *write down the chapter and verse numbers of the relevant passages in the column of each book accordingly.*

Click the Task A* Start *button on the small control panel.

Title	Matthew	Mark	Luke	John
E.g., The Demand for a Sign	16: 1-4	8: 10-13		
Q1.				6:1-15
Q2.		4:35-41		
Q3.	13:1-23			

Click the Task A* End *button on the small control panel.

A3.5 Post-Task A Questionnaire (2 Minutes)

Please circle your answer.

	Not at all		Some-what		Extremely
PTA1. Was it easy to do this portion of the Gospel Harmony?	1	2	3	4	5
PTA2. Are you satisfied with your performance?	1	2	3	4	5
PTA3. Did your previous knowledge help you with your answering?	1	2	3	4	5

……..Please turn to next page……..

A3.6 Story-Theme Map Example

The Holy Bible is composed of 66 separate works written by over 40 authors over a period of about two thousand years. It displays a remarkable harmony of thought, historical content, intent and expression. Like in the Gospel Harmony, many stories in the Bible are related, repeated and linked with each other, in order to reveal a theme. The following example illustrates how themes could be identified through a thread of relevant narratives.

Example: Jesus Raises a Widow's Son (Luke 7: 11-17)

Step 1: Choose the story or key words in the story as a query; look for similar stories using the given system.

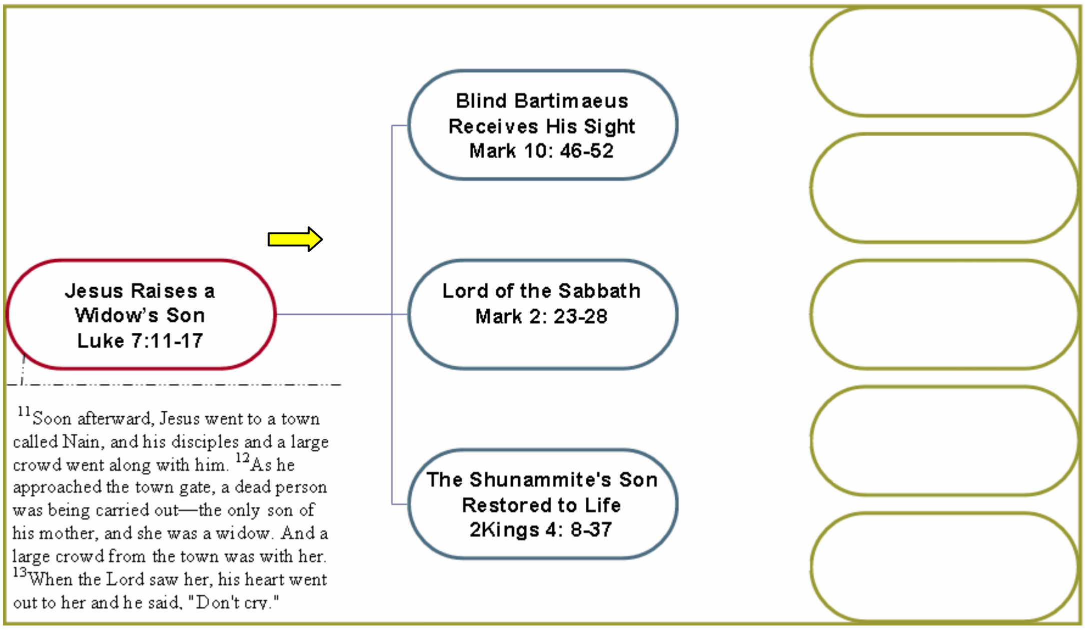

Jesus Raises a Widow's Son Luke 7:11-17
Blind Bartimaeus Receives His Sight Mark 10: 46-52
Lord of the Sabbath Mark 2: 23-28
The Shunammite's Son Restored to Life 2Kings 4: 8-37
[11]Soon afterward, Jesus went to a town called Nain, and his disciples and a large crowd went along with him. [12]As he approached the town gate, a dead person was being carried out—the only son of his mother, and she was a widow. And a large crowd from the town was with her. [13]When the Lord saw her, his heart went out to her and he said, "Don't cry."

Chart 10 First Step of Construction of a Story-Theme Map

Step 2: Use a relevant story as a query, look for similar stories of it.

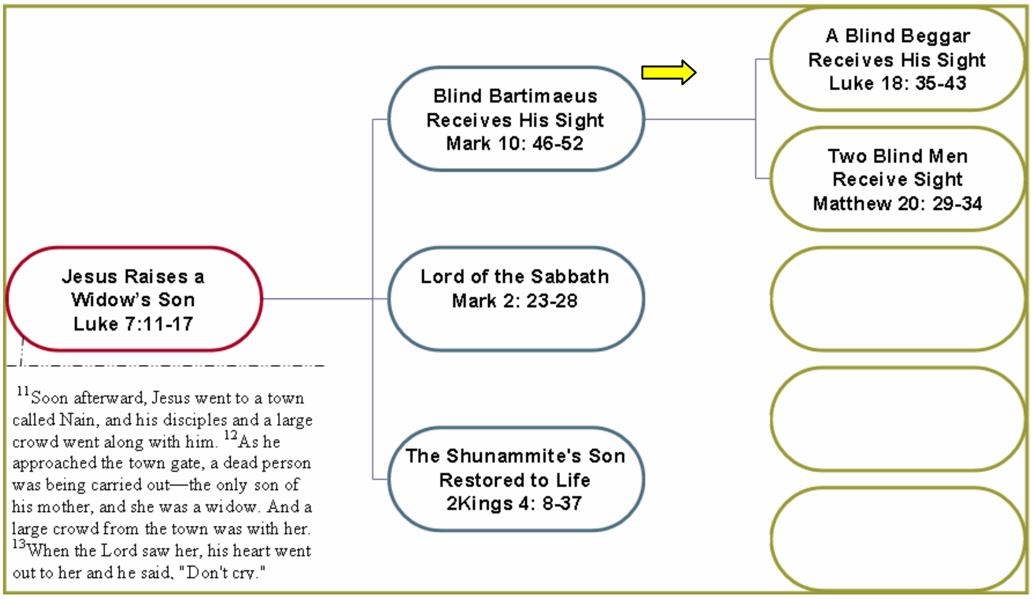

A Blind Beggar Receives His Sight Luke 18: 35-43
Blind Bartimaeus Receives His Sight Mark 10: 46-52
Two Blind Men Receive Sight Matthew 20: 29-34
Jesus Raises a Widow's Son Luke 7:11-17
Lord of the Sabbath Mark 2: 23-28
The Shunammite's Son Restored to Life 2Kings 4: 8-37
11Soon afterward, Jesus went to a town called Nain, and his disciples and a large crowd went along with him. 12As he approached the town gate, a dead person was being carried out—the only son of his mother, and she was a widow. And a large crowd from the town was with her. 13When the Lord saw her, his heart went out to her and he said, "Don't cry."

Chart 11 Second Step of Construction of a Story-Theme Map

Step 3: Generalize a theme with a thread of relevant stories.

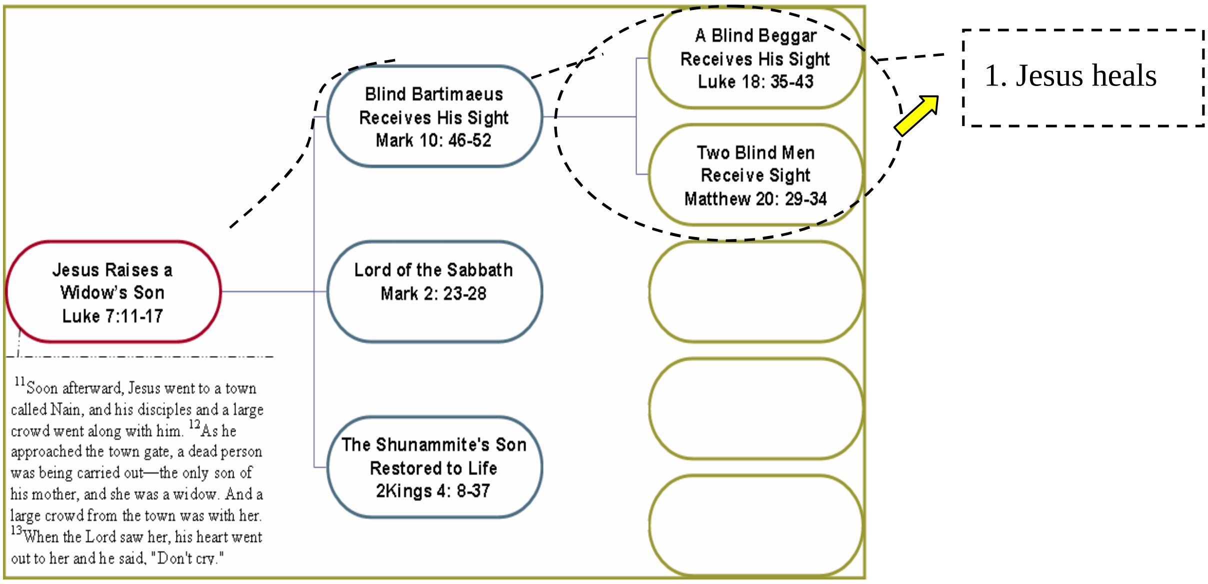
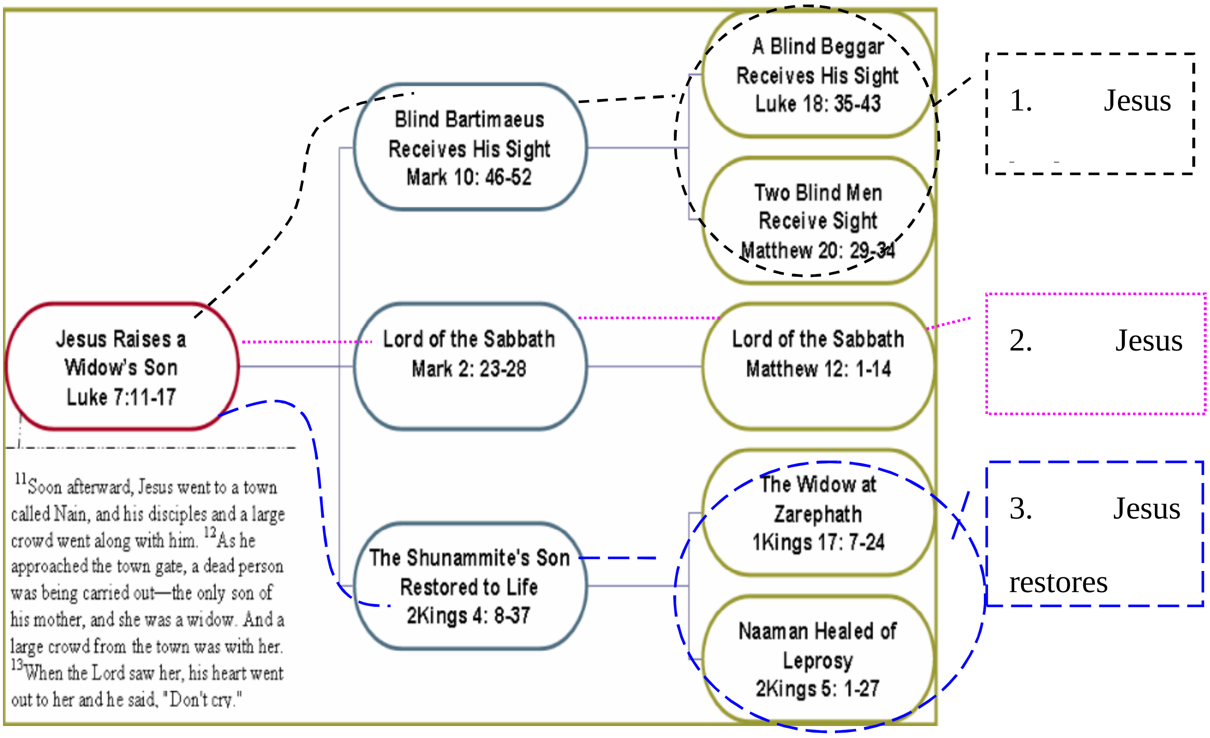

A Blind Beggar Receives His Sight Luke 18: 35-43
Blind Bartimaeus Receives His Sight Mark 10: 46-52
Two Blind Men Receive Sight Matthew 20: 29-34
1. Jesus heals
Jesus Raises a Widow's Son Luke 7:11-17
Lord of the Sabbath Mark 2: 23-28
The Shunammite's Son Restored to Life 2Kings 4: 8-37
11Soon afterward, Jesus went to a town called Nain, and his disciples and a large crowd went along with him. 12As he approached the town gate, a dead person was being carried out—the only son of his mother, and she was a widow. And a large crowd from the town was with her. 13When the Lord saw her, his heart went out to her and he said, "Don't cry."

Chart 12 Third Step of Construction of a Story-Theme Map

Repeat the steps 2 and 3, and finally, complete the entire story map.


Jesus Raises a Widow's Son Luke 7:11-17
Blind Bartimaeus Receives His Sight Mark 10: 46-52
A Blind Beggar Receives His Sight Luke 18: 35-43
Two Blind Men Receive Sight Matthew 20: 29-34
Lord of the Sabbath Mark 2: 23-28
Lord of the Sabbath Matthew 12: 1-14
The Shunammite's Son Restored to Life 2Kings 4: 8-37
The Widow at Zarephath 1Kings 17: 7-24
Naaman Healed of Leprosy 2Kings 5: 1-27
1. Jesus
2. Jesus
3. Jesus restores

[11]Soon afterward, Jesus went to a town called Nain, and his disciples and a large crowd went along with him. [12]As he approached the town gate, a dead person was being carried out—the only son of his mother, and she was a widow. And a large crowd from the town was with her. [13]When the Lord saw her, his heart went out to her and he said, "Don't cry."

Chart 13 Story-Theme Map Overview

……..After you understood this process, please turn to next page……..

A3.7 Task B (25/35 Minutes)

Following the “Link and Think” examples, read and scan relevant stories, complete a story map and identify the main themes.

***Hint**: use the given user interface,*

- *browse the passages; read the stories;*
- *find the relevant stories (you might need to choose query words);*
- *think for possible themes while you read; and*
- *complete the questions*

Click the Task B Start ***button on the small control panel.***

Open Acts chapter 17, read the story of **In Athens** from verse 16 to 34. Consider the questions Q4 - Q8:

Q4. Explain in your own words what did the Greeks believe?

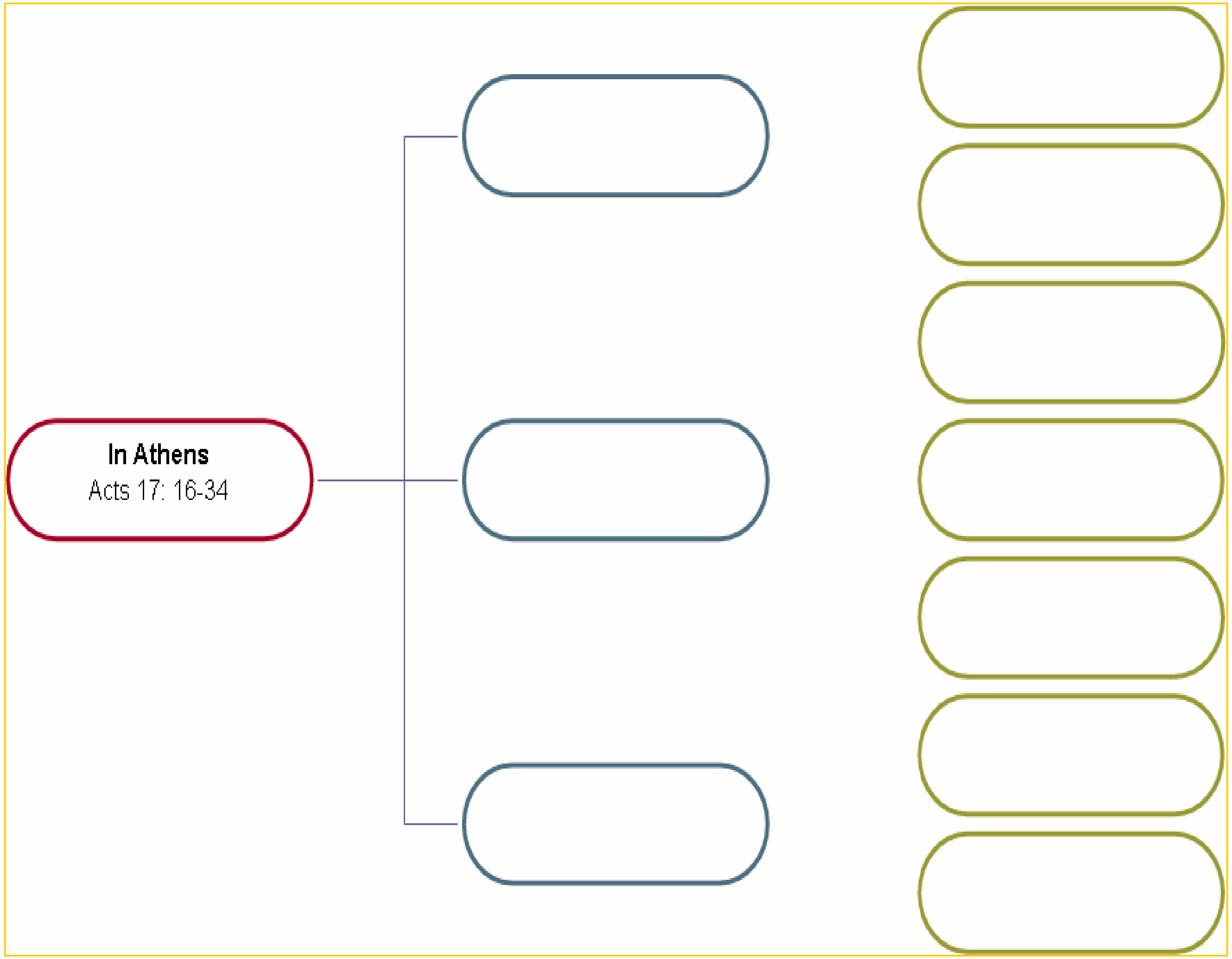
Q5. What is the core message in Paul’s speech?

Consider the following two questions:

Q6. **Use the given user interface,** can you complete the drawing of a relevant Story Map (left 2/3 of next page)? *Requirements: you must find at least one relevant story from the Old Testament.* If there are more relevant stories than the spaces, choose the most relevant ones according to your own judgement; if there are less relevant stories than the spaces, leave spaces blank.

Q7. Can you identify any themes with threads of narratives? Mark your answer on the Story Map (right 1/3 of next page).

……..Please turn to next page……..



Theme 1:

Theme 2:

Theme 3:

……..Please turn to next page……..

Q8. Now you have found "link(s)" for the story **In Athens**, and identified themes of threads of narratives. Do these links and themes help you to think more about the original story? If yes, please write down the **new** insights you have gained.

Click the Task B End ***button on the small control panel.***

A3.8 Post-Task B Questionnaire (3 Minutes)

Please circle your answer.

	Not at all		Some-what		Extremely
PTB1. Was it easy to do this reading task?	1	2	3	4	5
PTB2. Are you satisfied with your performance?	1	2	3	4	5
PTB3. Did your previous knowledge help you with your answering?	1	2	3	4	5

……..Please turn to next page……..

A3.9 Post-System Question (5 Minutes)

Please circle your answer.

	Not at all		Some-what		Extremely
PS1. How easy was it to learn to use this user interface?	1	2	3	4	5
PS2. How easy was it to use this user interface to find stories in the Bible?	1	2	3	4	5
PS3. How well did you understand how to use the user interface?	1	2	3	4	5
PS4. Did you enjoy using the system?	1	2	3	4	5
PS5. Did the system help you to complete the tasks?	1	2	3	4	5
PS6. Did the system help you to understand the stories?	1	2	3	4	5

PS7. Did you adopt any strategies when you used this user interface?

PS8. Please write down any other comments that you have about your reading and browsing experience with this e-Bible user interface here.

Close the user interface by clicking on the Close button Close on the small control panel.

……..Please wait for the instructions……..

A3.10 Break for 15 Minutes

Participants leave the lab for a cup of tea and some biscuits.

A3.11 Second System Training

See Appendix 5 for training notes on both systems.

A3.12 Task C (5 Minutes)

Following the example, complete the portion of Gospel Harmony:

***Hint**: read the given passage and*

- *use it as a query, and then search for relevant passages in the Gospels*
- *put the title of the event in the first column, and*
- *write down the chapter and verse numbers of the relevant passages in the column of each book accordingly.*

Click the Task C Start ***button on the small control panel.***

Title	Matthew	Mark	Luke	John
E.g., The Demand for a Sign	16: 1-4	8: 10-13		
Q1.	8: 5-13			
Q2.		11:27-33		
Q3.				20: 1-18

Click the Task C End ***button on the small control panel.***

A3.13 Post-Task C Questionnaire (2 Minutes)

Please circle your answer.

	Not at all		Some-what		Extremely
PTC1. Was it easy to do this portion of the Gospel Harmony?	1	2	3	4	5
PTC2. Are you satisfied with your performance?	1	2	3	4	5
PTC3. Did your previous knowledge help you with your answering?	1	2	3	4	5

……..Please turn to next page……..

A3.14 Task D (25/35 Minutes)

Following the "Link and Think" examples, read and scan relevant stories, complete a story map and identify the main themes.

Hint*: use the given user interface,*

- *browse the passages; read the stories;*
- *find the relevant stories (you might need to choose query words);*
- *think for possible themes while you read; and*
- *complete the questions*

Click the Task D ***button on the small control panel.***

Open Exodus chapter 3, read the story of **Moses and the Burning Bush**. Consider the questions Q4- Q8:

Q4. What were Moses' respons**es** when he was called?

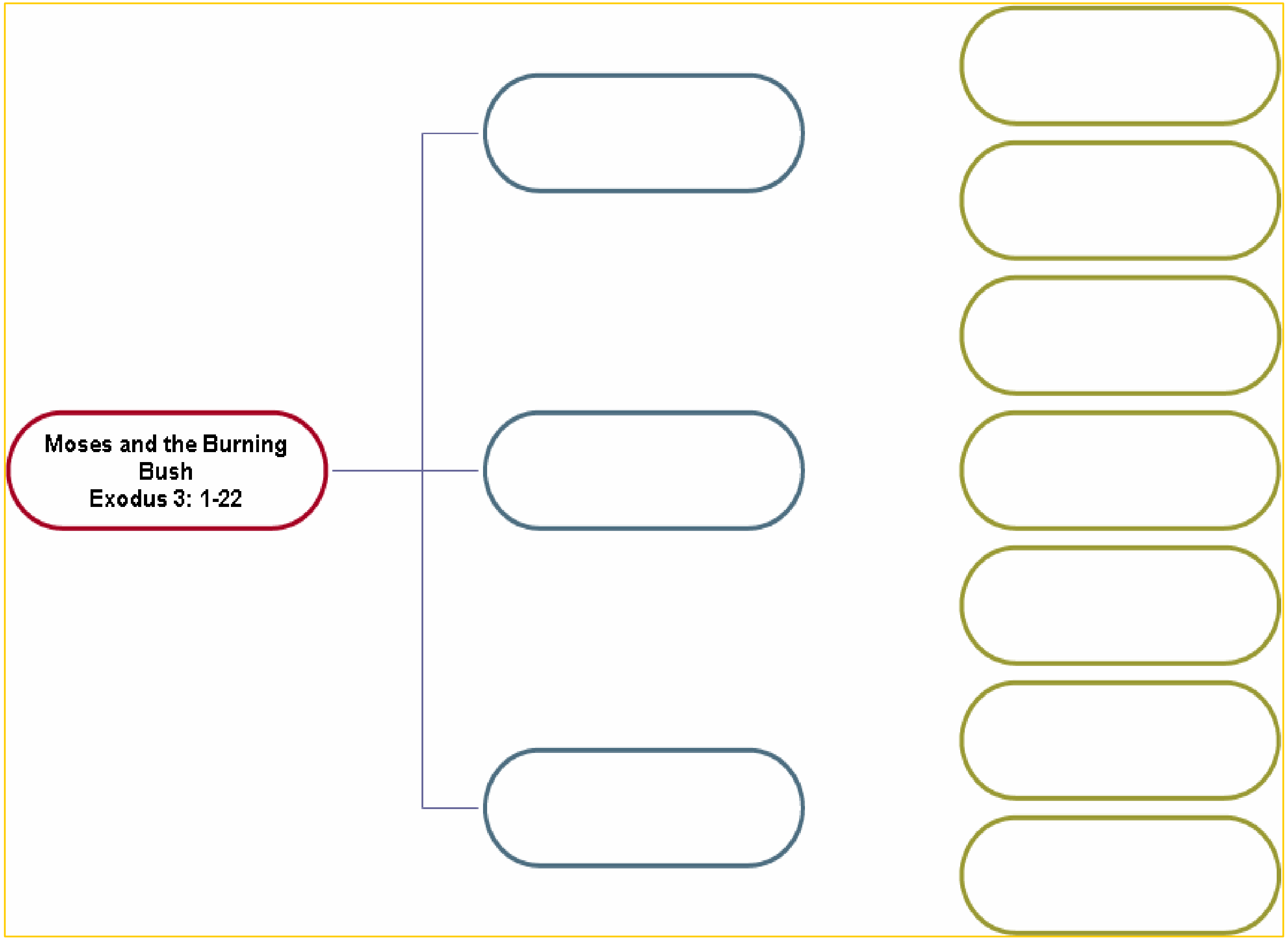
Q5. What is the core message in God's speech?

Consider the following two questions:

Q6. **Use the given user interface,** can you complete the drawing of a relevant Story Map (left 2/3 of next page)? *Requirements: you must find at least one relevant story from the New Testament.* If there are more relevant stories than the spaces, choose the most relevant ones according to your own judgement; if there are less relevant stories than the spaces, leave spaces blank.

Q7. Can you identify any themes with threads of narratives? Mark your answer on the Story Map (right 1/3 of next page).

……..Please turn to next page……..



Theme 1:

Theme 2:

Theme 3:

……..Please turn to next page……..

Q8. Now you have found “link(s)” for the story of **Moses and the Burning Bush**, and identified themes of threads of narratives. Do these links and themes help you to think more about the original story? If yes, please write down the new insights you have gained.

Click the Task D

button on the small control panel.

A3.15 Post-Task D Questionnaire (3 Minutes)

Please circle your answer.

	Not at all		Some-what		Extremely
PTD1. Was it easy to do this reading task?	1	2	3	4	5
PTD2. Are you satisfied with your performance?	1	2	3	4	5
PTD3. Did your previous knowledge help you with your answering?	1	2	3	4	5

……..Please turn to next page……..

A3.16 Post-System Questionnaire (5 Minutes)

This is same as section A3.9.

A3.17 Exit Questionnaire (5 Minutes)

Please consider the entire Bible reading experience that you just had with two user interfaces:

	Not at all		Some-what		Extremely
EX1. To what extend do you understand the reading tasks you performed?	1	2	3	4	5
EX2. How different from one another did you find the user interfaces?	1	2	3	4	5
	Dis-agree		Hard to say		Strongly Agree
EX3. Tasks in this study were similar to tasks that I normally do when reading my Bible	1	2	3	4	5
EX4. It was easy to navigate books and chapters in both user interfaces.	1	2	3	4	5
EX5. It was easy to read the selected passages	1	2	3	4	5
EX6. It was easy to track viewed pages.	1	2	3	4	5
EX7. It was easier searching for relevant stories with Blue than Orange, because there is a list of stories in the chapter.	1	2	3	4	5
EX8. It was easier searching for relevant stories with Orange than Blue, because there is a query search tool.	1	2	3	4	5

	Blue	No Difference	Orange
EX9. Which of the two systems did you find easier to learn to use?	1	2	3
EX10. Which of the two systems did you find easier to use?	1	2	3
EX11. Which of the two systems did you like the best overall?	1	2	3

EX12. Was there anything in particular you liked about user interface Orange and Blue?

Orange: ____

Blue: ____

EX13. Was there anything in particular you disliked about user interface Blue and Orange?

Blue: ____

Orange: ____

EX14. Please list any other comments that you have about your overall reading experience.

……..Thank you!……..

Appendix 4 Coding Scheme of Story-Theme Map

Identify the stories linked to each theme; read all the stories on the thread, and make a judgement of a relevance score for the theme.

A relevant score is a number between 0 and 3, where

R0, irrelevant;

R1, weak relevance; it is either an incomplete theme or a theme that covers only one story;

R2, mostly relevant; it is a theme that covers most of the stories;

R3, highly relevant; it is a precise theme that covers all the stories.

For example:

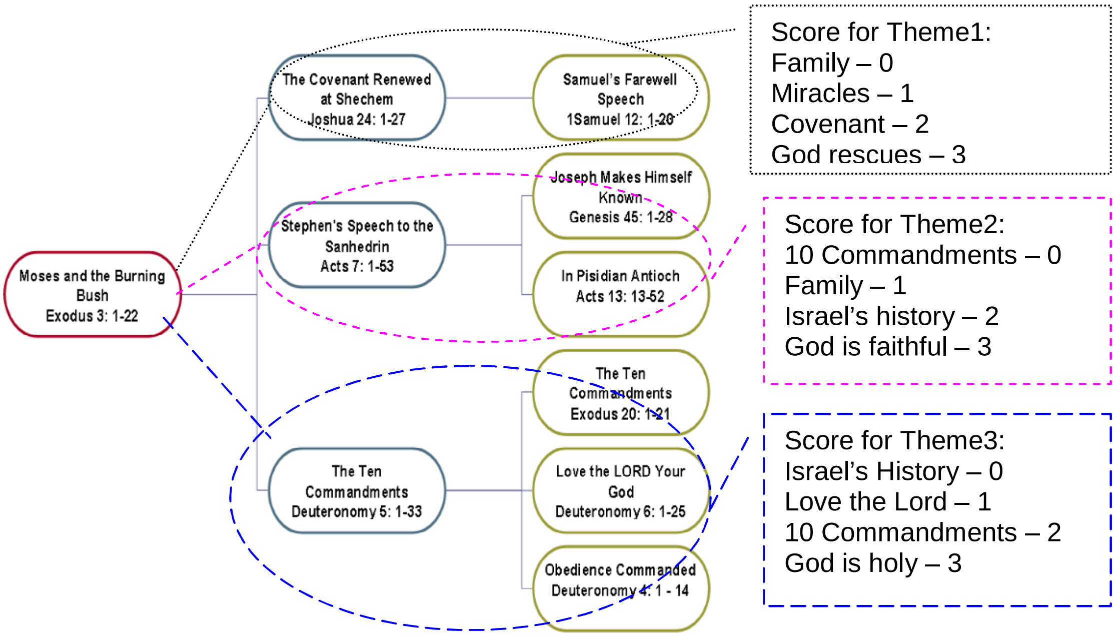

Moses and the Burning Bush Exodus 3: 1-22
The Covenant Renewed at Shechem Joshua 24: 1-27
Samuel's Farewell Speech 1Samuel 12: 1-26
Stephen's Speech to the Sanhedrin Acts 7: 1-53
Joseph Makes Himself Known Genesis 45: 1-28
In Pisidian Antioch Acts 13: 13-52
The Ten Commandments Deuteronomy 5: 1-33
The Ten Commandments Exodus 20: 1-21
Love the LORD Your God Deuteronomy 6: 1-25
Obedience Commanded Deuteronomy 4: 1 - 14
Score for Theme1:
Family – 0
Miracles – 1
Covenant – 2
God rescues – 3
Score for Theme2:
10 Commandments – 0
Family – 1
Israel's history – 2
God is faithful – 3
Score for Theme3:
Israel's History – 0
Love the Lord – 1
10 Commandments – 2
God is holy – 3

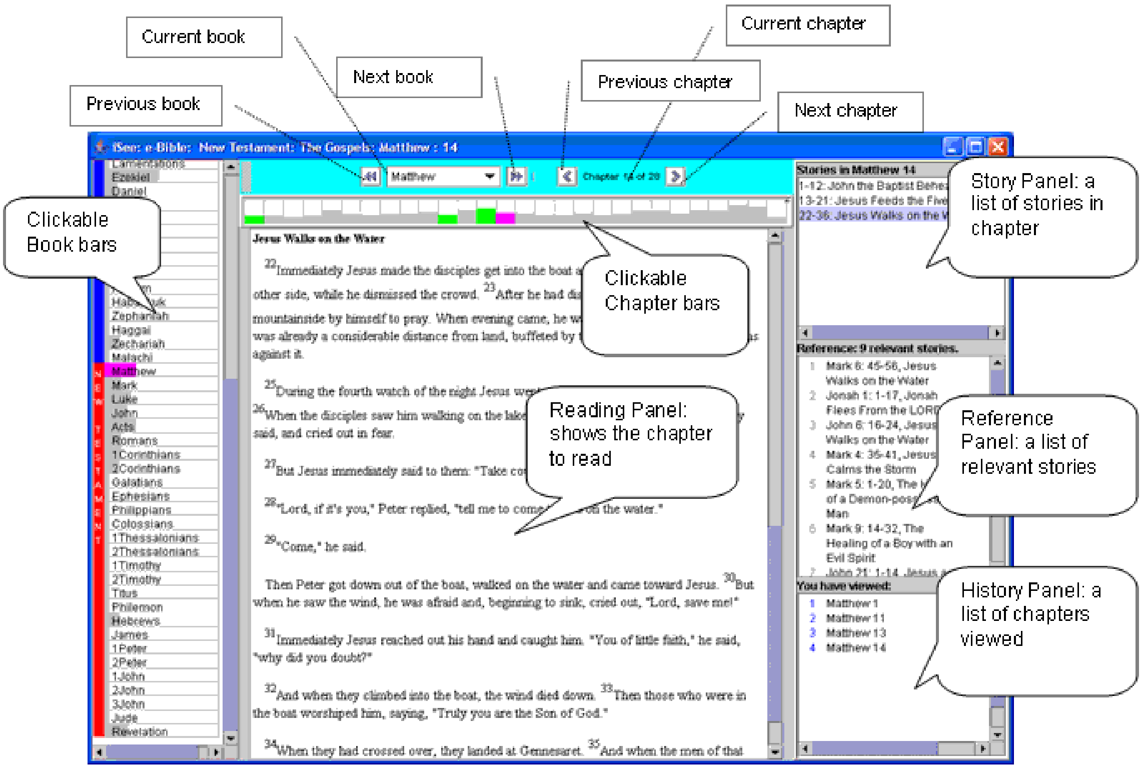
Chart 14 Example of a Story–Theme Map with Scores for each Theme Identified

The first thread of stories includes Moses and the Burning Bush, The Covenant Renewed at Shechem, and Samuel's Farwell Speech. There may be various themes identified of it. For example, if a reader put theme "Family" for these stories, she would get score 0, because it is irrelevant; theme “Miracles” would get score 1 as it is weak relevant (relevant to one story only); theme “covenant” will get score 2 as it is mostly relevant; theme “God rescues” will get score 3 as it is highly relevant and repeated in the three stories.

Appendix 5 Training Notes for iSee (Blue) and iSearch (Orange)

A5.1 User Manual and Training Task of iSee (Blue)

Click the Training Start *button on the small control panel.*


Current book
Next book
Previous book
Current chapter
Previous chapter
Next chapter
Clickable Book bars
Clickable Chapter bars
Reading Panel: shows the chapter to read
Story Panel: a list of stories in chapter
Reference Panel: a list of relevant stories
History Panel: a list of chapters viewed
iSee: e-Bible: New Testament: The Gospels: Matthew : 14
Matthew
Chapter 14 of 28
Stories in Matthew 14
Reference: 9 relevant stories.
You have viewed:

Jesus Walks on the Water

[22]Immediately Jesus made the disciples get into the boat a

other side, while he dismissed the crowd. [23]After he had dis

mountainside by himself to pray. When evening came, he w

was already a considerable distance from land, buffeted by t

against it.

[25]During the fourth watch of the night Jesus wen

[26]When the disciples saw him walking on the lake

said, and cried out in fear.

[27]But Jesus immediately said to them: "Take co

[28]"Lord, if it's you," Peter replied, "tell me to come on the water."

[29]"Come," he said.

Then Peter got down out of the boat, walked on the water and came toward Jesus. [30]But when he saw the wind, he was afraid and, beginning to sink, cried out, "Lord, save me!"

[31]Immediately Jesus reached out his hand and caught him. "You of little faith," he said, "why did you doubt?"

[32]And when they climbed into the boat, the wind died down. [33]Then those who were in the boat worshiped him, saying, "Truly you are the Son of God."

[34]When they had crossed over, they landed at Gennesaret. [35]And when the men of that

1-12: John the Baptist Behe
13-21: Jesus Feeds the Five
22-36: Jesus Walks on the W

1 Mark 6: 45-56, Jesus Walks on the Water
2 Jonah 1: 1-17, Jonah Flees From the LORD
3 John 6: 16-24, Jesus Walks on the Water
4 Mark 4: 35-41, Jesus Calms the Storm
5 Mark 5: 1-20, The H of a Demon-poss Man
6 Mark 9: 14-32, The Healing of a Boy with an Evil Spirit
7 John 21: 1-14, Jesus

1 Matthew 1
2 Matthew 11
3 Matthew 13
4 Matthew 14

Chart 15 User Manuel of iSee (Blue)

1. Go to Matthew chapter 13 by clicking the *Book bars* and *Chapter bars* accordingly.

2. Go to the next chapter using the "*next chapter*" ⟩ button.

3. Scroll down the *Reading panel* to verse 22-36 and read story "Jesus Walks on the Water".

4. Move your mouse to the *Story panel*; double click on the story title "Jesus Walks on the Water".

5. Look down the *Reference panel*; double click on one of the relevant stories.

6. The *Reading panel* is updated with a new chapter where the story is located.

7. Fill the blanks by the relevant stories you have found:

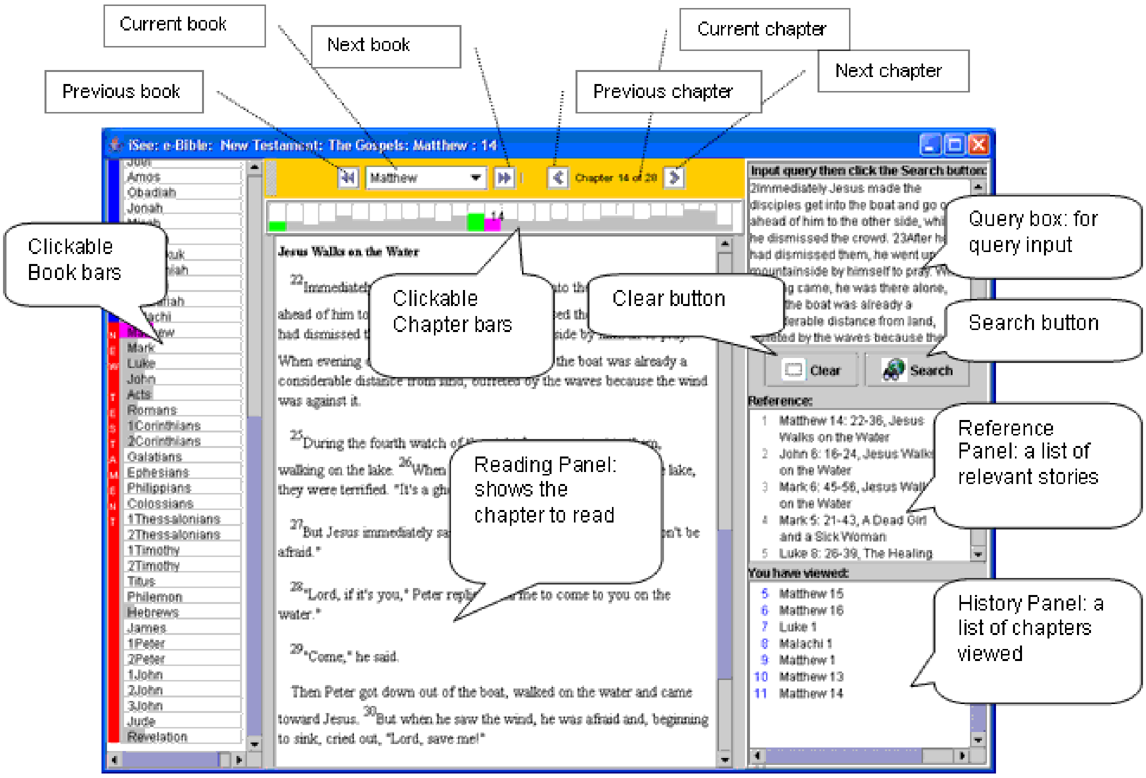
Title	Matthew	Mark	Luke	John
Jesus Walks on the Water	14:22-36			

Click the Training End *button on the small control panel.*

You will be ready to start two reading tasks using the interface Blue.

A5.2 User Manual and Training Task of iSearch (Orange)

***Click** the Training* Start *button on the small control panel.*



Jesus Walks on the Water

[22]Immediately … ahead of him to … had dismissed t… When evening … the boat was already a considerable distance from land, buffeted by the waves because the wind was against it.

[25]During the fourth watch of … walking on the lake. [26]When … lake, they were terrified. "It's a gh…

[27]But Jesus immediately sa… n't be afraid."

[28]"Lord, if it's you," Peter repli… me to come to you on the water."

[29]"Come," he said.

Then Peter got down out of the boat, walked on the water and came toward Jesus. [30]But when he saw the wind, he was afraid and, beginning to sink, cried out, "Lord, save me!"

Input query then click the Search button:

2Immediately Jesus made the disciples get into the boat and go … ahead of him to the other side, whi… he dismissed the crowd. 23After he had dismissed them, he went up … mountainside by himself to pray. … g came, he was there alone, … the boat was already a … derable distance from land, … ted by the waves because the

Reference:

1 Matthew 14: 22-36, Jesus Walks on the Water
2 John 6: 16-24, Jesus Walks on the Water
3 Mark 6: 45-56, Jesus Walk… on the Water
4 Mark 5: 21-43, A Dead Girl and a Sick Woman
5 Luke 8: 26-39, The Healing

You have viewed:

5 Matthew 15
6 Matthew 16
7 Luke 1
8 Malachi 1
9 Matthew 1
10 Matthew 13
11 Matthew 14

Chart 16 User Manuel of iSearch (Orange)

1. Go to Matthew chapter 13 by clicking the *Book bars* and *Chapter bars* accordingly.

2. Go to the next chapter using the "*next chapter*" > button.

3. Scroll down the *Reading panel* to verse 22-36 and read story "Jesus Walks on the Water".

4. Highlight the stories from the *Reading panel* (press down the left button on mouse, then drag over the required text); use Ctrl+C and Ctrl+V to copy and paste the text into the *Query box*.

5. Click on the "*Search*" button Search.

6. Look down the *Reference panel*; double click on one of the relevant stories.

7. The *Reading panel* is updated with a new chapter where the story is located.

8. Fill the blanks in the following table by the relevant stories you have found:

Title	Matthew	Mark	Luke	John
Jesus Walks on the Water	14:22-36			

Click the Training End *button on the small control panel.*

You will be ready to start two reading tasks using the interface Orange.