

國立交通大學

資訊科學與工程研究所

博士論文

摘錄式多文件自動化摘要方法之研究

A Study on Extraction-based
Multidocument Summarization

研究生：葉鎮源

指導教授：楊維邦 博士
柯皓仁 博士

中華民國九十七年三月

摘錄式多文件自動化摘要方法之研究

**A Study on Extraction-based
Multidocument Summarization**

研究生：葉鎮源

Student: Jen-Yuan Yeh

指導教授：楊維邦 博士
柯皓仁 博士

Advisors: Dr. Wei-Pang Yang
Dr. Hao-Ren Ke

國立交通大學
資訊科學與工程研究所
博士論文



Submitted to Institute of Computer Science and Engineering

College of Computer Science

National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy

in

Computer Science and Engineering

March 2008

Hsinchu, Taiwan, Republic of China

中華民國九十七年三月

摘錄式多文件自動化摘要方法之研究

研究生：葉鎮源

指導教授：楊維邦 博士
柯皓仁 博士

國立交通大學 資訊科學與工程研究所

摘 要

隨著資訊科技的快速發展，線上資訊的量及其可得性已大幅地增長。資訊爆炸導致產生資訊超載的現象，如何有效率地取得且有效地利用所需的資訊，已儼然成為人們生活中必須面對的迫切問題。文件自動化摘要(Text Summarization)技術由電腦分析文件內容，擷取出重要的資訊，並以摘要的形式呈現。此技術可以幫助人們處理資訊，於短時間內了解文件的內容，以作為決策的參考。

本論文探討多文件自動化摘要的方法，研究主題包含：(1) 多文件摘要(Multidocument Summarization)與(2) 以查詢為導向之多文件摘要(Query-focused Multidocument Summarization)。多文件摘要乃是從多篇主題相關的文件中產生單篇摘要；以查詢為導向之多文件摘要則是從多篇主題相關的文件中擷取與使用者興趣相關的內容，並依此產生單篇摘要。本論文採用語句摘錄(Sentence Extraction)的方法，判別語句的重要性，並逐字摘錄重要的語句以產生摘錄式摘要。其中，本論文的重點為語句重要性的計量及語句排序方法的研究。

針對多文件摘要，本論文提出一套以圖形為基礎的語句排序(Sentence Ranking)方法：iSpreadRank。此方法建構語句關係網路(Sentence Similarity Network)作為分析多文件的模型，並採用擴散激發理論(Spreading Activation)推導語句的重要性作為排序的依據。接著，依序挑選重要的語句以形成摘要；挑選語句時，以與先前被挑選的語句具較低資訊重複者為優先。實驗中，將此摘要方法

應用於 DUC 2004 的資料集。評估結果顯示，相較於 DUC 2004 當年度競賽的系統，本論文所提出的方法於 ROUGE 基準上有良好的表現。

針對以查詢為導向之多文件摘要，本論文結合：(1) 語句與查詢主題的相似度與(2) 語句的資訊代表性，提出一套語句重要性的計量方法。其中，利用潛在語意分析(Latent Semantic Analysis)，以計算語句與查詢主題於語意空間的相似度；並採用傳統摘要方法中探討語句代表性的特徵(Surface-level Features)，以評量語句的資訊代表性。本論文亦基於 Maximum Marginal Relevance 技術，考量資訊的重複性，提出一個適用於以查詢為導向之多文件摘要的語句摘錄方法。實驗中，將此摘要方法應用於 DUC 2005 的資料集。評估結果顯示，相較於 DUC 2005 當年度競賽的系統，本論文所提出的方法於 ROUGE 基準上有良好的表現。

關鍵詞：多文件摘要；一般性摘要；以查詢為導向之摘要；語句排序；語句摘錄；重複性資訊過濾；



A Study on Extraction-based Multidocument Summarization

Student: Jen-Yuan Yeh

Advisors: Dr. Wei-Pang Yang
Dr. Hao-Ren Ke

Institute of Computer Science and Engineering,
National Chiao Tung University

ABSTRACT

The rapid development of information technology over the past decades has dramatically increased the amount and the availability of online information. The explosion of information has led to information overload, implying that finding and using the information that people really need efficiently and effectively has become a pressing practical problem in people's daily life. Text summarization, which can automatically digest information content from document(s) while preserving the underlying main points, is one obvious technique to help people interact with information.

This thesis discusses work on summarization, including: (1) multidocument summarization, and (2) query-focused multidocument summarization. The first is to produce a generic summary of a set of topically-related documents. The second, a particular task of the first, is to generate a query-focused summary, which reflects particular points that are relevant to the user's desired topic(s) of interest. Both tasks are addressed using the most common technique for summarization, namely sentence extraction: important sentences are identified and extracted verbatim from documents and composed into an extractive summary. The first step towards sentence extraction is obviously to score and rank sentences in order of importance, which is the major focus of this thesis.

In the first task, a novel graph-based sentence ranking method, iSpreadRank, is proposed to rank sentences according to their likelihood of being part of the summary. The input documents are modeled as a sentence similarity network. iSpreadRank practically applies spreading activation to reason the relative importance of sentences based on the network structure. It then iteratively extracts one sentence at a time into the summary, which not only has high importance but also has low redundancy with the sentences extracted prior to it. The proposed summarization method is evaluated using the DUC 2004 data set and found to perform well in various ROUGE measures. Experimental results show that the proposed method is competitive to the top systems at DUC 2004.

In the second task, a new scoring method, which combines (1) the degree of relevance of a sentence to the query, and (2) the informativeness of a sentence, is proposed to measure the likelihood of sentences of being part in the summary. While the degree of query relevance of a sentence is assessed as the similarity between the sentence and the query computed in a latent semantic space, the informativeness of a sentence is estimated using surface-level features. Moreover, a novel sentence extraction method, inspired by maximal marginal relevance (MMR), is developed to iteratively extract one sentence at a time into the summary, if it is not too similar to any sentences already extracted. The proposed summarization method is evaluated using the DUC 2005 data set and found to perform well in various ROUGE measures. Experimental results show that the proposed method is competitive to the top systems at DUC 2005.

Keywords: multidocument summarization; generic summary; query-focused summary; sentence ranking; sentence extraction; redundancy filtering;

Contents

摘 要	I
ABSTRACT.....	III
Contents	V
List of Figures.....	VIII
List of Tables.....	XI
致 謝	XII
Chapter 1 Introduction.....	1
1.1 Background.....	3
1.1.1 History of text summarization	4
1.1.2 Summarization factors	6
1.1.3 Summarization techniques	8
1.1.4 Summary Evaluation.....	12
1.2 Tasks and Challenges	13
1.2.1 Multidocument summarization	13
1.2.2 Query-focused multidocument summarization.....	17
1.3 Guide to Remaining Chapters.....	20
Chapter 2 Literature Survey	21
2.1 Multidocument Summarization	21
2.2 Query-focused Multidocument Summarization.....	25
2.3 Related Research Projects.....	28
2.3.1 PERSIVAL	28
2.3.2 NewsBlaster	29

2.3.3	MEAD.....	29
2.3.4	GLEANS.....	30
2.3.5	NeATS.....	30
2.3.6	GISTexter.....	31
Chapter 3 Multidocument Summarization.....		32
3.1	Design.....	34
3.2	Algorithm.....	37
3.2.1	Text as a graph: sentence similarity network.....	37
3.2.2	Feature extraction.....	39
3.2.3	Ranking the importance of sentences	41
3.2.4	Sentence extraction.....	49
3.2.5	Sentence ordering.....	50
3.3	Evaluation.....	51
3.3.1	Data set and experimental setup	51
3.3.2	Evaluation method and metric.....	51
3.3.3	Results.....	53
3.3.4	Example output.....	58
3.4	Discussion.....	63
3.4.1	Sentence similarity network.....	63
3.4.2	The use of sentence-specific features	63
3.4.3	iSpreadRank.....	64
3.4.4	The proposed summarization approach.....	66
Chapter 4 Query-focused Multidocument Summarization.....		68
4.1	Design.....	70
4.2	Algorithm.....	73

4.2.1	Relevance between a sentence and the query	74
4.2.2	Feature extraction.....	78
4.2.3	Sentence scoring	81
4.2.4	Sentence extraction	83
4.2.5	Sentence ordering.....	84
4.3	Evaluation	85
4.3.1	Date set and experimental setup	85
4.3.2	Evaluation method and metric	85
4.3.3	Results.....	87
4.3.4	Example output	90
4.4	Discussion	109
4.4.1	Query relevance analysis by latent semantic analysis	109
4.4.2	The use of sentence-specific features	110
4.4.3	MMR vs. Modified MMR.....	110
4.4.4	The proposed summarization method.....	111
Chapter 5 Conclusions.....		112
5.1	Multidocument Summarization Framework	112
5.2	Contributions.....	114
5.3	Future Work	117
Bibliography		121

List of Figures

1.1. Overview of the summarization process.....	3
3.1. The proposed multidocument summarization approach.....	34
3.2. A sentence similarity network.....	38
3.3. The collection and the spread of activations for node A in one iteration.....	45
3.4. An example to explain how iSpreadRank works.....	48
3.5. The computation of $X(1)$	48
3.6. The algorithm of sentence extraction.....	49
3.7. The algorithm of sentence ordering.....	50
3.8. ROUGE-1 scores of system and human peers at DUC 2004.....	56
3.9. ROUGE-2 scores of system and human peers at DUC 2004.....	56
3.10. ROUGE-3 scores of system and human peers at DUC 2004.....	57
3.11. ROUGE-4 scores of system and human peers at DUC 2004.....	57
3.12. ROUGE-L scores of system and human peers at DUC 2004.....	58
3.13. ROUGE-W-1.2 scores of system and human peers at DUC 2004.....	58
3.14. ROUGE-1 scores of With-iSpreadRank (C+P+SF) for 50 clusters.....	59
3.15. System summary for d30045t.....	59
3.16. Model summary, created by B, for d30045t.....	60
3.17. Model summary, created by C, for d30045t.....	60
3.18. Model summary, created by E, for d30045t.....	60
3.19. Model summary, created by F, for d30045t.....	61
3.20. System summary for d30027t.....	61
3.21. Model summary, created by A, for d30027t.....	62
3.22. Model summary, created by C, for d30027t.....	62
3.23. Model summary, created by E, for d30027t.....	62

3.24. Model summary, created by G, for d30027t.....	63
4.1. The proposed query-focused multidocument summarization approach	70
4.2. The process of sentence extraction using MMR.....	83
4.3. ROUGE-2 scores of system and human peers at DUC 2005.....	90
4.4. ROUGE-SU4 scores of system and human peers at DUC 2005	90
4.5. ROUGE-2 scores of M4 for 50 clusters.....	91
4.6. ROUGE-SU4 scores of M4 for 50 clusters.....	91
4.7. Query statement for d357i	92
4.8. System summary for d357i	93
4.9. Model summary, created by D, for d357i	93
4.10. Model summary, created by E, for d357i.....	94
4.11. Model summary, created by F, for d357i	94
4.12. Model summary, created by I, for d357i.....	95
4.13. Query statement for d694j	96
4.14. System summary for d694j	96
4.15. Model summary, created by G, for d694j.....	97
4.16. Model summary, created by H, for d694j	97
4.17. Model summary, created by I, for d694j.....	98
4.18. Model summary, created by J, for d694j.....	98
4.19. Query statement for d376e.....	99
4.20. System summary for d376e	100
4.21. Model summary, created by A, for d376e.....	100
4.22. Model summary, created by B, for d376e.....	101
4.23. Model summary, created by C, for d376e.....	101
4.24. Model summary, created by D, for d376e.....	102
4.25. Model summary, created by E, for d376e	102

4.26. Model summary, created by G, for d376e	103
4.27. Model summary, created by H, for d376e.....	103
4.28. Model summary, created by I, for d376e	104
4.29. Model summary, created by J, for d376e	104
4.30. Query statement for d436j	105
4.31. System summary for d436j	106
4.32. Model summary, created by G, for d436j.....	106
4.33. Model summary, created by H, for d436j	107
4.34. Model summary, created by I, for d436j.....	108
4.35. Model summary, created by J, for d436j.....	108
5.1. Proposed framework for extraction-based multidocument summarization	113



List of Tables

1.1. Examples that employ extractive techniques.....	11
1.2. Examples that employ abstractive techniques	11
1.3. Query statement for set d357i.....	18
1.4. Query statement for set d376e	18
3.1. The sentence-specific feature set	40
3.2. The inferred weights of S_i at different iterations.....	49
3.3. ROUGE runtime arguments for DUC 2004.....	52
3.4. ROUGE-1 scores obtained in different experimental settings.....	54
3.5. Part of the official ROUGE-1 scores of Task 2 at DUC 2004	55
4.1. ROUGE runtime arguments for DUC 2005.....	86
4.2. Settings of different models.....	87
4.3. ROUGE-2 scores obtained in different experimental settings.....	87
4.4. ROUGE-SU4 scores obtained in different experimental settings.....	87
4.5. Part of the official ROUGE-2 scores at DUC 2005	89
4.6. Part of the official ROUGE-SU4 scores at DUC 2005	89

致 謝

本論文得以順利完成，首先要感謝指導教授楊維邦博士及柯皓仁博士。兩位教授於我的求學過程，歷經碩士及博士兩個階段，多年來一直悉心且不辭辛勞地引領、指導與鼓勵。恩師們的教導，啟發我對研究的興趣，使我得以一窺高深學術的殿堂，並走上研究之路。亦是恩師們的支持，我才能夠申請到出國訪問研究的機會。於國外學習的經驗，不僅開闊我的視野，也使得我的人生經歷更加豐富。此外，恩師們的學者風範以及待人處世的態度，日後將是我努力追隨的模範。

感謝本校孫春在教授、袁賢銘教授、梁婷教授，以及台灣大學項潔教授、清華大學金陽和教授與唐傳義教授、成功大學曾新穆教授。你們的不吝指教，給予我在研究方向及論文內容的諸多寶貴建議與指正。同時，感謝 Columbia University 的 Kathleen R. McKeown 教授與 Owen Rambow 博士，謝謝你們於我在美國進行訪問研究時的照顧與指導。

感謝實驗室的夥伴們，由於你們對我的關懷與照顧，讓我的研究生活不感孤單。每次的討論與腦力激盪，每每提供我不同的意見，使我獲得許多新的想法。同時，也由衷地感激過去所有曾經陪伴、幫助與鼓勵我的朋友們。

最後，感謝我親愛的父母與家人們。謝謝你們長久以來 100% 的支持與鼓勵，使我能專心致力於研究工作，並且堅持下去完成學業。僅以此論獻予你們，同時致上我最真摯的愛與祝福。

鎮源

于 新竹 2008.03

Chapter 1

Introduction

The rapid development of information technology over the past decades has brought human beings into the Information Age. With the advent of new technologies, the amount and the availability of online information have dramatically increased. There has been a large amount of information produced in the last decade. However, the process of information production never ends and is even going on at an extremely rapid growth rate. For example, the Internet Archive¹, an archive of snapshots of the Web, has collected almost 2 petabytes of data and is currently growing at a rate of 20 terabytes per month. The explosion of information has led to *information overload* (i.e., a state of having too much information to make a decision or remain informed about a topic), implying that finding and using the information that people really need efficiently and effectively has become a pressing practical problem in people's daily life.

An information retrieval (IR) system, (e.g., search engines, such as Google², Microsoft Live Search³, and Yahoo! Search⁴) can greatly facilitate the discovery of information by retrieving documents, which seem to be relevant to a user query. However, hundreds or even thousands of hits might be returned for a search, by which the user is often overwhelmed. Hence, it is still desirable to have other kinds of applications (e.g., document clustering, text categorization, question answering, and topic detection and tracking) to help people interact with information.

¹ <http://www.archive.org/>

² <http://www.google.com.tw/>

³ <http://www.live.com/>

⁴ <http://search.yahoo.com/>

Text summarization (TS), which can automatically digest information content from document(s) while preserving the underlying main points, is obviously one such application. This technique can potentially reduce the amount of text that people need to read, since, instead of a full document (or a set of related documents), only a brief summary needs to be read. For instance, by providing snippets of text for each match returned in a query, search engines can significantly help users identify the most relevant documents in a short time. The following gives other scenarios, mentioned in [57], where text summarization might be beneficial: (1) put a book on the scanner, turn the dial to ‘2 pages’, and read the result, (2) download 1,000 documents from the Web, send them to the summarizer, and select the best ones by reading the summaries of the clusters, and (3) forward the Japanese email to the summarizer, select ‘1 par’, and skim the translated summary. In general, text search and summarization are the two essential technologies that complete each other [45]: while text search engines serve as information filters to sift out an initial set of relevant documents, text summarizers play the role of information spotters to help users spot a final set of desired documents.

In this thesis, we present work on multidocument summarization, a task of producing a single summary of multiple documents on the same (or related) topic. In the following, Section 1.1 first gives a general background on text summarization, presenting an overview of the summarization process, discussing summarization factors, and sketching briefly the history of research in summarization, as well as the categorization of summarization techniques. Section 1.2 introduces the tasks and the challenges that are addressed in this thesis. Finally, Section 1.3 provides a guide to the remaining chapters.

1.1 Background

Text summarization is the creation of a shorten version of a text (or texts), while still preserving the underlying main points of the original text(s). By definition, text summarization is:

- (a) *the process of distilling the most important information from a source (or sources) to produce an abridged version for a particular user (or users) and task (or tasks) [87]; or*
- (b) *the process of taking a textual document, extracting content from it and presenting the most important content to the user in a condensed form and in a manner sensitive to the user's or application's needs [85].*

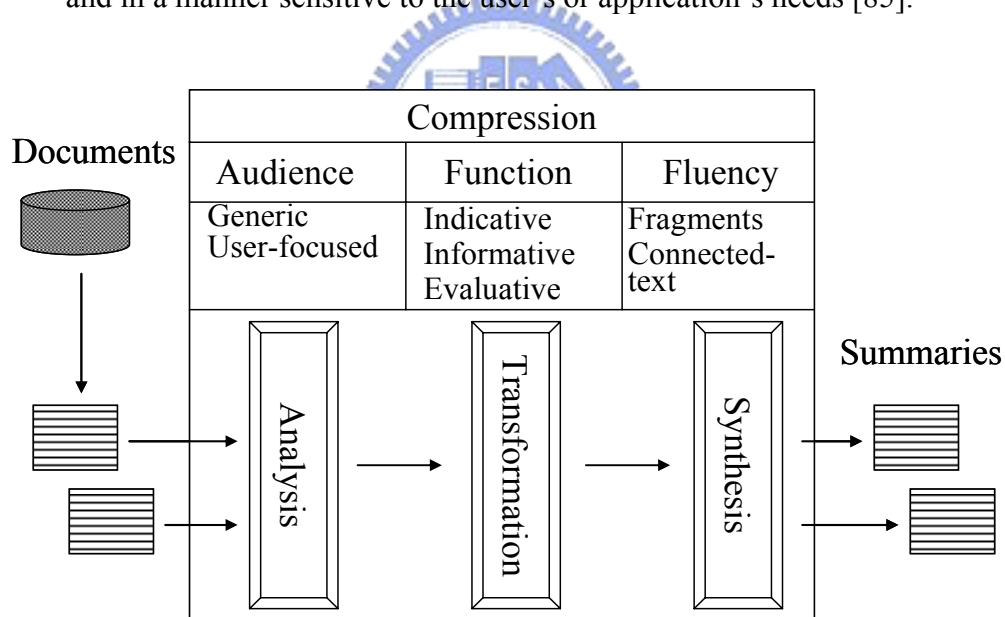


Fig. 1.1. Overview of the summarization process [87]

In general, the main challenge in text summarization is to identify the informative segments at the expense of the rest [110]. Fig. 1.1 illustrates a high-level overview of the process of text summarization. The input could be a single document or multiple related documents. The output summary may be an extract of the source(s) or an abstract. The summarization process, as mentioned in [87], can be decomposed

into three phases: (1) *analysis*, (2) *transformation*, and (3) *synthesis*. The analysis phase analyzes the input text and interprets it into a source representation. The transformation phase transforms the analysis results into a summary representation. Finally, the synthesis phase takes the summary representation as input, and produces an appropriate summary corresponding to the user's need.

In the whole process, factors, such as audience, function, and fluency, will lead to different types of the desired output summaries. This is further discussed in Section 1.1.2. As also shown in Fig. 1.1, another important factor in summarization is the *compression rate*, which is the ratio of the length of the summary to that of the original text(s). While the compression rate decreases, the length of summary gets shorter, indicating that more information is lost. The length of summary, on the other hand, becomes longer as the compression rate increases. However, in such a case, it tends to include more insignificant or redundant information in the summary. Traditionally, a compression rate ranging from 1% to 30% will produce a good summary [87]. See also [44], [48], [66] for more discussion.

1.1.1 History of text summarization

Text summarization has its first inception in the late 1950s, for the hope to automatically create abstracts from scientific articles. Due to the lack of powerful computers and the difficulties in nature language processing (NLP), early works were only based on the use of heuristics, such as term frequency (e.g., [82], [113]), lexical cues (e.g., [36]), and location (e.g., [36]), to determine which information units (e.g., words, phrases, sentences, or paragraphs) should be included into the summary. The principal shortcoming of this kind of approaches is that they depend very much on the particular format and style of writing [55], which limits these approaches only in

special domains.

In the late 1970s and the 1980s, research works turned to complex text processing, where techniques developed in artificial intelligence (AI) were exploited; for instance, the use of logic and production rules (e.g., [42]), scripts (e.g., [33], [69]), and semantic networks (e.g., [115]). See also [114]. The intuition behind these studies is to model text entities in knowledge representations (e.g., frames and templates) and to extract relationships between entities by inference. While these approaches have proven successful to a degree, the major drawback is that limitedly-defined knowledge representations may result in incomplete analysis of entities and their relationships.

Since the 1990s, dominant approaches turned to finding characteristic text units using statistical methods, techniques developed in information retrieval (IR), as well as hybrid approaches. See [7], [20], [29], [45], [55], [103], [119]. Early works mainly focused on analysis and representation in symbolic level (or word level), and did not take into account semantics, such as synonymy, polysemy, and term dependency [55]. Fortunately, from the mid-90s, the issue of semantics has been gradually addressed because more reliable natural language processing tools, such as for information extraction (IE) and sentence parsing, become available [127].

In recent years, supervised learning-based methods play an important role in text summarization. For example, [66], [76], [90], [128]. With the application of machine learning (ML), classification rules from documents and their corresponding summaries can be learned to determine whether a text unit should be included in the summary. The process of supervised summarization mainly consists of two phases: (1) *training*, and (2) *test*. While the training phase extracts appropriate features from the

training data set, and employs a learning algorithm to generate pattern rules, the test phase applies the rules on new documents to produce summaries. The advantage is that supervised learning-based methods can deliver effective systems without the effort of summarization model analysis [127].

Started from the late-90s, numerous large-scale evaluation programs (e.g., SUMMAC⁵, DUC⁶, and NTCIR-TSC⁷) and workshops have been run to measure the performance of summarization systems. Many standard collections for training and test on evaluation of summarization methods have been established recently, which leads to the great encouragement of the summarization work. For instance, since 2001, DUC (Document Understanding Conferences), sponsored by the National Institute of Standards and Technology (NIST), has held several evaluation competitions in *single-document summarization*, *multidocument summarization*, *cross-lingual summarization*, and *query-focused summarization*.



1.1.2 Summarization factors

The design and evaluation of a summarization system usually depend on several factors (e.g., the type of input documents, the purpose that the final summary should serve, and the ways of presenting a summary) that it takes into account during the development. The following factors, as outlined in [55], [107], are traditionally considered to yield the main categorization of research in text summarization. See also [16], [26], [126] for more discussion.

(1) Format: extract and abstract

⁵ http://www-nlpir.nist.gov/related_projects/tipster_summac/.

⁶ <http://duc.nist.gov/>.

⁷ <http://research.nii.ac.jp/ntcir/>.

A summary can be in the form of an *extract* of the source(s) or an *abstract*. An extract is usually created by the selection and verbatim inclusion of text units (e.g, sentences, paragraphs, and even phrases) from the original text(s). In contrast, an abstract involves the fusion of information content and the presentation of information in novel phrasings by natural language generation.

(2) Context: generic and query-focused

A summary can either be *generic* or *query-focused* (i.e., “user-focused” in Fig. 1.1). A generic summary reflects the author’s point of view and mainly concerns “what” described in the text(s). A query-focused summary, on the other hand, is customized to satisfy the user’s information need and to reflect particular points that are relevant to the user’s desired topic(s) of interest.

(3) Genre: indicative, informative, and evaluative

A summary can be *indicative*, only to suggest what a particular subject of the source(s) is about, without conveying any specific content. As an example, a list of keywords is an indicative summary. An *informative* summary conveys information pertinent to the source(s) and attempts to stand in place of the source(s) as a surrogate. A summary, such as a book review, is *evaluative* (or *critical*) to offer a critique of the source(s) [125].

(4) Dimension : single-document and multidocument

There are *single-document* and *multidocument* summarization, with respect to the number of input documents. While the input of single-document summarization (SDS) comprises only one document, multidocument

summarization (MDS) takes as input a set of topically-related documents, such as news articles on the same event. It would be much beneficial to create a summary of multiple documents. However, the need to identify important similarities and differences across documents makes multidocument summarization more challenging than single-document summarization [110].

(5) Linguality: monolingual, multilingual, and cross-lingual

The linguistic property is usually related to multidocument summarization. Monolingual summarization deals with documents, which are written in only one language. Multilingual summarization tries to determine the relevance of text portions and to generate multilingual texts based on information in different source languages. See [21], [70], [112]. As for cross-lingual summarization, the output summary is translated into another language different from the input one.

1.1.3 Summarization techniques

Nowadays, text summarization has reached a relatively mature stage [95]. Many summarization techniques have been developed and evaluated in the past decades. There are several ways to characterize the various summarization techniques. Based on the level of text processing, [87] categorized summarization techniques as approaching the problem at the *surface*, *entity*, or *discourse* levels. Surface-level approaches represent information using shallow features (e.g., term frequency, location, cue word, etc.) and combine these features to yield a salience function to measure the significance of information. For example, [54], [66], [76], [84], [90], [91], [103], [111], [128], [137]. Entity-level approaches model text entities and their relationships (e.g., co-occurrence, co-reference, syntactic relations, logical relations, etc.) and determine salient information based on text-entity models. See [6], [7], [33],

[51], [62], [69], [97], [135]. Discourse-level approaches model the global structure of the text (e.g., document format, rhetorical structure, etc.) and its relation to communicative goals. For instance, [17], [89].

[48], instead, classified summarization techniques into *knowledge-poor* or *knowledge-rich*, according to how much domain knowledge is involved in the summarization process. Knowledge-poor approaches do not consider any knowledge pertaining to the domain to which text summarization is applied. Therefore, they can be easily applied to any domains. For example, [45], [54], [55], [78], [91], [98], [119]. The principle behind knowledge-rich approaches is that the understanding of the meaning of a text can benefit the generation of a good summary. These approaches rely on a knowledge base of rules, which must be acquired, maintained, and then adapted to different domains. See [7], [49], [51], [62], [97], [116], [135]. In general, surface-level approaches are known as knowledge-poor approaches, while entity-level and discourse-level approaches are recognized as knowledge-rich approaches.

The following provides some examples of summarization techniques. [76] exploited a selection function for extraction, and used machine-learning to automatically learn an appropriate function to combine different heuristics. [5], [66], [90] regarded the task as a classification problem, and employed Bayesian classifier to determine which sentence should be included in the summary. [7] created summaries by finding lexical chains, relying on word distribution and lexical links among them to approximate content, and providing a representation of the lexical cohesive structure of the text. [6] used co-reference chains to model the structure of a document and to indicate sentences for inclusion in a summary. [45] proposed two methods: one used relevance measure to rank sentence relevance, and the other used latent semantic analysis to identify semantically important sentences. [55] attempted to create a robust

summarization system, based on the hypothesis: *summarization = topic identification + interpretation + generation*. The identification stage is to filter the input and retain the most important topics. In the interpretation stage, two or more extracted topics are fused into one or more unifying concept(s). The generation stage reformulates the extracted and fused concepts and then generates an appropriate summary.

1.1.3.1 Extraction vs. abstraction

The most common way to differentiate summarization techniques is the format of the summary being produced. Based on this, current summarization techniques can be characterized as *extraction* or *abstraction*. Today, most summarization systems, in fact, follow a broadly-used summarization model, namely *sentence extraction*, to produce extractive summaries. The paradigm identifies and extracts key sentences verbatim from the input source(s) based on a variety of different criteria and then concatenates them together to form a summary. [1] recognized two main categories of extractive techniques: (1) each sentence is assigned a weight based on various surface-level features, and is ranked in relation to the other sentences, so that the top-*n* ranked sentences could be extracted, and (2) machine learning and language processing techniques are employed to detect important sentences based on a graph representation of the input document(s). For a study discussing the potential and limitations of sentence extraction, please refer to [80].

Table 1.1 gives some examples that employ extractive techniques. The *input* field indicates the number of input documents and in what language that they are written. The *purpose* field concerns whether the summary is indicative, informative, or evaluative, generic or user-oriented. The *output* field means the “material” to create a summary. Finally, the *method* field outlines the specific methods used in the

summarization process.

Table 1.1. Examples that employ extractive techniques (excerpted from [1])

	Input	Purpose	Output	Method
[21]	Multidocument, Multilingual (English, Chinese)	Generic, Domain-specific (news)	Sentences	Use of keywords
[36]	Single-document, English	Generic, Domain-specific (scientific articles)	Sentences	Statistics (surface-level features), Use of thematic keywords
[82]	Single-document, English	Generic, Domain-specific (technical papers)	Sentences	Statistics (surface-level features)
[86]	Multidocument, English	User-oriented, General purpose	Text regions	Graph-based, Use of cohesion relations
[89]	Single-document, English	Generic, Domain-specific (scientific articles)	Sentences	Tree-based, RST
[119]	Single-document, English	Generic, General purpose	Paragraphs	Graph-based, Statistics (similarity)

Table 1.2. Examples that employ abstractive techniques (excerpted from [1])

	Input	Purpose	Output	Method
[7]	Single-document, English	Generic, Domain-specific (news)	Clusters	Syntactic processing
[33]	Single-document, English	Informative, User-oriented, Domain-specific	Scripts	Script activation
[97]	Multidocument, English	Informative, User-oriented, Domain-specific	Templates	Information Extraction
[117]	Single-document, English	Informative, User-oriented, Domain-specific	Ontology-based representation	Syntactic processing, Ontology-based Annotation

While the pre-dominant techniques are extractive, some summarization systems adopt abstractive techniques, in which the most important information is encoded and

fed into a natural language generation system to generate a summary in novel phrasings. [1] distinguished abstractive techniques into two categories: (1) the most important information is identified using prior knowledge about the structure of information, which is represented by cognitive schemas (e.g., scripts and templates), and (2) the most important is identified based on a semantic representation (e.g., noun phrases and their relations) of the document(s). Table 1.2 gives some examples that employ abstractive techniques.

1.1.4 Summary Evaluation

Evaluation is a critical issue in summarization. However, it has been proven as a difficult problem to evaluate the quality of a summary, principally because there is no obvious “ideal” summary due to the subjective aspect of summarization [110]. Therefore, the summarization community has practically used multiple model summaries for system evaluation to help alleviate this problem.

In general, the existing methods for evaluating text summarization approaches can be broadly classified into: (1) *extrinsic* evaluation, and (2) *intrinsic* evaluation (see [87]). The first judges the quality of a summary based on how it affects the completion of other tasks. For example, [21] proposed an evaluation model using question-answering: both the original text and its summary are processed by a question-answering system to extract answers for questions and the precisions and the recalls on the retrieved answer sets are compared. The second, on the other hand, judges the quality of summary based on coverage between the summary and model summaries, user judgments of informativeness, etc. For instance, SEE (Summary Evaluation Environment) [122] supports human evaluation, where an interface is provided for assessors to judge the quality of summaries in grammatically, cohesion,

and coherence. Automatic evaluation methods, such as ROUGE [79] and Pyramid [52], which measure the coverage, also fall in this category.

1.2 Tasks and Challenges

There are two research tasks discussed in this thesis: (1) *multidocument summarization*, and (2) *query-focused multidocument summarization*. The first focuses on producing a *generic* summary of a set of topically-related documents, while the second focuses on, given a user query, generating a *query-focused* summary of a set of topically-related documents to reflect particular points that are relevant to the user's desired topic(s) of interest. Both tasks are addressed in this thesis using the most common technique for summarization, namely sentence extraction: important sentences are identified and extracted verbatim from documents and composed into an extractive summary. The first step towards sentence extraction is obviously to score and rank sentences in order of importance, which is the major focus of this thesis.

1.2.1 Multidocument summarization

Early works on text summarization dealt with single-document summarization. Since the late-90s, the rapid increase and the availability of online texts have made multidocument summarization a worth problem to be solved. Given a collection of documents on the same (or related) topic (e.g., news articles on the same event from several newswires), summaries that deliver the majority of information content among documents and emphasize the differences would be significantly helpful to a reader. However, it is much harder towards multidocument summarization than towards single-document summarization, since several unique issues, such as anti-redundancy and content ordering, need to be addressed. In general, the major challenge of

multidocument summarization is to discover similarities across documents, as well as to identify distinct significant aspects from each one.

By the definition given in [110], multidocument summarization is the process of producing a single summary of a set of related documents, where three major issues need to be addressed: (1) *identifying important similarities and differences among documents*, (2) *recognizing and coping with redundancy*, and (3) *ensuring summary coherence*. Previous works have investigated various methods for solving these issues. For instance, sentence clustering to identifying similarities (e.g., [29], [44], [53], [96]), information extraction to facilitating the identification of similarities and differences (e.g., [97]), maximum marginal relevance (MMR) [20] and cross-sentence informational subsumption (CSIS) [111] to removing redundancy, and information fusion (e.g., [9]) and sentence ordering (e.g., [8]) to generating coherent summaries. For a general overview of the current state of the art, please refer to Chapter 2.

While many approaches to single-document summarization have been extended to deal with multidocument summarization (e.g., [22], [49], [77], [84]), there are still a number of new issues, as briefed below, needed to be addressed. See also [44].

(1) Lower compression rate:

Traditionally, a compression rate ranging from 1% to 30% is suitable for single-document summarization [87]. However, for multidocument summarization, the degree of compression rate is typically much low. For example, [44] found that a compression to the 1% or 0.1% level is required for summarizing 200 documents.

(2) Anti-redundancy:

The degree of redundancy in information contained in a group of related documents is usually high, due to the reason that each document in the group is apt to describe the main points as well as necessary shared background [44]. Therefore, it is necessary to minimize redundancy in the summary of multiple documents (i.e., to avoid including similar or redundant information into the summary).

(3) Information fusion:

One problem of the selection of a subset of similar passages in extraction-based approaches is the production of a summary biased towards some sources. Information fusion, which synthesizes common information, such as repetitive phrases, in the set of related passages into the summary, can alleviate this problem by the use of reformulation rules.

(4) Content ordering:

Content ordering is the organization of information from different sources to ensure the coherence of the summary. In single-document summarization, content ordering could be decided, based on the precedence orders in the original document. In multidocument summarization, instead, no single document can provide a global ordering of information in the summary.

In this study, we focus on extraction-based multidocument summarization to produce an extractive generic summary for a set of related news articles on the same event. In the approach that we propose in Chapter 3, the multidocument summarization task is divided into three sub-tasks: (1) *ranking sentences according to their likelihood of being part of the summary*, (2) *eliminating redundancy while*

extracting the most important sentences, and (3) organizing extracted sentences into a summary.

The focus of the proposed approach is a novel sentence ranking method to perform the first sub-task. The idea of modeling a single document into a text relationship map [119] is extended to model a set of topically-related documents into a sentence similarity network (i.e., a network of sentences, with a node referring to a sentence and an edge indicating that the corresponding sentences are related to each other), based on which a graph-based sentence ranking algorithm, *iSpreadRank*, is proposed.

iSpreadRank hypothesizes that the importance of a sentence in the network is related to the following factors: (1) the number of sentences to which it connects, (2) the importance of its connected sentences, and (3) the strength of relationships between it and its connected sentences. In other words, *iSpreadRank* supposes that a sentence, which connects to many of the other important sentences, is itself likely to be important. To realize this hypothesis, *iSpreadRank* practically applies spreading activation [106] to iteratively re-weight the importance of sentences by spreading their sentence-specific feature scores throughout the network to adjust the importance of other sentences. Consequently, a ranking of sentences indicating the relative importance of sentences is reasoned.

Given a ranking of sentences, in the second sub-task, a strategy of redundancy filtering, based on cross-sentence informational subsumption [111], is utilized to iteratively extract one sentence at a time into the summary, if it is not too similar to any sentences already included in the summary. In practice, this strategy only extracts high-scoring sentences with less redundant information than others. Finally, in the

third sub-task, a sentence ordering policy, which considers together topical relatedness and chronological order between sentences, is employed to organize extracted sentences into a coherent summary.

The proposed summarization method is evaluated using the DUC 2004 data set, and found to perform well. Experimental results show that the proposed method obtained a ROUGE-1 score of 0.38068, which is competitive to that of the 1st-ranked system at DUC 2004.

1.2.2 Query-focused multidocument summarization

Query-focused multidocument summarization is a particular task of multidocument summarization. Given a cluster of documents relevant to a specific topic, a query statement consisted of a set of related questions, and a user profile, the task is to create a brief, well-organized, fluent summary which either answers the need for information expressed in the query statement or explains the query, at the level of granularity specified in the user profile. Table 1.3 and Table 1.4 give examples of the query statements. The level of granularity, here, can be either specific or general: while a general summary prefers a high-level generalized description biased to the query, a specific summary should describe and name specific instances of events, people, places, etc.

As stated in [3], this task can be seen as topic-oriented, informative multidocument summarization, where the goal is to produce a single text as a compressed version of a set of documents with a minimum loss of relevant information. This suggests that a good summary for query-focused multidocument summarization should not only best satisfy the need for information expressed in the query statement but also need to cover as much of the important information as

possible across documents [136].

Table 1.3. Query statement for set d357i with granularity specified as “specific”

<pre><topic> <num> d357i </num> <title> Boundary disputes involving oil </title> <narr> What countries are or have been involved in land or water boundary disputes with each other over oil resources or exploration? How have disputes been resolved, or towards what kind of resolution are the countries moving? What other factors affect the disputes? </narr> <granularity> specific </granularity> </topic></pre>

Table 1.4. Query statement for set d376e with granularity specified as “general”

<pre><topic> <num> d376e </num> <title> World Court </title> <narr> What is the World Court? What types of cases does the World Court hear? </narr> <granularity> general </granularity> </topic></pre>

In general, the challenges of query-focused multidocument summarization are twofold. The first one is to identify important similarities and differences among documents, which is a common issue of multidocument summarization. The second one is the need to take into account query-biased characteristics during the summarization process.

In this study, we focus on extraction-based query-focused multidocument summarization to produce an extractive query-focused summary, which reflects particular points relevant to user’s interests, for a set of related news articles on the same event. In the approach that we propose in Chapter 4, the query-focused multidocument summarization task is divided into four sub-tasks: (1) *examining the degree of relevance between each sentence and the query statement*, (2) *ranking*

sentences according to their degree of relevance to the query and their likelihood of being part of the summary, (3) eliminating redundancy while extracting the most important sentences, and (4) organizing extracted sentences into a summary.

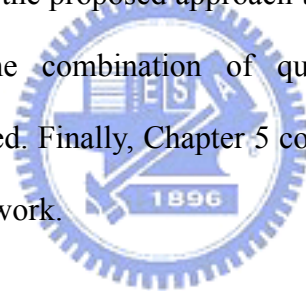
The first sub-task is addressed as a query-biased sentence retrieval task. For each sentence s , given a query q , the degree of relevance between s and q is measured as the degree of similarity between them, i.e., $sim(s, q)$. Three similarity measures are proposed to assess $sim(s, q)$. The first is computed as the dot production of the vectors of s and q in the vector space model. The second exploits latent semantic analysis (LSA) [32] to fold s and q into a reduced semantic space and computes their similarity based on the transformed vectors of s and q in the semantic space. Finally, with the idea of model averaging, the third combines the similarities obtained from the first and the second in a linear manner.

In the second sub-task, several surface-level features are extracted to measure how representative a sentence is with respect to the whole document cluster. The feature scores, acting as the strength of representative power (i.e., the informativeness) of each sentence, are combined with the degree of relevance between the sentence and the query to score all sentences. As for the third sub-task, a novel sentence extraction method, inspired by maximal marginal relevance (MMR) [20] for redundancy filtering, is utilized to iteratively extract one sentence at a time into the summary, if it is not too similar to any sentences already included in the summary. In one iteration, all the remaining unselected sentences are re-scored and ranked using a modified MMR function, so as to extract the sentence with the highest score. Finally, in the fourth sub-task, all extracted sentences are simply ordered chronologically to form a coherent summary.

The proposed summarization method is evaluated using the DUC 2005 data set, and found to perform well. Experimental results show that the proposed method obtained a ROUGE-2 score of 0.07265 and a ROUGE-SU4 score of 0.12568, which are competitive to those of the 1st-ranked and 2nd-ranked systems at DUC 2005.

1.3 Guide to Remaining Chapters

The remainder of this thesis is organized as follows: Chapter 2 provides a survey of the current state of the art in multidocument summarization and query-focused multidocument summarization. While Chapter 3 introduces the proposed approach to multidocument summarization that is based on a graph-based sentence ranking algorithm, Chapter 4 presents the proposed approach to query-focused multidocument summarization in which the combination of query-biased characteristics and surface-level features is studied. Finally, Chapter 5 concludes this thesis and provides possible directions for future work.



Chapter 2

Literature Survey

In this chapter, Section 2.1 and Section 2.2 provide a sketch of the current of the art of multidocument summarization, and of query-focused multidocument summarization, respectively. Section 2.3 introduces some example research projects in the field.

2.1 Multidocument Summarization

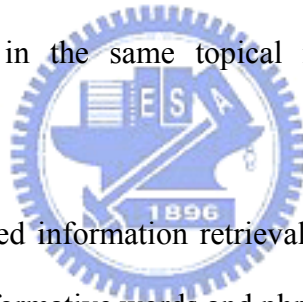
[97] pioneered work on multidocument summarization. They established relationships between news stories by aggregating similar extracted templates using logical relationships, such as agreement and contradiction. The summary was constructed by a sentence generator based on the facts and their relationships in the templates. These template-based methods are still of interests recently (see [51], [135]). However, manual efforts are required to define domain-specific templates, while poorly-defined templates may lead to incomplete extraction of facts.

Most recent studies have adopted clustering to identify themes⁸ (i.e., clusters of common information) (e.g., [9], [14], [29], [44], [53], [96], [101]). These approaches are founded on an observation that multiple documents concerning a particular topic tend to contain redundant information, as well as information unique to each [29]. Once themes have been recognized, a representative passage in each theme is selected and included in the summary; alternatively, repeated phrases in clusters are exploited to generate an abstract-like summary by information fusion [110].

Typical research on theme clustering is briefed as follows. [9] and [96]

⁸ A theme, also viewed as a sub-topic, is defined as a group of passages (such as sentences and paragraphs) which all convey approximately the same information [96].

discovered common themes using graph-based clustering based on features, such as word co-occurrence, noun phrase matching, synonymy matching, and verb semantic matching. Similar phrases in the identified themes were synthesized into a summary by information fusion using natural language generation. [44] grouped paragraphs into clusters and collected, into the summary, from each group a significant passage with large coverage and low redundancy, measured by maximal marginal relevance (MMR) [20]. This strategy aimed at high relevance to the query and keeping redundancy low in the summary. [29] evaluated several policies for choosing indicative sentences from sentence clusters and concluded that the best policy is to extract sentences with the highest sum of relevance scores for each cluster. [101] clustered sentences as topical regions. Seed paragraphs, each having the maximum total similarity with others in the same topical region, are considered as the representative passages.



Other studies have applied information retrieval and statistical methods to find salient concepts, as well as informative words and phrases in multiple documents (e.g., [43], [49], [65], [77], [111]). For instance, [111] detected a set of statistically important words as the topic centroid of a document cluster, which is treated as a feature and considered together with other heuristics to extract sentences. [77] recognized key concepts by calculating log-likelihood ratios of unigrams, bigrams and trigrams of terms, and then clustered these concept-bearing terms to detect sub-topics. Each sentence in the document set was ranked using key concepts in order to produce an extractive summary. [49] discussed different strategies to create signatures of topic themes and evaluated methods to use them in summarization.

Surface-level features extended from the well-developed single-document summarization methods have also been exploited (e.g., [54], [84], [91], [111]).

Heuristics-based approaches selectively combine features to yield a scoring function for the discrimination of salient text units. Commonly used features include sentence position, sum of TF-IDF in a sentence, similarity with headline, sentence cluster similarity, etc. Alternatively, there are approaches that apply machine learning to automatically combine surface-level features from a corpus of documents and their summaries. For instance, [54] used support vector machines (SVM) [132] to learn a sentence ranking model.

Techniques depending on a thorough analysis of the discourse structure of the text have been explored (e.g., [11], [15], [22], [62], [139]). [139] developed a cross-document structure theory (CST) to define the cross-document rhetorical relationships between sentences across documents. The cohesion of extractive summaries is found to be meliorated by the CST relationships. [15] and [22] built lexical chains to identify topics in the input texts. Sentences are ranked according to the number of word co-occurrences in the chains and sentences. [11] constructed noun phrase co-reference chains across documents based on a set of predefined word-level fuzzy relations. The most important noun phrases in important chains are selected to score sentences.

Researchers have also investigated graph-based approaches. [86] modeled term occurrences as a graph using cohesion relationships (e.g., synonymy, and co-reference) among text units. The similarities and differences in documents are successfully pinpointed by applying spreading activation [106] and graph matching. Sentences are extracted based on a scoring function which measures term weights in the activated graph. [124] constructed a graph using the similarity relations between sentences. The summary is generated by traversing sentences along a shortest path of the minimum cost from the first to the last sentence. [138] presented a bipartite graph of texts where

spectral graph clustering is applied to partition sentences into topical groups.

Some graph-based methods employ the concept of centrality in social network analysis. [119] first attempted such an approach for single-document summarization. They proposed a text relationship map to represent the structure of a document, and utilized the degree-based centrality to measure the importance of sentences. Later works following the idea of graph-based document models employed distinct ranking algorithms to determine the centralities of sentences. [39] recognized the most significant sentences by a sentence ranking algorithm, LexRank, which performs PageRank [18] on a sentence-based network according to the hypothesis that sentences similar to many other sentences are central to the topic. [38], [133] examined biased PageRank to extract the topic-sensitive structure beyond the text graph for question-focused summarization. [98] examined several graph ranking methods originally proposed to analyze webpage prestige, including PageRank and HITS [64], for single-document summarization. [100] extended the algorithm of [98] for multiple documents. A meta-summary of documents is produced from a set of single-document summaries in an iterative manner. [140] proposed a cue-based hub-authority approach that brings surface-level features into a hub/authority framework. HITS is used in their work to rank sentences.

Last but not least, other graph-based works build a dependency graph with a word as a node and a syntactical relation as a link. One good example is [130] for event-focused news summarization, which employed PageRank to identify word entities participating in important events or relationships among all documents.

2.2 Query-focused Multidocument Summarization

The major difference of query-focused multidocument summarization, compared to multidocument summarization, is the need to measure the relevance of a sentence to the user query. Hence, most research works have regarded query-biased sentence retrieval as the first step towards query-focused multidocument summarization. For example, [31] employed a Bayesian language model to estimate the relevance between a sentence and the query. They found that the Bayesian model consistently works well even when there is significantly less information in the query. [47] presented a system which measures relevance and redundancy of sentences using a latent semantic space, constructed over a very large corpus. [59] combined multiple strategies, including relevance-based language modeling [68], latent semantic indexing [32], and word overlap, to identify query-relevant sentences. [136] proposed concept links to compute the similarity between a sentence and the query in semantic level. They showed that concept links outperforms similarity measures based on word co-occurrence since semantically-related words are highlighted. [121] investigated a tree matching algorithm to obtain a similarity between a sentence and the query based on their dependency parsing trees.

Other studies have applied statistical methods to detect query-related words, based on which the relevance of a sentence to the query is assessed. Typical examples are given as follows. [46] compared two weighting schemes for estimating word importance: (1) raw word frequency, and (2) log-likelihood ratio (LLR). They concluded that LLR is more suitable for query-focused summarization since it is more sensitive to query relevance. [131] computed the importance of each word as a linear combination of the unigram probabilities derived from the query and those from the

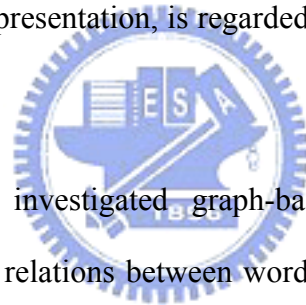
documents. Sentences, which have more words with the highest importance, are extracted to produce a summary. [27] exploited key-phrase extraction to identify relevant terms and used machine learning to select significant key-phrases. Summaries are generated according to the coverage of query-relevant phrases contained in the sentences.

Methods have also been explored to combine query-dependent and query-independent (i.e., surface-level) features for the assessment of the importance of a sentence. For instance, [72] and [73] exploited various features to judge whether a sentence is appropriate to be included in the summary. The features that they used include word-based features (e.g., word overlap, and cosine similarity), entity-based features (e.g., named entity), semantic-based features (e.g., WordNet-based similarity), and global features (e.g., sentence length, and position). [60] calculated the relevance of a sentence to the query based on relevance-based language models [68] along with semantic representation of words in HAL semantics spaces [83]. The relevance of a sentence is combined with the informativeness of a sentence, which is measured using a unigram-based language model trained on the Web.

The application of machine learning has been tried to automatically combine features from a corpus of documents and their summaries to learn a sentence ranking model. [24] introduced an “oracle” score, based on the probability distribution of unigrams trained from human summaries. Each sentence is scored as the average of the probabilities of words that it contains. [40] combined two models to obtain a global sentence ranking model for extraction: (1) query-neutral ranking trained using a perceptron-based ranking model [25], and (2) query-focused ranking learned based on raw term frequency. [129] presented a trainable extractive summarization system that learns a log-linear sentence ranking model by maximizing three metrics of

sentence goodness, based on ROUGE [79] and Pyramid [52] scores against model summaries. [41] and [74] employed support vector machines (SVM) [132] to automatically combine various features to generate a scoring function for extraction.

Some approaches depend on deep discourse analysis to extract query-relevant sentences. [142] utilized lexical chains and document index graphs to score sentences. [75] built, for each document, a set of lexical chains, and merged chains into a global representation of the document cluster. [62] represented documents as a composite topic tree, in which each node stands for one topic identified from documents. Using the topic tree, nodes containing query-related information are extracted to form a summary. [56] proposed a method based on basic elements. A basic element, defined as a head-modifier-relation representation, is regarded as a basic unit to determine the salience of a sentence.



Researchers have also investigated graph-based approaches. [86] used a document graph to formalize relations between words inside a document. Spreading activation [106] and graph matching are applied to perform query-biased summarization. [17] created a graph representation for a document based on the rhetorical structure theory (RST) [88]. Given the graph model, a graph search algorithm is exploited to identify relevant sentences. [38] and [133] examined biased PageRank [18] to extract the topic-sensitive structure beyond the text graph. [134] employed manifold-ranking [141] to rank sentences based on the biased information richness of a sentence.

Last but not least, [123] proposed a summarizer that focuses on subjectivity analysis. The summarizer generates summaries to reflect information need based on subjectivity clues. [10] introduced a statistical model for query-relevant

summarization. The statistical information is learned with a collection of FAQs using maximum-likelihood estimation. [12] created a system to produce summaries for definitional and biographical question-answering. [67] presented a framework for question-directed summarization, which uses multiple question decomposition and summarization strategies to create a single responsive summary-length answer for a complex question.

2.3 Related Research Projects

This section offers a brief introduction of example research projects in the field.

2.3.1 PERSIVAL

PERSIVAL (PErsonalized Retrieval and Summarization of Image, Video, And Language)⁹ [94] is designed to provide personalized access to a distributed patient care digital library. The system consists of: (1) a query component that uses clinical context to help formulate user queries, (2) a search component that uses machine learning to find relevant sources and patient information, and (3) a personalized presentation component that uses patient information and domain knowledge to summarize related multimedia resources. A multidocument summarization system, CENTRIFUSER [61] (see also [37], [62], [63]), is integrated in PERSIVAL to support personalized summarization. CENTRIFUSER models all the input documents into a composite topic tree, with a node standing for one topic (e.g., disease, symptom, etc.) extracted from documents. Using the topic tree, CENTRIFUSER determines which parts of the tree are relevant to the query and the patient information, and then extracts related parts to create a summary.

⁹ <http://persival.cs.columbia.edu/>

2.3.2 NewsBlaster

NewsBlaster¹⁰ [92], [93] is an on-line news summarization system, which supports topic detection, tracking, and summarization for daily browsing of news. The core summarization module, Columbia Summarizer, is composed of: (1) router, (2) MultiGen [96], and DEMS [120]. The router determines which type an input event cluster is and forwards the cluster to a suitable summarization module. The type, here, can be *single-event*, *multi-event*, *biography*, and *other*. MultiGen generates a concise summary based on the detection of similarities and differences across documents. Machine learning and statistical techniques are exploited to identify groups of similar passages (i.e., themes), followed by information fusion [9] to synthesize common information into an abstractive summary using natural language generation. While MultiGen is designed to cope with topically-related documents, DEMS is more general for loosely-related documents. DEMS combines features for new information detection and uses heuristics to extract important sentences into a summary.

2.3.3 MEAD

MEAD¹¹ [111] is an essentially statistical summarizer in public domain, developed to produce extractive summaries for either single- or multi-document summarization by sentence extraction. MEAD consists of: (1) feature extractor, (2) sentence scorer, and (3) sentence re-ranker. The feature extractor extracts summarization-related features from sentences, such as *position*, *centroid*, *cosine with query*, and *length*. The sentence scorer combines various features to measure the salience of a sentence. Finally, the sentence re-ranker iteratively selects candidate summary sentences while redundant sentences are avoided by checking similarity against prior selected ones.

¹⁰ <http://newsblaster.cs.columbia.edu/>

¹¹ <http://www.summarization.com/mead/>

NewsInEssence¹² [108] and WebInEssence [109] are two practical applications of MEAD. Given the user's interest, NewsInEssence retrieves related news articles from different online newswires and produces an extractive summary according to the user-specified parameters. WebInEssence, instead, is integrated into a general-purpose Web search engine to summarize the returned search results.

2.3.4 GLEANS

GLEANS [30] is a multidocument summarization system. The system classifies document clusters into a category, in which the content is about *single person*, *single event*, *multiple events*, or *natural disaster*. For each category, GLEANS maintains a set of predefined templates. Text entities and their logical relations are first identified, and mapped into canonical, database-like representations. Then, sentences, which conform to predefined coherence constraints, are extracted to form the final summary.

2.3.5 NeATS

NeATS [77], [78] is an extractive summarizer for multidocument summarization. The system is composed of: (1) content selection, (2) content filtering, and (3) content presentation. The content selection module recognizes key concepts by calculating likelihood ratios of unigrams, bigrams, and trigrams of terms. The content filtering module extracts sentences based on term frequency, sentence position, stigma words, and maximum marginal relevance [20]. Finally, the content presentation module exploits term clustering and explicit time annotation to organize important sentences into a coherent summary.

iNeATS [71] is a derivative of NeATS. The system allows users to dynamically

¹² <http://www.newsinessence.com/>

control over the summarization process. Furthermore, it supports the linking from the summary to the original documents, as well as the visualization of the spatial information, indicated in the summary, on a geographical map.

2.3.6 GISTexter

GISTexter [51] is designed to produce both extracts and abstracts for single- and multi-document summarization. The core of GISTexter is an information extraction (IE) system, CICERO [50], which identifies entities and fills relevant information, such as text snippets and co-reference information, into predefined IE-style templates using pattern rules. To generate summaries, GISTexter chooses representative templates and extracts source sentences for the template snippets.

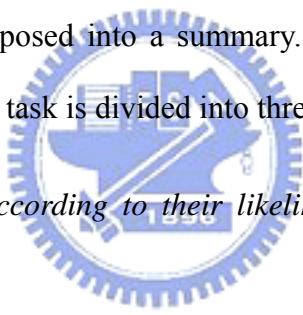


Chapter 3

Multidocument Summarization

Multidocument summarization refers to the process of producing a single summary of a set of *topically-related* documents (i.e., a set of documents on the same or related, but unspecified topic). In this chapter, we deal with multidocument summarization using an extraction-based approach to create an extractive generic summary of multiple documents.

The proposed approach follows the most common technique for summarization, namely *sentence extraction*: important sentences are identified and extracted verbatim from documents and are composed into a summary. In the proposed approach, the multidocument summarization task is divided into three sub-tasks:

- 
- (1) *Ranking sentences according to their likelihood of being part of the summary;*
 - (2) *Eliminating redundancy while extracting the most important sentences;*
 - (3) *Organizing extracted sentences into a summary.*

The focus of the proposed approach is a novel sentence ranking method to perform the first sub-task. The idea of modeling a single document into a text relationship map [119] is extended to model a set of topically-related documents into a sentence similarity network (i.e., a network of sentences, with a node referring to a sentence and an edge indicating that the corresponding sentences are related to each other), based on which a graph-based sentence ranking algorithm, *iSpreadRank*, is proposed.

iSpreadRank hypothesizes that the importance of a sentence in the network is

related to the following factors: (1) the number of sentences to which it connects, (2) the importance of its connected sentences, and (3) the strength of relationships between it and its connected sentences. In other words, iSpreadRank supposes that a sentence, which connects to many of the other important sentences, is itself likely to be important. To realize this hypothesis, iSpreadRank practically applies spreading activation [106] to iteratively re-weight the importance of sentences by spreading their sentence-specific feature scores¹³ throughout the network to adjust the importance of other sentences. Consequently, a ranking of sentences indicating the relative importance of sentences is reasoned.

Given a ranking of sentences, in the second sub-task, a strategy of redundancy filtering, based on cross-sentence informational subsumption [111], is utilized to iteratively extract one sentence at a time into the summary, if it is not too similar to any sentences already included in the summary. In practice, this strategy only extracts high-scoring sentences with less redundant information than others. Finally, in the third sub-task, a sentence ordering policy, which considers together topical relatedness and chronological order between sentences, is employed to organize extracted sentences into a coherent summary.

This chapter is structured as follows: Section 3.1 introduces the design of the proposed approach to multidocument summarization. Section 3.2 describes technical details of the proposed graph-based sentence ranking algorithm, iSpreadRank, as well as the proposed summarization approach. The experimental results are reported in Section 3.3 and finally Section 3.4 provides discussions about the proposed summarization approach in different aspects.

¹³ The sentence-specific feature scores work as the local information of every sentence, and are considered together with relationships between sentences to help derive the global information of sentences (i.e., the relative importance of sentences).

3.1 Design

Fig. 3.1 illustrates the design of the proposed approach to multidocument summarization. The input is a group of topically-related documents. The output is a concise summary which provides the condensed essentials of the input documents. The summarizer takes all the documents as a single document and produces an extractive summary by selecting characteristic sentences from the document group. All sentences in the document group are first ranked according to their degree of importance. Based on the ranking of sentences, the summarizer then iteratively extracts one sentence at a time, which not only is important but also has less redundancy than other sentences extracted prior to it. The extraction finishes once the required length of the summary is met. The extracted sentences are finally composed into the output summary.

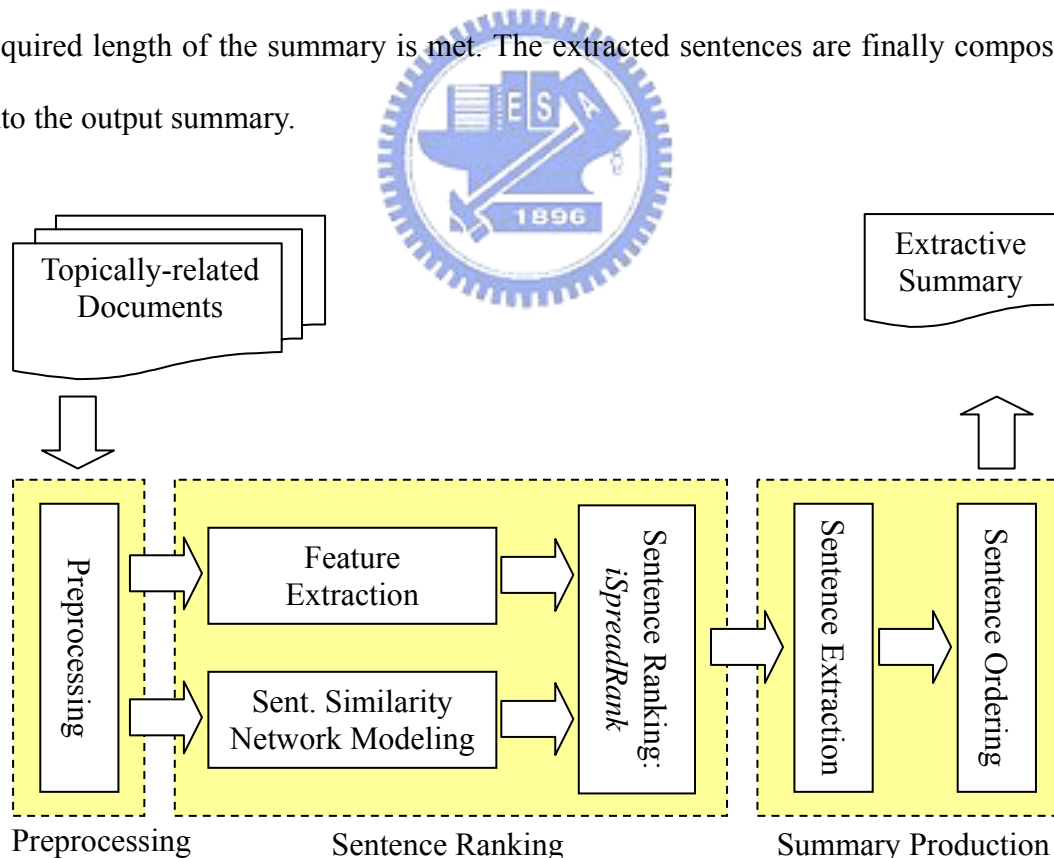


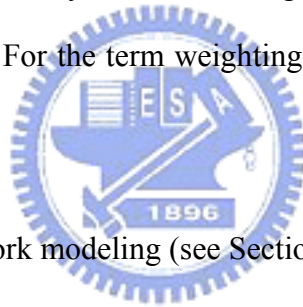
Fig. 3.1. The proposed multidocument summarization approach

The whole summarization process can be decomposed into three phases: (1) the *preprocessing* phase preprocesses the input documents, (2) the *sentence ranking* phase

ranks sentences according to their likelihood of being part of the summary, and (3) the *summary production* phase creates the output summary. The entire process, as shown in Fig. 3.1, can be further divided into several stages, namely *preprocessing*, *feature extraction*, *sentence similarity network modeling*, *sentence ranking*, *sentence extraction*, and *sentence ordering*. They are outlined as follows, in order of execution:

(1) Preprocessing:

Several linguistic analysis steps are carried out in this stage. A tokenizer segments text into words, numbers, symbols and punctuations. A sentence splitter identifies the boundaries of sentences. A passage indexer constructs a vector representation of every sentence using the well-known TF-IDF term weighting scheme [118]. For the term weighting scheme, please refer to Section 3.2.1.



(2) Sentence similarity network modeling (see Section 3.2.1):

The input documents are transformed into a *sentence similarity network*, with a node referring to a sentence, and an edge indicating that the corresponding sentences are related to each other. The relationship between a pair of sentences is measured as the level of their lexical overlap.

(3) Feature extraction (see Section 3.2.2):

A feature profile is created to capture the values of various sentence-specific features of all sentences. Three surface-level features are employed: (1) *centroid*, (2) *position*, and (3) *first-sentence overlap*. The feature scores, acting as the local information of every sentence, are integrated into the sentence ranking algorithm

to help derive the global information of sentences (i.e., the relative importance of sentences).

(4) Sentence ranking (see Section 3.2.3):

A graph-based sentence ranking algorithm, *iSpreadRank*, takes a sentence similarity network and a feature profile as inputs, and applies spreading activation [106] to iteratively re-weight the importance of sentences by spreading their sentence-specific feature scores (computed in the feature extraction stage) throughout the network to adjust the importance of other sentences. A ranking of sentences is finally inferred in order of their relative importance.

(5) Sentence extraction (see Section 3.2.4):

A sentence extraction module, based on cross-sentence informational subsumption [111] for redundancy filtering, iteratively examines sentences in the rank order, and adds one sentence at a time into the summary, if it is not too similar to any sentences already in the summary. Here, the degree of redundancy between two sentences is determined by a threshold imposed on the sentence similarity. In this way, only high-scoring sentences with less redundant information than others are extracted into the summary.

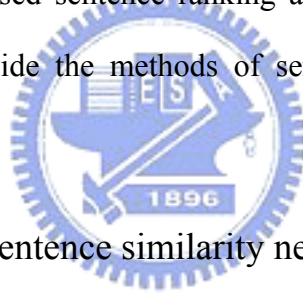
(6) Sentence ordering (see Section 3.2.5):

The final summary is structured in the following steps: Semi-similar sentences in the extracted sentence set are first grouped together, based on another similarity threshold smaller than that used in sentence extraction. Each group is then ordered chronologically into a macro-ordering according to the

earliest timestamp of the sentences within it. Finally, micro-ordering is applied to sort all sentences in each group in chronological order. This policy, considering together topical relatedness and chronological order between sentences, is similar to the augmented sentence ordering algorithm proposed in [8], in which the topical relatedness between sentences is determined by text cohesion in their original documents.

3.2 Algorithm

Section 3.2.1 describes the modeling of a group of documents into a sentence-based network. Section 3.2.2 presents the extraction of sentence-specific features. Section 3.2.3 introduces the graph-based sentence ranking algorithm, iSpreadRank. Section 3.2.4 and Section 3.2.5 provide the methods of sentence extraction and sentence ordering, respectively.



3.2.1 Text as a graph: sentence similarity network

[119] used the techniques for inter-document link generation to produce intra-document links between passages of a document, and obtained a *text relationship map* (or a *content similarity network*) to characterize the structure of the text based on its linkage patterns. In this section, the same idea is extended to model a group of documents into a network of sentences that are related to each other, resulting in a *sentence similarity network*.

Fig. 3.2 gives an example of the network. A sentence similarity network is defined as a graph with nodes and edges linking nodes. Each node in the network stands for a sentence. Two sentences are connected if and only if they are similar to each other. Hence, an edge between two nodes indicates that the corresponding two

sentences are considered to be “semantically related” [119].

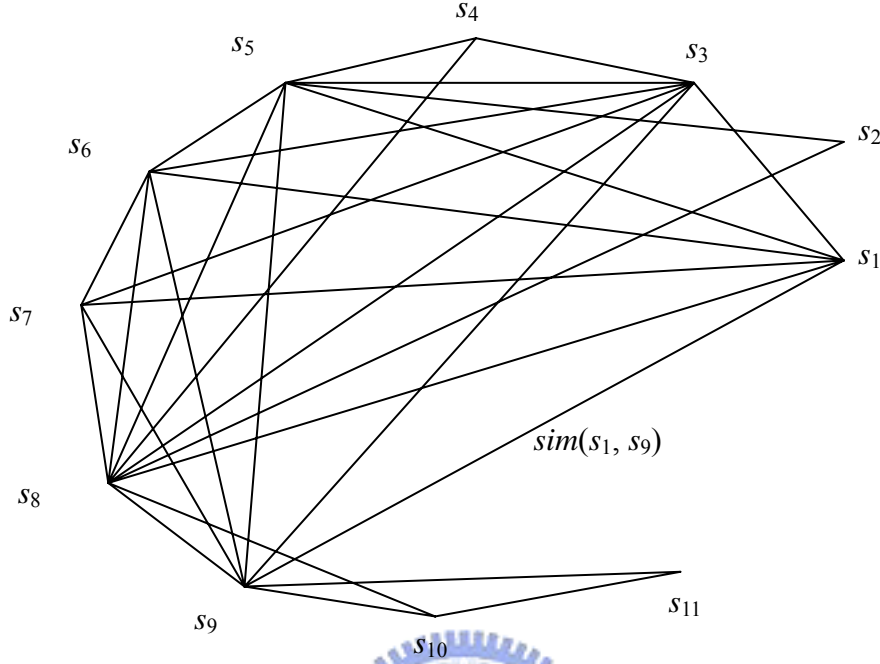


Fig. 3.2. A sentence similarity network

In order to construct such a network, each sentence is represented as a vector of weighted terms, based on which the similarity between two sentences is obtained to determine if there exists an edge between them. Let $W = \{t_1, \dots, t_n\}$ ($|W| = n$) denote the set of index terms in the document group. The vector representation of a sentence s_j is specified by Eq. (3.1), where $w_{i,j}$ is the TF-IDF weight of term t_i in s_j , given in Eq. (3.2).

$$s_j = \langle w_{1,j}, w_{2,j}, \dots, w_{n,j} \rangle \quad (3.1)$$

$$w_{i,j} = \frac{tf_{i,j}}{\max_l tf_{l,j}} \times \log\left(\frac{N}{n_i}\right) \quad (3.2)$$

In Eq. (3.2), $tf_{i,j}$ is the number of occurrences of t_i in s_j , N indicates the number of sentences in the document group, and n_i denotes the number of sentences where t_i appears.

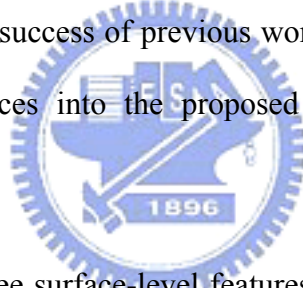
The degree of similarity between a pair of sentences s_i and s_j is computed, by Eq. (3.3), as the cosine of the angle between the vectors of \vec{s}_i and \vec{s}_j .

$$\text{sim}(s_i, s_j) = \frac{\vec{s}_i \cdot \vec{s}_j}{|\vec{s}_i| \times |\vec{s}_j|} \quad (3.3)$$

An edge between s_i and s_j exists if $\text{sim}(s_i, s_j)$ is greater than a similarity threshold, α . In the current implementation, α is empirically set to 0.1.

3.2.2 Feature extraction

In the literature, a variety of surface-level features have been profitably employed to determine the likelihood of sentences of being part of the summary (e.g., [66], [76], [103], [137]). Inspired by the success of previous works, we also attempt to integrate the feature scores of sentences into the proposed graph-based sentence ranking algorithm.



We take into account three surface-level features: (1) *centroid*, (2) *position*, and (3) *first-sentence overlap*. All of these features (see Table 3.1) have been evaluated and found as effective predictors of the salience of sentences for multidocument summarization in [111].

(1) f_1 : Centroid

This feature measures the relatedness between a sentence and the centroid of the input set of documents. A sentence with more centroid words is considered to be more central to the topic.

(2) f_2 : Position

Important sentences tend to appear in particular positions (e.g., the beginning or the end) in the document. This feature is computed as inversely proportional to the position of a sentence from the beginning.

(3) f_3 : First-sentence overlap

The first sentence usually provides an overview of a document. This feature is determined as the inner-product similarity of a sentence and the first sentence in the same document.

Table 3.1. The sentence-specific feature set (excerpted from [111])

Feature name	Feature value
f_1 : Centroid	$Score_{f_1}(s) = \sum_{w \in s} f(w, s) \times C(w)$ <p>$f(w, s)$: the number of occurrences of term w in sentence s $C(w)$: the centroid value of w</p>
f_2 : Position	$Score_{f_2}(s) = \frac{(n_d - i + 1)}{n_d}$ <p>i: the position of sentence s in document d n_d: the total number of sentences in d</p>
f_3 : First-sentence overlap	$Score_{f_3}(s) = \vec{s}_1 \cdot \vec{s}$ <p>s_1: the first sentence in the same document with sentence s</p>
Combination	$Score(s) = \sum_i w_i \times Score_{f_i}(s)$ <p>w_i: the weight of f_i in the linear combination</p>

A feature profile is generated to capture the scores of features of all sentences, and is input to the sentence ranking algorithm. Each feature score in the feature profile is further normalized into the range between 0 and 1. The feature scores, acting as the local information of every sentence, are integrated into the sentence ranking algorithm to help derive the global information of sentences (i.e., the relative importance of sentences).

3.2.3 Ranking the importance of sentences

The proposed sentence ranking algorithm, iSpreadRank, is designed to rank the importance of sentences for extraction-based summarization. iSpreadRank practically applies spreading activation [106] to realize the hypothesis that the importance of a sentence in the network is related to the following factors: (1) the number of sentences to which it connects; (2) the importance of its connected sentences, and (3) the strength of relationships between it and its connected sentences.

Spreading activation is originally developed in psychology to explain the cognitive process of human comprehension through semantic memory (see [4], [23], [106]). The theory claims that human's long-term memory is structured as an associative network in which similar memory units have strong connections and dissimilar units have none or weak connections. Accordingly, a memory retrieval is viewed as a searching across the network by activating a set of source nodes with *stimuli* (or *energy*), then iteratively propagating the energy in parallel along links throughout the network to other connected nodes, to discover more related nodes with hidden information.

Spreading activation has recently been applied in many other research fields, such as information retrieval (e.g., [13]), hypertext structure analysis (e.g., [105]), Web trust management (e.g., [143]), collaborative recommendation (e.g., [58]), and so forth. This section takes spreading activation one step further, and discusses the combination of sentence-specific feature scores and the sentence similarity network model together, under the framework of spreading activation, to reason the relative importance of sentences.

3.2.3.1 iSpreadRank

The inputs to iSpreadRank comprise a sentence similarity network (see Section 3.2.1) and a feature profile (see Section 3.2.2). The output is a ranking of sentences indicating the importance of all sentences, in the order from the highest to the lowest. iSpreadRank adopts a particular model of spreading activation, namely the *Leaky Capacitor Model* [4], and operates in three steps: (1) *initialization*, (2) *inference*, and (3) *prediction*.

The initialization step transforms the input sentence similarity network into a matrix representation for later computation. The inference step applies spreading activation to reason the relative importance of sentences, where the sentence-specific local importance of each sentence, initialized by the input feature profile, is iteratively spread throughout the whole network to adjust the importance of other neighboring sentences. In this step, the algorithm iterates until an equilibrium state of the network is obtained. Finally, the prediction step outputs a ranking of sentences according to the inference results in the inference step. In summary, the goal of iSpreadRank is to re-weight similar sentences with similar degree of importance, and hence they are ranked in close positions in the reasoned ranking.

(1) Initialization:

Let $G = (V, E)$ represent the sentence similarity network with the set of nodes $V = \{s_1, \dots, s_m\}$ and the set of edges E , where s_i denotes a sentence, and E is a subset of $V \times V$. For simplicity, every node with no edges connecting it to other nodes is eliminated from G . Such a weighted graph representation of the input document group can be transformed into an adjacency matrix, A , with rows and columns labeled by sentence nodes, and each entry a_{ij} initialized by Eq. (3.4).

Notably, A is a symmetric matrix since G is an undirected graph.

$$a_{ij} = a_{ji} = \begin{cases} 0 & \text{if } i = j \\ \text{sim}(s_i, s_j) & \text{if } i \neq j \end{cases} \quad (3.4)$$

In Eq. (3.4), $\text{sim}(s_i, s_j)$, as defined in Eq. (3.3), indicates the similarity between a pair of sentences s_i and s_j and $\text{sim}(s_i, s_j) \geq \alpha$. Note that α is the similarity threshold mentioned in Section 3.2.1.

(2) Inference:

Each node in the network has an activation level¹⁴. The algorithm iteratively updates the activations of all nodes (i.e., sentences) over discrete time until it is stopped by the user, or a termination condition is triggered. In one iteration, each node obtains a new activation level by collecting the activations from its connected nodes, and then propagates the new activation along links to its neighbors as a function of its current activation and the strength of relationships between nodes.

The iteration itself can be mathematically defined in a simple linear algebra formula. Let X represent an m -dimensional vector to capture the activations of nodes $\{s_1, \dots, s_m\}$ in the network. A particular vector, $X(0)$, is the activation vector at the initial step where the activation of each sentence node is initialized as its sentence-specific feature score computed by feature extraction. At iteration t , the algorithm maintains the activation vector $X(t)$ using Eq. (3.5).

$$X(t) = X(0) + MX(t-1), \quad M = \sigma R^T \quad (3.5)$$

¹⁴ The term “activation” in this chapter is interchangeable with the term “importance.” It is used here in order to follow the terminology of spreading activation.

In Eq. (3.5), σ ($0 \leq \sigma < 1$) is a spreading factor determining the propagation efficiency that a node converts the activations from its neighbors to its own activation (i.e., the level of activation propagated from a node's neighbors to the node). It is assigned to 0.7 heuristically in the current implementation. The matrix R is obtained from A by Eq. (3.6). Since the initialization step removes nodes with no edges, R is a stochastic matrix (i.e., for each row i in R , $\sum_j r_{ij} = 1$).

$$r_{ij} = \frac{a_{ij}}{\sum_k a_{ik}} \quad (3.6)$$

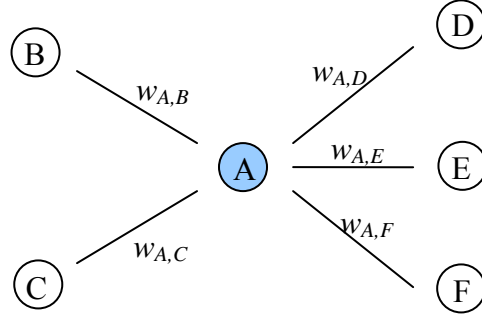
The algorithm iterates until a stable equilibrium of the network is obtained (i.e., the convergence state is reached). Practically, a stopping condition is used to judge the convergence of the algorithm and to terminate the iterations. In this study, each iteration is followed by a checkpoint to determine whether the criterion in Eq. (3.7) is satisfied.

$$\sum_i |X_i(t) - X_i(t-1)| \leq \varepsilon \quad (3.7)$$

In Eq. (3.7), $X_i(t)$ refers to the activation of node i at step t , and ε is a negligible number, which is set to 0.0001 heuristically.

Theoretically, Eq. (3.7) measures the L_1 -norm of the residual vector: $X(t) - X(t-1)$. The algorithm terminates at iteration t when the sum of changes of the activations for all nodes with respect to prior iteration $t-1$ is not greater than a predefined threshold ε .

As an example, Fig. 3.3 illustrates the collection and the spread of activations for node A in one iteration.



Step (a): Collecting the activations from neighbors

$$\begin{aligned}
 X_A(t) = & X_A(0) + \sigma \frac{w_{B,A}}{\sum_K w_{B,K}} X_B(t-1) + \sigma \frac{w_{C,A}}{\sum_K w_{C,K}} X_C(t-1) \\
 & + \sigma \frac{w_{D,A}}{\sum_K w_{D,K}} X_D(t-1) + \sigma \frac{w_{E,A}}{\sum_K w_{E,K}} X_E(t-1) + \sigma \frac{w_{F,A}}{\sum_K w_{F,K}} X_F(t-1)
 \end{aligned}$$

Step (b): Spreading the current activation to neighbors

$$\begin{aligned}
 X_B(t+1) = & X_B(0) + \sigma \frac{w_{A,B}}{\sum_K w_{A,K}} X_A(t) + \dots \\
 X_C(t+1) = & X_C(0) + \sigma \frac{w_{A,C}}{\sum_K w_{A,K}} X_A(t) + \dots \\
 X_D(t+1) = & X_D(0) + \sigma \frac{w_{A,D}}{\sum_K w_{A,K}} X_A(t) + \dots \\
 X_E(t+1) = & X_E(0) + \sigma \frac{w_{A,E}}{\sum_K w_{A,K}} X_A(t) + \dots \\
 X_F(t+1) = & X_F(0) + \sigma \frac{w_{A,F}}{\sum_K w_{A,K}} X_A(t) + \dots
 \end{aligned}$$

Fig. 3.3. The collection and the spread of activations for node A in one iteration

(3) Prediction:

When iSpreadRank ends, the network is in a *stable* state with each node labeled with a numeric weight as its final degree of importance. iSpreadRank outputs a ranking of sentences according to the importance of all sentences inferred in the inference step. (N.B., for those sentences without connections to other sentences, their initial feature scores are used for ranking.)

3.2.3.2 The convergence of iSpreadRank

The convergence of iSpreadRank is proven via Proposition 1. It is guaranteed that there is a t since $(I - \sigma R^T)^{-1} X(0)$ does exist. By Proposition 1, it is proven that for such a t , Eq. (3.7) is satisfied (and iSpreadRank terminates) and iSpreadRank converges at the t -th iteration. The detail proof steps of Proposition 1 are given in the following:

Proposition 1. For some t , $t > 0$,

- (a) $\sum_i |X_i(t) - X_i(t-1)| \leq \varepsilon$. \Leftrightarrow (b) iSpreadRank converges at the t -th iteration.
 (b) iSpreadRank converges at the t -th iteration. \Leftrightarrow (c) $X(t) \approx (I - \sigma R^T)^{-1} X(0)$.
 (a) $\sum_i |X_i(t) - X_i(t-1)| \leq \varepsilon$. \Leftrightarrow (c) $X(t) \approx (I - \sigma R^T)^{-1} X(0)$.

I: (a) \Rightarrow (b).

Proof. Consider $X(t+1)$ and $X(t)$. According to Eq. (3.5), the following equations hold.

$$(I.1): X(t+1) = X(0) + \sigma R^T X(t)$$

$$(I.2): X(t) = X(0) + \sigma R^T X(t-1)$$

Since $\sum_i |X_i(t) - X_i(t-1)| \leq \varepsilon$ and ε is negligible, assume $X(t) = X(t-1)$. By replacing $X(t)$ in Equation (I.1) with $X(t-1)$, Equation (I.3) is obtained.

$$(I.3): X(t+1) = X(0) + \sigma R^T X(t-1)$$

From Equation (I.2) and Equation (I.3), $X(t+1) = X(t)$.

By induction, it is easily verified that

$$\forall t', t' = t + c \text{ and } c \geq 0, X(t') = X(t'-1) \text{ holds.}$$

Hence, iSpreadRank converges at the t -th iteration.

II: (b) \Rightarrow (a).

Proof. Since iSpreadRank converges at the t -th iteration,

$$\forall t', t' = t + c \text{ and } c \geq 0, X(t') \approx X(t'-1) \text{ holds.}$$

Then, $\sum_i |X_i(t') - X_i(t'-1)| \leq \varepsilon$.

III: (a) \Leftrightarrow (b).

Proof. From I: (a) \Rightarrow (b) and II: (b) \Rightarrow (a), it is proven.

IV: (b) \Rightarrow (c).

Proof. Since iSpreadRank converges at the t -th iteration, assume $X(t) = X(t-1)$. By replacing $X(t-1)$ in Eq. (3.5) with $X(t)$, it is easily verified that

$$(I - \sigma R^T)X(t) = X(0).$$

Let $P = I - \sigma R^T$, $P^T = I - \sigma R$. Since R is a stochastic matrix and its diagonals are all 0s, and $0 \leq \sigma < 1$, P^T is a strictly diagonally dominant matrix. The Gerschgorin circle theorem [102] assures that the inverse of P^T exists. Since $P^T = I - \sigma R$ is invertible, $P = I - \sigma R^T$ is also invertible and hence $X(t) = (I - \sigma R^T)^{-1} X(0)$.

V: (c) \Rightarrow (b).

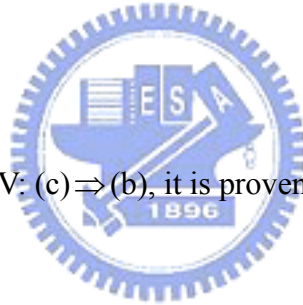
Proof. Suppose iSpreadRank does not converge at the t -th iteration and assume $X(t) \not\approx X(t-1)$. Similarly, by Eq. (3.5), it is easily verified that

$$(I - \sigma R^T)X(t) \not\approx X(0).$$

As in IV: (b) \Rightarrow (c), $P = I - \sigma R^T$ is invertible and hence $X(t) \not\approx (I - \sigma R^T)^{-1} X(0)$, which is contradictory to the given $X(t) \approx (I - \sigma R^T)^{-1} X(0)$. Therefore, iSpreadRank converges at the t -th iteration.

VI: (b) \Leftrightarrow (c).

Proof. From IV: (b) \Rightarrow (c) and V: (c) \Rightarrow (b), it is proven.



VII: (a) \Leftrightarrow (c).

Proof. From III: (a) \Leftrightarrow (b) and VI: (b) \Leftrightarrow (c), it is proven.

3.2.3.3 Example

Fig. 3.4 illustrates how iSpreadRank works to re-weight the importance of sentences.

Fig. 3.4 (a) displays the initial state of the network before iSpreadRank is applied. The initial sentence ranking is: $Rank(S_2) = Rank(S_3) = Rank(S_4) > Rank(S_1)$. Given this network, iSpreadRank runs and terminates at the converged state, as depicted in Fig. 3.4 (b), and outputs a new sentence ranking: $Rank(S_2) = Rank(S_3) > Rank(S_1) > Rank(S_4)$. It can be seen that S_1 is promoted to the position before S_4 in the new ranking.

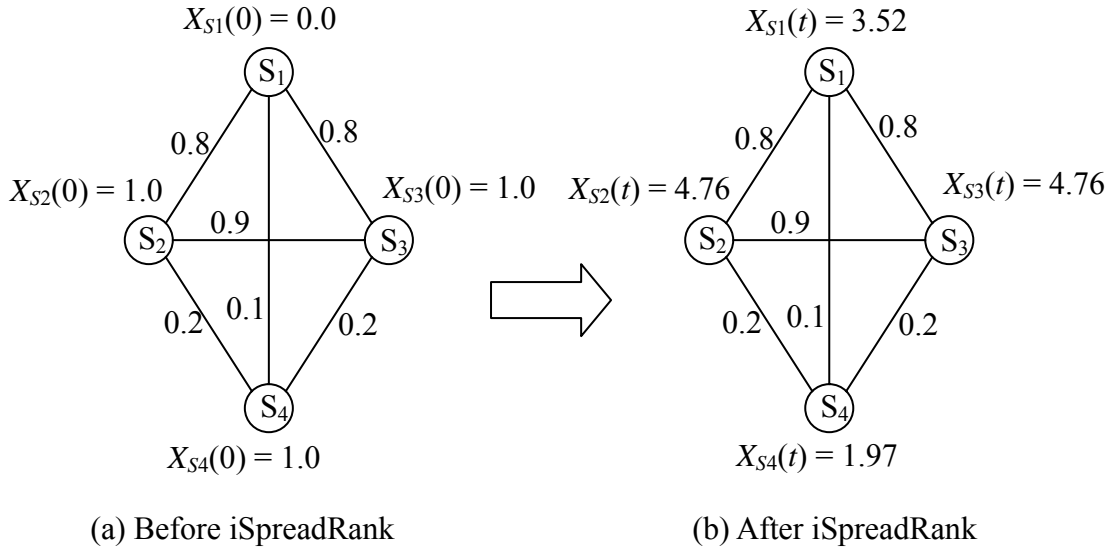


Fig. 3.4. An example to explain how iSpreadRank works, with the spreading factor $\sigma = 0.8$. (a) the initial state of the network before iSpreadRank is applied; (b) the converged state when iSpreadRank terminates at iteration t

As an example, the detail of computation for $X(1)$ is given in Fig. 3.5.

$$A = \begin{bmatrix} 0.0 & 0.8 & 0.8 & 0.1 \\ 0.8 & 0.0 & 0.9 & 0.2 \\ 0.8 & 0.9 & 0.0 & 0.2 \\ 0.1 & 0.2 & 0.2 & 0.0 \end{bmatrix}, \quad R = \begin{bmatrix} \frac{0.0}{1.7} & \frac{0.8}{1.7} & \frac{0.8}{1.7} & \frac{0.1}{1.7} \\ \frac{0.8}{0.8} & \frac{0.0}{0.9} & \frac{0.9}{0.9} & \frac{0.2}{0.2} \\ \frac{1.9}{0.8} & \frac{1.9}{0.9} & \frac{1.9}{0.0} & \frac{1.9}{0.2} \\ \frac{1.9}{0.1} & \frac{1.9}{0.2} & \frac{1.9}{0.2} & \frac{1.9}{0.0} \\ \frac{0.5}{0.5} & \frac{0.5}{0.5} & \frac{0.5}{0.5} & \frac{0.5}{0.5} \end{bmatrix}, \quad X(0) = \begin{bmatrix} 0.0 \\ 1.0 \\ 1.0 \\ 1.0 \end{bmatrix}$$

Set $\sigma = 0.8$

$$X(1) = \begin{bmatrix} 0.0 \\ 1.0 \\ 1.0 \\ 1.0 \end{bmatrix} + 0.8 \times \begin{bmatrix} \frac{0.0}{1.7} & \frac{0.8}{1.9} & \frac{0.8}{1.9} & \frac{0.1}{0.5} \\ \frac{0.8}{0.8} & \frac{0.0}{0.9} & \frac{0.9}{1.9} & \frac{0.2}{0.5} \\ \frac{1.9}{0.8} & \frac{0.9}{0.9} & \frac{0.0}{0.5} & \frac{0.2}{0.5} \\ \frac{1.9}{0.1} & \frac{1.9}{0.2} & \frac{1.9}{0.2} & \frac{1.9}{0.0} \\ \frac{0.5}{1.7} & \frac{0.5}{1.9} & \frac{0.5}{1.9} & \frac{0.5}{0.0} \end{bmatrix} \times \begin{bmatrix} 0.0 \\ 1.0 \\ 1.0 \\ 1.0 \end{bmatrix} = \begin{bmatrix} 0.8337 \\ 1.6989 \\ 1.6989 \\ 1.1684 \end{bmatrix}$$

Fig. 3.5. The computation of $X(1)$

Table 3.2 presents the weights of the inferred importance of S_i at different iterations. According to this table, the weight of S_1 raises more rapidly than the weight

of S_4 during the inference iterations. This is because S_1 is strongly related to S_2 and S_3 , and therefore it receives more weights distributed from them. In contrast, S_2 and S_3 propagate fewer weights to S_4 since S_4 has weak connections with S_2 and S_3 . Consequently, S_1 obtains a new weight, $X_{S_1}(t) = 3.5193$, which is much larger than the new weight of S_4 , $X_{S_4}(t) = 1.9667$. Furthermore, S_1 , S_2 , and S_3 together form a feedback loop, giving them the highest weights in the end.

Table 3.2. The inferred weights of S_i at different iterations ($\sigma = 0.8$)

Iteration	S_1	S_2	S_3	S_4
0	0.0000	1.0000	1.0000	1.0000
1	0.8337	1.6989	1.6989	1.1684
5	2.4058	3.5114	3.5114	1.6392
10	3.1543	4.3489	4.3489	1.8594
20	3.4802	4.7131	4.7131	1.9552
\approx Convergence	3.5193	4.7568	4.7568	1.9667

3.2.4 Sentence extraction

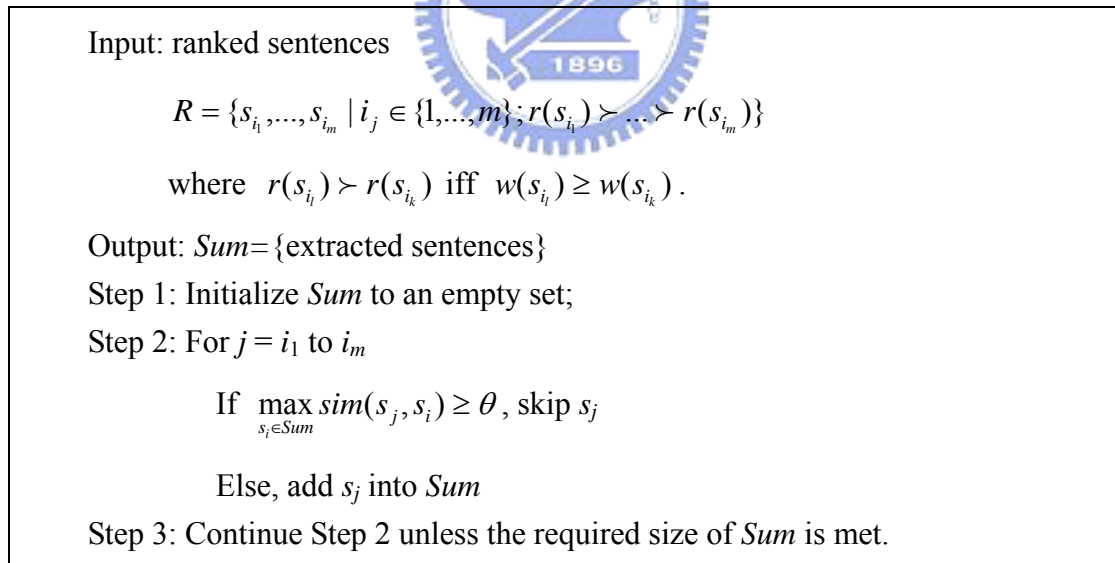


Fig. 3.6. The algorithm of sentence extraction

The sentence extraction method, based on cross-sentence informational subsumption [111] for redundancy filtering, iteratively examines sentences in the rank order, and adds one sentence at a time into the summary, if it is not too similar to any sentences

already in the summary. Here, the degree of redundancy between two sentences is determined by a threshold imposed on the sentence similarity. In this way, only high-scoring sentences with less redundant information than others are extracted into the summary. Fig. 3.6 shows the algorithm of sentence extraction where $r(s_i) \succ r(s_{i_k})$ means the position of s_i in the ranking is prior to that of s_{i_k} , $w(s_i)$ is the weight of s_i , and $sim(s_j, s_i)$ stands for the similarity between s_j and s_i . In the current implement, θ is heuristically set to 0.7.

3.2.5 Sentence ordering

Fig. 3.7 provides the algorithm of sentence ordering. The final summary is structured in the following steps: Semi-similar sentences in the extracted sentence set are first grouped together, based on another similarity threshold smaller than that used in sentence extraction. Each group is then ordered chronologically into a macro-ordering according to the earliest timestamp of the sentences within it. Finally, micro-ordering is applied to sort all sentences in each group in chronological order. This policy, considering together topical relatedness and the chronological order between sentences, is similar to the augmented sentence ordering algorithm proposed in [8], in which the topical relatedness between sentences is determined by text cohesion in their original documents. In the current implementation, δ is set to 0.5.

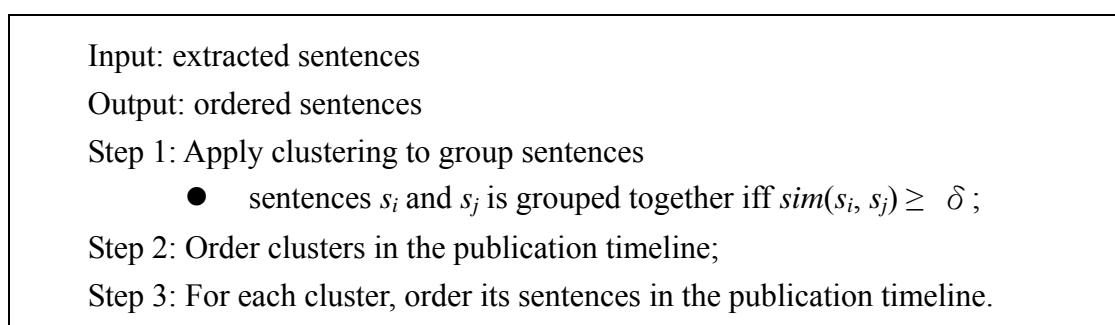


Fig. 3.7. The algorithm of sentence ordering

3.3 Evaluation

This section describes the data set, evaluation method, and the experimental results.

3.3.1 Data set and experimental setup

The DUC 2004 data set from DUC (Document Understanding Conferences) [34] was tested to examine the effectiveness of the proposed summarization method. The guideline of Task 2 at DUC 2004 was followed to produce generic extractive summaries. In the task, the goal is to generate a short summary of roughly 665 bytes in length to provide the condensed essentials of an input group of topically-related news articles.

The total number of document groups is 50. Each group contains 10 news articles on average. The documents came from the AP newswire and New York Times newswire. For each group, 4 NIST assessors were each asked to read all the documents and to create a brief summary. The assessors can either select representative sentences from documents to form an extractive summary or write an abstract using their own words. The manually-generated summaries are treated as gold-standard summaries to evaluate the qualities of machine-generated summaries.

It should be clear that in the process of creating machine-generated and manual summaries, the associated topic descriptions of the document groups are not given as input. Thus, the summaries are not created to be focused in any particular way, but to represent all the content of the document group to some degree for understanding.

3.3.2 Evaluation method and metric

The machine-generated summaries were evaluated using ROUGE (Recall-Oriented

Understudy for Gisting Evaluation, alias RED) automatic n -gram matching [79]. ROUGE¹⁵ is a recall-oriented scoring metric for fix-length summaries, which adopts ideas from BLEU (BiLingual Evaluation Understudy) [104] to determine the quality of a summary. It generally counts as a performance indicator the number of co-occurrences between machine-generated and ideal summaries in different word units, such as n -gram, word sequences and word pairs.

Following the guideline to apply ROUGE for evaluation, all machine-generated summaries need to be truncated before evaluation, if the summary length is beyond the target length. Hence, we produced summaries with exactly 665 bytes, in order to have a fair evaluation. Model summaries (i.e., manual summaries) were not truncated before evaluation, but in another way, the length control option was set to truncate them when running ROUGE. Jackknifing was implemented so that human and system scores can be compared. ROUGE v.1.2.1 was used and the runtime arguments of ROUGE for evaluation are listed in Table 3.3.

Table 3.3. ROUGE runtime arguments for DUC 2004

```
ROUGE -a -c 95 -b 665 -m -n 4 -w 1.2
```

- a: Evaluate all systems
- c 95: Calculate 95% confidence intervals
- b 665: Truncate model and peer summaries at 665 bytes
- m: Stem models and peers using Porter's stemming algorithm
- n 4: Calculate ROUGE-1, ROUGE-2, ROUGE-3, ROUGE-4
- w 1.2: Use 1.2 as the weighting factor for LCS-W
- Do not drop stop words

The basic official ROUGE scores at DUC 2004 are the 1-gram, 2-gram, 3-gram, 4-gram, and *longest common substring* scores. The 1-gram ROUGE score (a.k.a. ROUGE-1) has been found to correlate very well with human judgments at a

¹⁵ Note that a scoring method, such as ROUGE, based on matching of n -grams between the system output and the ideal summary is promising but not sufficient [79].

confidence level of 95%, based on various statistical metrics [79]. Therefore, in this section, we only report the ROUGE-1 scores.

3.3.3 Results

Table 3.4 lists the ROUGE-1 scores of different experiments and their 95% confidence intervals in brackets. *Feature* denotes which sentence-specific feature is used to estimate the importance of every sentence. *Without-iSpreadRank* scores sentences only by features, while *With-iSpreadRank* applies the proposed iSpreadRank for sentence ranking. *Improvement* refers to the difference between the ROUGE-1 scores and the relative improvement in the parentheses when With-iSpreadRank is compared to Without-iSpreadRank. The relative improvement here is calculated as $(b - a)/a \times 100$ when b is compared to a . In addition, in the bottom of Table 3.4, two baselines are presented as well for comparison purposes. *Random Baseline* randomly extracts sentences from the input document group. The reported result is averaged from 10 random runs. *NIST Baseline*, the official baseline at DUC 2004, simply outputs the first 665 bytes of the most recent document.

Several interesting results are found. First, With-iSpreadRank performs significantly better than the two baselines. Second, With-iSpreadRank is superior to Without-iSpreadRank, which demonstrates that the use of sentence-specific features in iSpreadRank is an effective sentence ranking method. The average improvement is observed to decrease when the initial importance of sentences is determined by more features. The average improvement is 3.21% when only one feature is used, becoming 2.23% when employing two features, 1.97% when all features are examined. This phenomenon merits further investigation. Third, a particular experiment (see Feature: $EV=1$) was conducted in which iSpreadRank initially assigned every sentence an

equal feature score of 1.0. In this case, iSpreadRank depends much on the relationships between sentences, and hence ranks sentences, similar to many other sentences, in higher positions. As expected, this model is inferior to other models where *real* sentence-specific features are considered. This result reflects to the hypothesis of iSpreadRank that the importance of a sentence in the network is related to the following factors: (1) the number of sentences to which it connects; (2) the importance of its connected sentences, and (3) the strength of relationships between it and its connected sentences.

Table 3.4. ROUGE-1 scores obtained in different experimental settings

Feature	Without-iSpreadRank	With-iSpreadRank ($\sigma = 0.7$)	Improvement
EV=1	–	0.36218 [0.34611, 0.37825]	–
Centroid (C)	0.35033 [0.33354, 0.36712]	0.36722 [0.35308, 0.38136]	+0.0169 (4.82%)
Position (P)	0.36524 [0.35290, 0.37758]	0.37756 [0.36324, 0.39188]	+0.0123 (3.37%)
SimWithFirst (SF)	0.36524 [0.35290, 0.37758]	0.37052 [0.35903, 0.38201]	+0.0053 (1.45%)
C+P	0.36974 [0.35807, 0.38141]	0.37701 [0.36429, 0.38973]	+0.0073 (1.97%)
C+SF	0.36923 [0.35747, 0.38099]	0.37821 [0.36551, 0.39091]	+0.0090 (2.44%)
P+SF	0.36524 [0.35290, 0.37758]	0.37355 [0.36063, 0.38647]	+0.0083 (2.27%)
C+P+SF	0.37333 [0.36182, 0.38484]	0.38068 [0.36804, 0.39332]	+0.0074 (1.97%)
Random Baseline: 0.31549 [0.30332, 0.32766]			
NIST Baseline: 0.32419 [0.30922, 0.33916]			

Table 3.5 shows the official ROUGE-1 scores of human assessors and the top 5 systems for Task 2 at DUC 2004. In this table, *SYSID* signifies the peer codes of participants: letters stand for human assessors, and numbers represent machine systems. The scores indicate, at the 95% confidence level, that With-iSpreadRank does not outperform the best machine (*SYSID*: 65) in any settings. However, four of

them performed better than the second best system (SYSID: 104), namely (1) With-iSpreadRank (C+P+SF), (2) With-iSpreadRank (C+SF), (3) With-iSpreadRank (C+P), and (4) With-iSpreadRank (P). Overall, the proposed summarization method is found to perform well. Our best model, With-iSpreadRank (C+P+SF) has a ROUGE-1 score of 0.38068, which is competitive to that of the 1st-ranked system (i.e., SYSID: 65) at DUC 2004. Fig. 3.8 to Fig. 3.13 show the different ROUGE scores of system and human peers at DUC 2004.

Table 3.5. Part of the official ROUGE-1 scores of Task 2 at DUC 2004

SYSID	ROUGE-1	95% Confidence Interval
H	0.41828	[0.40193, 0.43463]
F	0.41246	[0.39161, 0.43331]
E	0.41038	[0.38817, 0.43259]
D	0.40594	[0.38700, 0.42488]
B	0.40428	[0.37946, 0.42910]
A	0.39325	[0.37218, 0.41432]
C	0.39039	[0.37149, 0.40929]
G	0.38902	[0.36793, 0.41011]
65	0.38224	[0.36941, 0.39507]
104	0.37443	[0.36354, 0.38532]
35	0.37430	[0.36121, 0.38739]
19	0.37386	[0.36080, 0.38692]
124	0.37064	[0.35782, 0.38346]
2 (NIST Baseline) (Rank: 25/35)	0.32419	[0.30922, 0.33916]
Best machine (SYSID = 65)	0.38224	[0.36941, 0.39507]
Median machine (SYSID = 138)	0.34299	[0.32805, 0.35793]
Worst machine (SYSID = 111)	0.24190	[0.23038, 0.25342]
Avg. of human assessors	0.40300	[0.38247, 0.42353]

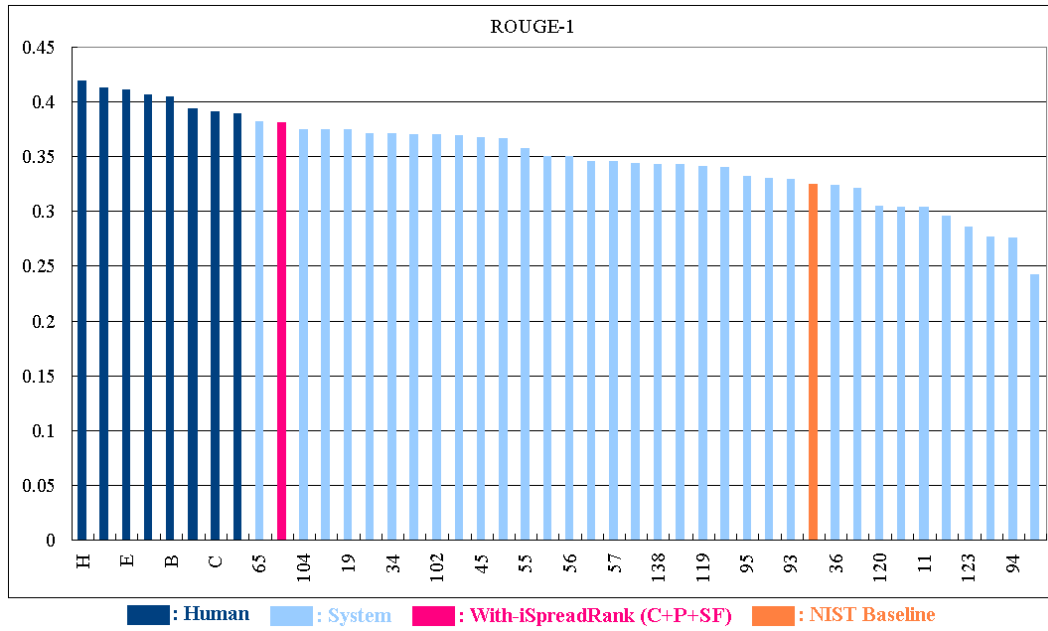


Fig. 3.8. ROUGE-1 scores of system and human peers at DUC 2004

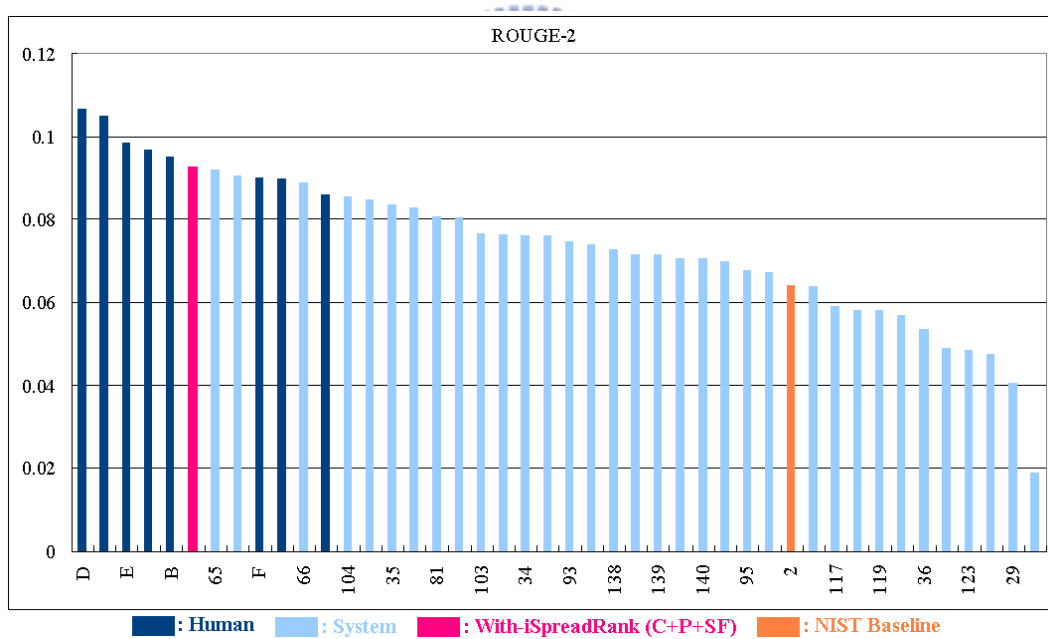


Fig. 3.9. ROUGE-2 scores of system and human peers at DUC 2004

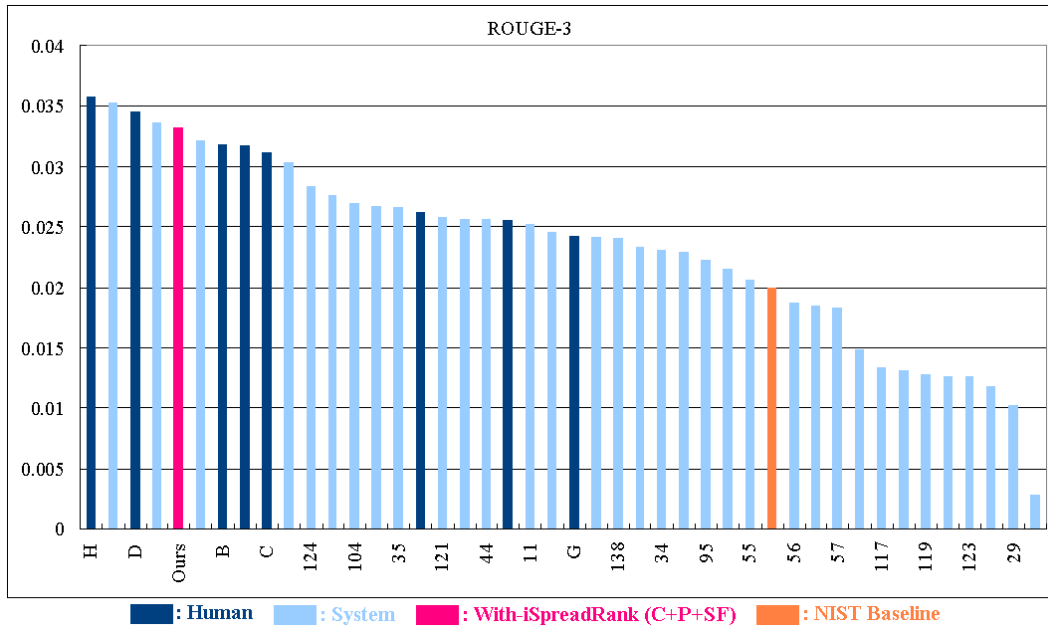


Fig. 3.10. ROUGE-3 scores of system and human peers at DUC 2004

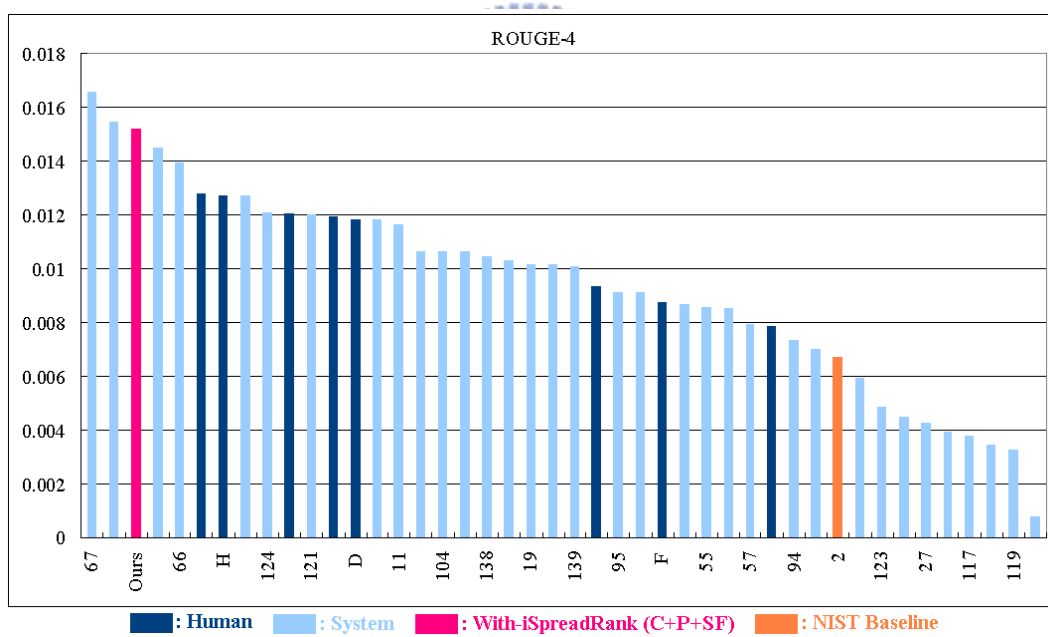


Fig. 3.11. ROUGE-4 scores of system and human peers at DUC 2004

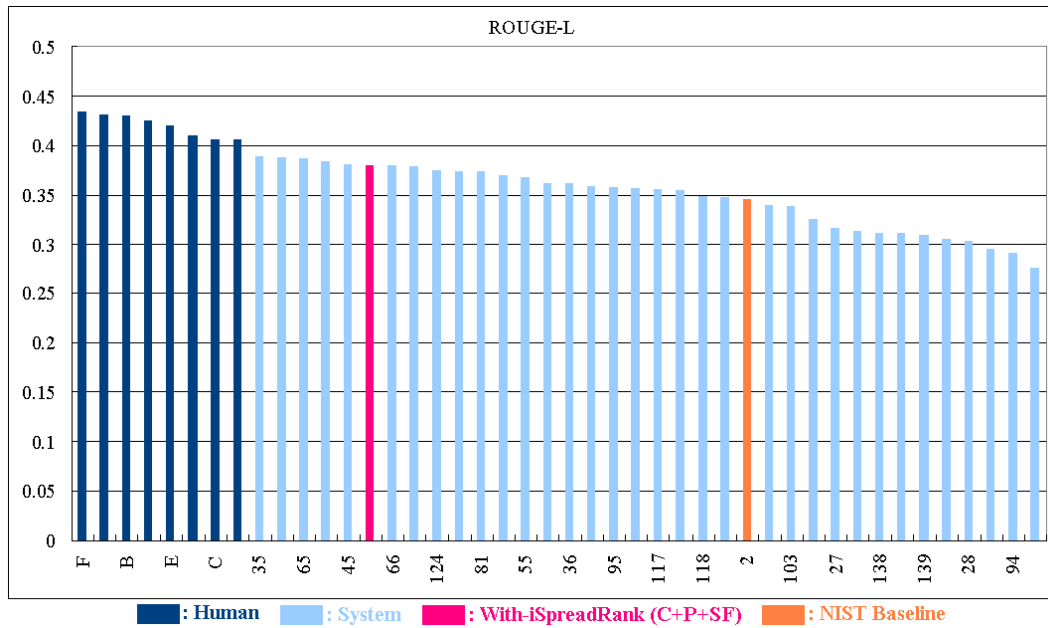


Fig. 3.12. ROUGE-L scores of system and human peers at DUC 2004

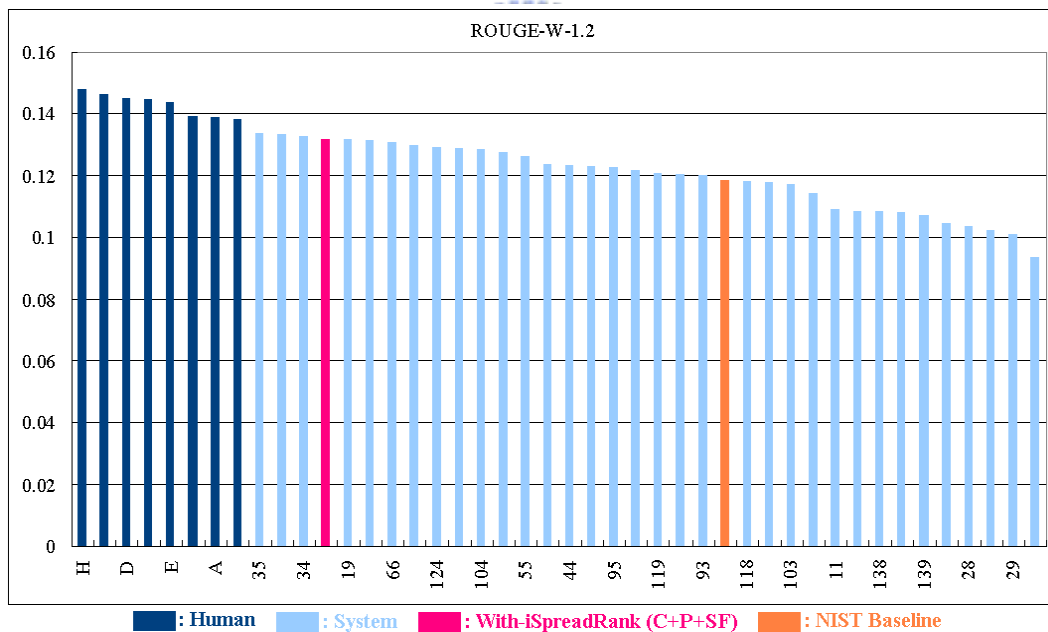


Fig. 3.13. ROUGE-W-1.2 scores of system and human peers at DUC 2004

3.3.4 Example output

Fig. 3.14 provides the ROUGE-1 scores of our best model, With-iSpreadRank (C+P+SF), for 50 clusters. The best ROUGE-1 score is 0.46211 for set d30045t, the median ROUGE-1 score is 0.38293 for set d30040t, and the worst ROUGE-1 score is

0.25670 for set d30027t.

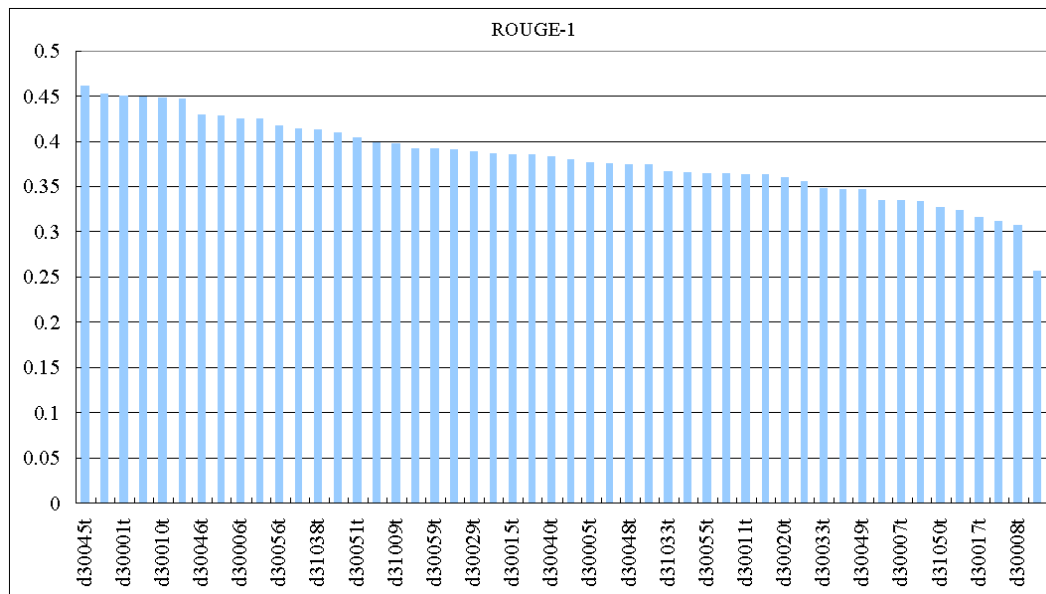


Fig. 3.14. ROUGE-1 scores of With-iSpreadRank (C+P+SF) for 50 clusters

Fig. 3.15 shows the system output for set d30045t, a set about the merger of Exxon Corp. and Mobil Corp., and Fig. 3.16 to Fig. 3.19 provide the human summaries. In this example, we obtained a *good* ROUGE-1 score of 0.46211: when compared to model summaries (C, E, F), the ROUGE-1 score is 0.43658; when compared to model summaries (B, E, F), the ROUGE-1 score is 0.47289; when compared to model summaries (B, C, F), the ROUGE-1 score is 0.45672; when compared to model summaries (B, C, E), the ROUGE-1 score is 0.48225.

System Summary (~665 bytes)

The boards of Exxon Corp. and Mobil Corp. are expected to meet Tuesday to consider a possible merger agreement that would form the world's largest oil company, a source close to the negotiations said Friday. Exxon and Mobil, the nation's two largest oil companies, confirmed Friday that they were discussing a possible merger, and antitrust lawyers, industry analysts and government officials predicted that any deal would require the sale of important large pieces of such a new corporate behemoth. Oil stocks led the way as investors soaked up the news of continuing talks between Exxon and Mobil on a merger that would create the world's largest oil company. Whe

Fig. 3.15. System summary for d30045t

Model Summary: B

Exxon and Mobil discuss combining business operations. A possible Exxon-Mobil merger would reunite 2 parts of Standard Oil broken up by the Supreme Court in 1911. Low crude oil prices and the high cost of exploration are motives for a merger that would create the world's largest oil company. As Exxon-Mobil merger talks continue, stocks of both companies surge. The merger talks show that corporate mergers are back in vogue. Antitrust lawyers, industry analysts, and government officials say a merger would require divestitures. A Mobil employee worries that a merger would put thousands out of work, but notes that his company's stock would go up.

Fig. 3.16. Model summary, created by B, for d30045t

Model Summary: C

In a move considered unthinkable a few years ago, Exxon Corp. and Mobile Corp, have entered into negotiations which could result in a merger of the two companies. Such a merger, should it occur, would form the world's largest oil company and the largest U.S. company, placing it above Wal-Mart. The merger, and talks like it among other oil companies, is being prompted By low petroleum prices and high production costs. Talks of a merger have sent the price of stocks of both companies soaring. The merger could prompt anti-trust action and the merging companies would have to divest themselves of some interests. Mobile workers fear a merger will cost them jobs.

Fig. 3.17. Model summary, created by C, for d30045t

Model Summary: E

Exxon Corp and Mobil Corp are reported to be discussing a business merger. Other oil companies have merged to compensate for low oil prices and increasing costs of oil exploration in more remote areas. The mergers are consistent with a trend in corporate marriages that is changing U.S. economic history. The mergers are pushing stocks up and the Exxon-Mobil merger could benefit consumers and lead to further savings. Some people believe the merger would require selling of large corporate pieces and put thousands out of work. If the companies merge, it would be the largest U.S. company and bigger than the world's largest oil company, Royal Dutch/Shell Group.

Fig. 3.18. Model summary, created by E, for d30045t

Model Summary: F

Exxon and Mobil consider merger. Partnerships already formed. Oil prices are lowest in 12 years and future exploration will be costly. The new company would be largest in the US and put back together pieces of Standard Oil, a monopoly broken up by courts. Experts mixed on merger's advantages. It would be an anti-trust test, since companies are involved in many facets of the business, require the sale of large units. Refinery workers, others would lose jobs. There is an upswing in corporate mergers, pushed by bull market and recognition that it's hard to increase revenue internally. Merger anticipation sent stocks higher in oil, internet and computers.

Fig. 3.19. Model summary, created by F, for d30045t

Fig. 3.20 shows the system output for set d30027t, a set about the worldwide financial crises in 1998, and Fig. 3.21 to Fig. 3.24 provide the human summaries. In this example, we obtained a *bad* ROUGE-1 score of 0.25670: when compared to model summaries (C, E, G), the ROUGE-1 score is 0.25240; when compared to model summaries (A, E, G), the ROUGE-1 score is 0.24667; when compared to model summaries (A, C, G), the ROUGE-1 score is 0.26129; when compared to model summaries (A, C, E), the ROUGE-1 score is 0.26645.

System Summary (~665 bytes)

I want to repeat once more _ there is no program," Prime Minister Yevgeny Primakov said. It is worth noting," Fischer said this week, that our programs in Asia _ in Indonesia, Korea and Thailand _ only took hold after there was a change in government." If the Communist Party has its way _ and it has been planning for months _ millions of Russians will take to the streets on Wednesday for some of the biggest demonstrations in years. When the world's finance ministers and central bankers gathered last year in Hong Kong, they nervously congratulated each other for containing _ at least for the moment _ a nasty financial brush fire in Asia. I have no doubt t

Fig. 3.20. System summary for d30027t

Model Summary: A

In October 1998 amid worldwide financial crises, particular concern focused on Russia where economic meltdown was exacerbated by conflicted politics. President Yeltsin's latest Prime Minister, Primakov, was supported by Communists and when word leaked out that a Communist economic program was under consideration, Yeltsin denounced it. Primakov then assured the public that "there is no program," suggesting that there would not be until the International Monetary Fund (IMF) came forth with a massive loan. IMF demanded a sound economic program before approving loan payment. Meanwhile neighboring Ukraine felt economic effects of the IMF-Primakov standoff.

Fig. 3.21. Model summary, created by A, for d30027t

Model Summary: C

As world finance and banking representatives met in Washington, the economic news continued to be bleak. IMF officials had predicted a banner year, but stocks continued to slide worldwide and the DOW probably would record its worst third quarter loss in eight years. Russia and Ukraine have been especially hard hit by the crisis. In Russia, Prime Minister Primakov had no plan to solve the problem as the economy continued to suffer. Postal service was threatened as the Post Office could not pay its bills. President Kuchma of Ukraine called for changes in market reform even as the Parliament turned down a bill to restore lost savings.

Fig. 3.22. Model summary, created by C, for d30027t

Model Summary: E

Fifteen months of world economic turmoil are threatening political stability. Lowering Federal Reserve interest rates is not countering the crisis. IMF is worried about the turndown in Japan, economic meltdown in Russia, depression in Indonesia, and anxiety about Latin America where investors are pulling out. IMF critics say it needs to understand national politics better and focus on social issues. Russia's economic confusion is upsetting the US. Russia is considering hard currency controls, demanding IMF loans and will not end government privatization. Ukraine, affected by Russia, is trying to save its fast-developing money system and keep investors.

Fig. 3.23. Model summary, created by E, for d30027t

Model Summary: G

Early October was fraught with economic woes as the International Monetary Fund prepared for its annual meeting. The IMF faces criticism for ignoring the social costs of its actions and being a pawn to Western demands. A small cut in US interest rates lowered markets worldwide. Russia, whose economy collapsed in August, was looking for a cure--possibly instituting Soviet-style measures. Key issues were stopping dollars from leaving the country and getting foreign investment end IMF loans. The postal service was in chaos, owing everyone. Demonstrations were expected. The Ukraine also struggled, especially to keep banks working. An IMF loan was on the way.

Fig. 3.24. Model summary, created by G, for d30027t

3.4 Discussion

This section provides general discussion on the proposed summarization approach.

3.4.1 Sentence similarity network

One problem of a sentence similarity network constructed using the cosine similarity is the lack of type or context in a link [119]. Fortunately, this problem could be alleviated by considering semantic-level text analysis when defining the similarity between text units (see [53], [99], [137]). In [137], they found that the similarity computed by latent semantic analysis improves the performance of degree-centrality-based single-document summarization. According to their observations, we presume that the improvement of relationships between sentences might directly profit iSpreadRank. This issue is left to future work.

3.4.2 The use of sentence-specific features

With the use of sentence-specific features, iSpreadRank operates like a learning process in which the initial labeling of every sentence is determined according to its feature score, and the final labeling of sentences is achieved based on the feature scores of sentences and the relationships between sentences. In this study, we had

tested three features: *centroid*, *position*, and *first-sentence overlap*, as well as various combinations of them, to understand how they affect the performance of iSpreadRank.

The evaluation results, as shown in Table 3.4, roughly give the idea that the performance is improved when sentence-specific features are considered. Still, it is worth studying to discover more features that are advantageous to iSpreadRank. This issue is left as an open question, since to examine the whole feature space is not straightforward.

3.4.3 iSpreadRank

Recall that iSpreadRank applies a particular model of spreading activation, namely Leaky Capacitor Model (LCM) [4]. The original LCM formulates the flow of activations of all the nodes over time by Eq. (3.8)¹⁶.

$$X(t) = C + MX(t-1), \quad M = [(1-\gamma)I + \sigma R^T] \quad (3.8)$$

where C indicates a vector capturing the set of energized nodes and their activations at iteration t , M represents a matrix to manage the flow and the decay of activation among nodes, $\gamma \in [0,1]$ determines the relaxation of node activation, I denotes the identity matrix, and σ and R are the same as in Eq. (3.5).

Obviously, iSpreadRank is a derivative of LCM since it simply fixes $C = X(0)$ and $\gamma = 1$ in all iterations. However, iSpreadRank is very different from LCM in terms of its goal and how it is achieved. In general, LCM only activates a *subset* of nodes in each iteration; iSpreadRank, in contrast, propagates the activations of *all*

¹⁶ This matrix calculus is excerpted from [105] with adaptations in correspondence to the terminology used in this study.

nodes into the network (i.e., all nodes are activated). Furthermore, while LCM is designed to identify hidden nodes related to the activated source nodes according to some criterion, the goal of iSpreadRank is to assess the relative importance of all nodes.

In the following, iSpreadRank is discussed in different aspects.

(1) Spreading factor

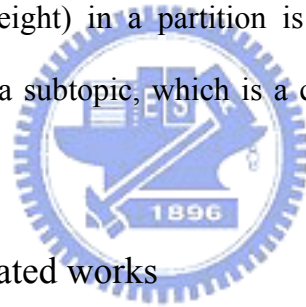
The value of σ generally depends on different applications, and should be tuned after running a number of preliminary experiments. With a high value of σ , the activation of a node propagated to its neighbors is in large amount. In this case, iSpreadRank outputs a ranking relying significantly on global information of the whole network. With a low value of σ , the propagation of activations among nodes becomes moderate, leading to an output ranking close to the initial ranking provided by the sentence scoring function based on sentence-specific features.

(2) Edge-weight normalization

In Eq. (3.5), R is a stochastic matrix, each row of which sums to one. In iSpreadRank, this matrix is employed for the purpose of edge-weight normalization. The design of edge-weight normalization is to divide the activation of a node propagated to its neighbors into the amount proportional to the relative weight between the node and each connected node [143]. From the perspective of a probabilistic interpretation, the relative edge weights define the propagation probabilities between nodes: at each node i , the activation is propagated to another node j with probability r_{ij} .

3.4.4 The proposed summarization approach

In essence, the proposed method can be regarded as a theme clustering based approach. As you have seen, iSpreadRank re-weights similar sentences with similar degree of importance, and ranks them in close positions in the inferred ranking. As a result, a sequence of similar sentences with close weights constitutes a partition of the ranking. Now, consider the sentence extraction module in Fig. 3.1; it sequentially examines sentences in the rank order, and adds one sentence at a time into the summary if it is not too similar to any sentences already in the summary. Successive sentences after a selected sentence are thus skipped until a dissimilar sentence is found. Based on these principles, the selection of the preceding sentence (i.e., the sentence with the highest weight) in a partition is similar to the extraction of a representative sentence from a subtopic, which is a common strategy used in theme clustering based approaches.



3.4.4.1 Comparison to related works

A majority of graph-based methods (e.g., [39], [100], [140]) assess the centralities of sentences using graph-based ranking algorithms originally developed to analyze webpage prestige, including PageRank [18] and HITS [64]. Conversely, the proposed iSpreadRank borrows concepts from spreading activation [106] that originated in psychology to explain the cognitive process of human comprehension. iSpreadRank further considers sentence-specific feature scores to help estimate the importance of sentences, while related works are only based on relationships between sentences (i.e., the network structure).

The use of sentence-specific features in this work resembles that of [140]. However, this work is quite distinct from theirs due to the underlying ranking

algorithm and the summary generation strategy. [39] also made use of heuristic features. Different from this study, heuristic features in their work are not integrated within the ranking algorithm; instead, the graph-based centrality is viewed as another feature, and is linearly combined with other features to yield a sentence scoring function.



Chapter 4

Query-focused Multidocument Summarization

Query-focused multidocument summarization is a particular task of multidocument summarization. Given a cluster of documents relevant to a specific topic, a query statement consisted of a set of related questions, and a user profile, the task is to create a brief, well-organized, fluent summary which either answers the need for information expressed in the query statement or explains the query, at the level of granularity specified in the user profile. The level of granularity, here, can be either specific or general: while a general summary prefers a high-level generalized description biased to the query, a specific summary should describe and name specific instances of events, people, places, etc. In this chapter, we deal with query-focused multidocument summarization using an extraction-based approach, which is enhanced with query-biased characteristics, to create an extractive query-focused summary of multiple documents that reflects particular points relevant to user's interests.

The proposed approach follows the most common technique for summarization, namely *sentence extraction*: important sentences¹⁷ are identified and extracted verbatim from documents and are composed into a summary. In the proposed approach, the query-focused multidocument summarization task is divided into four sub-tasks:

- (1) *Examining the degree of relevance between each sentence and the query statement;*
- (2) *Ranking sentences according to their degree of relevance to the query and their likelihood of being part of the summary;*

¹⁷ In this chapter, important sentences are referred to as sentences which either contribute to meeting the information need in the query or provide a general description of the document cluster.

(3) *Eliminating redundancy while extracting the most important sentences;*

(4) *Organizing extracted sentences into a summary.*

The first sub-task is addressed as a query-biased sentence retrieval task. For each sentence s , given a query q , the degree of relevance between s and q is measured as the degree of similarity between them, i.e., $sim(s, q)$. Three similarity measures are proposed to assess $sim(s, q)$. The first is computed as the dot production of the vectors of s and q in the vector space model. The second exploits latent semantic analysis (LSA) [32] to fold s and q into a reduced semantic space and computes their similarity based on the transformed vectors of s and q in the semantic space. Finally, with the idea of model averaging, the third combines the similarities obtained from the first and the second in a linear manner. In the second sub-task, several surface-level features are extracted to measure how representative a sentence is with respect to the whole document cluster. The feature scores, acting as the strength of representative power (i.e., the informativeness) of each sentence, are combined with the degree of relevance between the sentence and the query to score all sentences. As for the third sub-task, a novel sentence extraction method, inspired by maximal marginal relevance (MMR) [20] for redundancy filtering, is utilized to iteratively extract one sentence at a time into the summary, if it is not too similar to any sentences already included in the summary. In one iteration, all the remaining unselected sentences are re-scored and ranked using a modified MMR function, so as to extract the sentence with the highest score. Finally, in the fourth sub-task, all extracted sentences are simply ordered chronologically to form a coherent summary.

This chapter is structured as follows: Section 4.1 introduces the design of the proposed approach to query-focused multidocument summarization. Section 4.2 describes technical details of the proposed summarization approach. The experimental

results are reported in Section 4.3 and finally Section 4.4 provides discussions about the proposed summarization approach in different aspects.

4.1 Design

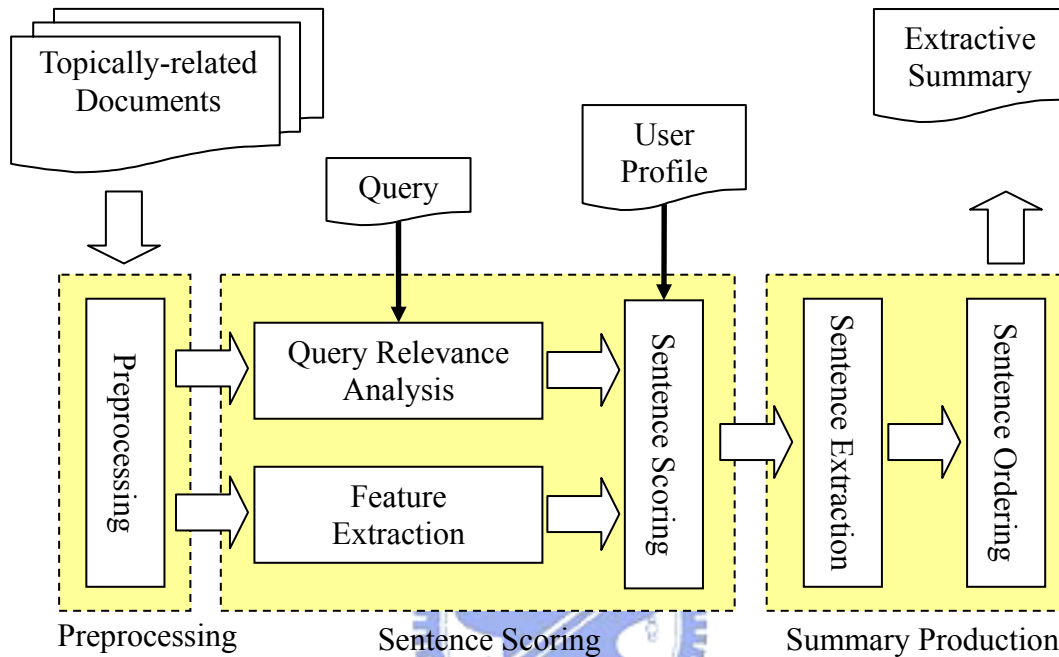


Fig. 4.1. The proposed query-focused multidocument summarization approach

Fig. 4.1 provides the design of the proposed approach to query-focused multidocument summarization. The inputs comprise a cluster of documents, a query statement, and a user profile. The output is a concise summary which either answers the need for information expressed in the query statement or explains the query, at the level of granularity specified in the user profile. The summarizer takes all the documents as a single document and produces an extractive summary by selecting sentences from the document group, which are relevant to the query and are representative with respect to the whole cluster. All sentences in the document group are first scored in consideration of the degree of relevance between a sentence and the query, as well as the strength of representative power of the sentence, which is assessed based on various surface-level features. The summarizer then iteratively

extracts one sentence at a time, which not only has a high score but also has less redundancy than other sentences extracted prior to it. The extraction finishes once the required length of the summary is met. The extracted sentences are finally composed into the output summary.

The whole summarization process can be decomposed into three phases: (1) the *preprocessing* phase preprocesses the input document and the query statement, (2) the *sentence scoring* phase scores each sentence by taking into account its relevance to the query and its feature scores, and (3) the *summary production* phase creates the output summary. The entire process, as shown in Fig. 4.1, can be further divided into several stages, namely *preprocessing*, *query relevance analysis*, *feature extraction*, *sentence scoring*, *sentence extraction*, and *sentence ordering*. They are outlined as follows, in order of execution:

(1) Preprocessing



Several linguistic analysis steps on the documents and the query are conducted in this stage, including tokenization, sentence boundary detection, Part-of-Speech (POS) tagging, stopword removal, word stemming and named entity (NE) recognition. Any words that are tagged with NN (Noun), VB (Verb), JJ (Adjective), or RB (Adverb) are viewed as valid unigrams and are used to generate bigrams and trigrams. For all unigrams, as well as bigrams and trigrams that appear in more than l sentences, they are included in a list of index terms. l , here, is heuristically set to 3 in the current implementation. With the index terms, a passage indexer constructs a vector representation of every sentence in the documents and the query using the TF-IDF term weighting scheme, proposed in [2]. For the term weighting scheme, please refer to Section 4.2.1.

(2) Query relevance analysis (see Section 4.2.1)

For each sentence s , given a query q , the degree of relevance between s and q is measured as the degree of similarity between them, i.e., $sim(s, q)$ ¹⁸. Three similarity measures are proposed to assess $sim(s, q)$. The first is computed as the dot production of the vectors of s and q in the vector space model. The second exploits latent semantic analysis (LSA) [32] to fold s and q into a reduced semantic space and computes their similarity based on the transformed vectors of s and q in the semantic space. Finally, the third combines the similarities obtained from the first and the second in a linear manner.

(3) Feature extraction (see Section 4.2.2)

A feature profile is created to capture the values of various sentence-specific features of all sentences. Six surface-level features are employed: (1) *position*, (2) *avg. TF-IDF weight*, (3) *similarity with title*, (4) *similarity with document centroid*, (5) *similarity with topic centroid*, and (6) *num. of named entities*. The feature scores, acting as the strength of representative power (or the informativeness) of each sentence, are combined together with the degree of relevance between the sentence and the query to score all the sentences in the stage of sentence scoring.

(4) Sentence scoring (see Section 4.2.3)

This module scores all sentences as the judgment to determine whether a sentence should be extracted into the summary. The scoring function that we employ is a weighted function, which takes into account together the degree of

¹⁸ The query statement q is first divided into several query sentences. While computing $sim(s, q)$, we only consider the maximum of all similarities of s over all query sentences.

relevance of a sentence to the query and its feature scores. If a general summary is expected, features, except the number of named entities, are considered in the function. All features, on the other hand, are taken into account in the function for producing a specific summary.

(5) Sentence extraction (see Section 4.2.4)

A sentence extraction module, based on maximal marginal relevance [20] for redundancy filtering, iteratively extracts one sentence at a time into the summary, if it is not too similar to any sentences already in the summary. In one iteration, all the remaining unselected sentences are re-scored and ranked using a modified MMR function, so as to extract the sentence with the highest score. In this way, only high-scoring sentences with less redundant information than others are extracted into the summary.

(6) Sentence ordering (see Section 4.2.5)

The extracted sentences are concatenated in chronological order to form the output summary. The following two criteria are applied to ensure the coherence of the summary: (1) if two sentences are extracted from different documents, they are ordered chronologically, and (2) if two sentences come from the same documents, their order remains the same as they are in the original document.

4.2 Algorithm

Section 4.2.1 defines the degree of relevance between a sentence and the query. Section 4.2.2 describes in detail different features that we employ to assess the strength of representative power of a sentence. Section 4.2.3 presents a weighted

scoring function to decide how possible a sentence belongs to the summary. While Section 4.2.4 introduces a novel redundancy filtering method for sentence extraction, Section 4.2.5 provides the method of sentence ordering.

4.2.1 Relevance between a sentence and the query

In this study, both the query and the sentence are represented as vectors of weighted terms, based on which the degree of similarity between the query and a sentence is determined. Let $W = \{t_1, \dots, t_i, t_{i+1}, \dots, t_j, t_{j+1}, \dots, t_m\}$ ($|W| = m$) denote the set of index terms, which is composed of unigrams $\{t_1, \dots, t_i\}$, bigrams $\{t_{i+1}, \dots, t_j\}$, and trigrams $\{t_{j+1}, \dots, t_m\}$. The vector representation of a sentence s is specified by Eq. (4.1), where $w_{i,s}$ is the TF-IDF weight of term t_i in s , given in Eq. (4.2).

$$s = \langle w_{1,s}, w_{2,s}, \dots, w_{m,s} \rangle \quad (4.1)$$

$$w_{i,s} = \log(tf_{i,s} + 1) \times \log\left(\frac{N + 1}{0.5 + n_i}\right) \quad (4.2)$$

In Eq. (4.2), $tf_{i,s}$ is the number of occurrences of t_i in s , N indicates the number of sentences in the document group, and n_i denotes the number of sentences where t_i appears.

Similarly, the vector representation of the query q is defined by Eq. (4.3), in which $w_{i,q}$ is a variant of the raw frequency of term t_i in q , given in Eq. (4.4).

$$q = \langle w_{1,q}, w_{2,q}, \dots, w_{m,q} \rangle \quad (4.3)$$

$$w_{i,q} = \log(tf_{i,q} + 1) \quad (4.4)$$

In Eq. (4.4), $tf_{i,q}$ is the number of occurrences of t_i in q .

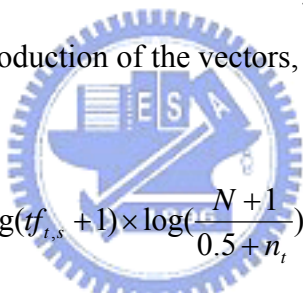
For each sentence s , given a query q , the degree of relevance between s and q is

measured as the degree of similarity between them, i.e., $sim(s, q)$. Three similarity measures are proposed to assess $sim(s, q)$. The first is computed as the dot production of the vectors of s and q in the vector space model (VSM). The second exploits latent semantic analysis (LSA) [32] to fold s and q into a reduced semantic space and computes their similarity based on the transformed vectors of s and q in the semantic space. Finally, the third combines, based on model averaging, the similarities obtained from the first and the second in a linear manner.

(1) Similarity based on VSM: $sim_1(s, q)$

Since s and q are both represented as vectors of weighted terms, as shown in Eq. (4.1) and Eq. (4.3), the VSM-based similarity between s and q is computed, by Eq. (4.5), as the dot production of the vectors, \vec{s} and \vec{q} .

$$sim_1(s, q) = \vec{s} \cdot \vec{q} \tag{4.5}$$

$$= \sum_{t \in q} \log(tf_{t,q} + 1) \times \log(tf_{t,s} + 1) \times \log\left(\frac{N + 1}{0.5 + n_t}\right)$$


Note that Eq. (4.5) has been shown effective for query-biased sentence retrieval in [2] and is used here as a competitive baseline.

(2) Similarity based on LSA: $sim_2(s, q)$

One problem of the VSM-based similarity measure is the principle of lexical matching. The number of words in sentences is relatively small and hence the number of matched keywords between s and q may not be significant, leading to a relevant sentence being judged irrelevant to the query, if they do not share the same terms. To address this problem, we propose the use of latent semantic analysis (LSA) [32] to relate s and q semantically.

Latent semantic analysis has recently been profitably employed in information retrieval to overcome the problem of lexical matching (e.g., [19], [32], [35]). It is a technique to extract the inherent latent structure in word usage, mainly based on the co-occurrences of words appearing in the data [32]. In this study, we apply latent semantic analysis at the sentence level: fold s and q into a reduced semantic space (i.e., latent structure) and compute the similarity based on the transformed vectors of s and q in the semantic space¹⁹.

The process of latent semantic analysis consists of four steps: (1) *singular value decomposition*, (2) *dimension reduction*, (3) *folding-in*, and (4) *computing $sim_2(s, q)$* . The following elucidates these steps in detail.

Let $A_{m \times n}$ be a word-by-sentence matrix²⁰, as shown in Eq. (4.6), with row i denoting an index term t_i , column j representing a sentence s_j , and entry $a_{i,j}$, specified by Eq. (4.2), signifying the weight of t_i in s_j .

$$A_{m \times n} = \begin{array}{c|cccc} & s_1 & s_2 & \cdots & s_n \\ \hline t_1 & a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ t_2 & a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ t_m & a_{m,1} & a_{m,2} & \cdots & a_{m,n} \end{array} \quad (4.6)$$

Singular value decomposition (SVD) is first performed on A and decomposes it into $A_{m \times n} = T_{m \times n} S_{n \times n} D^T_{n \times n}$, where T is an $m \times n$ matrix of left singular vectors, S is an $n \times n$ matrix with a diagonal $(\sigma_1, \dots, \sigma_n)$ and zeros elsewhere, and D is an $n \times n$ matrix of right singular vectors. While $rank(A) = r$, S holds the following property: $\sigma_1 \geq \dots \geq \sigma_r > \sigma_{r+1} = \dots = \sigma_n = 0$.

¹⁹ In the reduced space, s and q still have a high similarity even if they do not share any terms, as long as their terms are semantically related (or similar) [32].

²⁰ Without loss of generality, m is assumed to be greater than or equivalent to n .

Theoretically, matrices T and D have orthonormal columns (i.e., the column vectors have unit length and are all orthogonal to each other), and hence $U^T U = V^T V = I$, where I is the identity matrix. In interpretations, S is viewed as a semantic space (or the latent structure) inherent in the data, and T and D are regarded as the corresponding semantic representations of terms and sentences in S respectively.

In the second step, dimension reduction is performed on S , in order to derive a reduced semantic space. Only the first k ($k \leq r$) columns of T and D are kept and S is pruned into an $k \times k$ matrix, resulting in three matrices, T_k , D_k , and S_k . The choice of k depends on different applications: while a large k reflects all the real structure in the data, a small k removes noises from the data. Note that the purpose of dimension reduction is to relate words (or sentences) to k latent semantics. In the current implementation, k is fixed to 10 heuristically.

The following step folds A and q into the reduced space, S_k . Using Eq. (4.7), a new matrix, $\tilde{A}_{n \times k}$, is obtained, with row i standing for the semantic representation of sentence s_i in the reduced space, S_k .

$$\tilde{A}_{n \times k} = A^T T_k S_k^{-1} \quad (4.7)$$

Similarly, a query vector, $q = \langle w_{1,q}, w_{2,q}, \dots, w_{m,q} \rangle$, can be also projected into the same space, S_k , using Eq. (4.8).

$$\tilde{q}_{1 \times k} = q^T T_k S_k^{-1} \quad (4.8)$$

Finally, by multiplying matrices \tilde{A} and \tilde{q} , we have the proposed LSA-based similarity between s and q , $sim_2(s, q)$, as presented in Eq. (4.9).

$$\begin{aligned}
& [sim_2(s_1, q) \quad sim_2(s_2, q) \quad \cdots \quad sim_2(s_n, q)]_{1 \times n} \\
& = \tilde{q} \cdot \tilde{A}^T \\
& = (q^T T_k S_k^{-1}) (A^T T_k S_k^{-1})^T \\
& = q^T T_k (S_k^{-1})^2 T_k^T A
\end{aligned} \tag{4.9}$$

From the perspective of information retrieval, $q^T T_k (S_k^{-1})^2 T_k^T$, here, is viewed as a process of pseudo-expansion on q . Hence, $sim_2(s, q)$ still gives s a similarity of a non-zero value, even if s does not share any terms with q .

(3) Hybrid similarity: $sim_3(s, q)$

The hybrid similarity is practically defined as a linear combination of $sim_1(s, q)$ and $sim_2(s, q)$ to take advantages of both models. The proposed similarity measure is specified in Eq. (4.10).

$$sim_3(s, q) = \alpha \cdot sim_1(s, q) + (1 - \alpha) \cdot sim_2(s, q) \tag{4.10}$$

where α is set empirically to 0.5 in the current implementation.

4.2.2 Feature extraction

In the literature, a variety of surface-level features have been profitably employed to determine the likelihood of sentences of being part of the summary (e.g., [66], [76], [103], [137]). Inspired by the success of previous works, we also attempt to employ the feature scores of sentences to measure how representative a sentence is, with respect to the whole document cluster.

We take into account six surface-level features: (1) *position*, (2) *avg. TF-IDF weight*, (3) *similarity with title*, (4) *similarity with document centroid*, (5) *similarity with topic centroid*, and (6) *num. of named entities*. A feature profile is generated to capture the scores of features of all sentences. Each feature score in the feature profile

is normalized into the range between 0 and 1. The feature scores, acting as the strength of representative power (i.e., the informativeness) of every sentence, are further combined together with the degree of relevance between the sentence and the query, in order to score all sentences for extraction.

For a sentence s , the feature values of s are defined as follows:

(1) f_1 : Position

Important sentences tend to appear in particular positions (e.g., the beginning or the end) in a document. This feature, given in Eq. (4.11), is computed as inversely proportional to the position of a sentence from the beginning [54].

$$Score_{f_1}(s) = 1 - \frac{NC(s)}{|D|} \quad (4.11)$$

In Eq. (4.11), $|D|$ is the number of words in document D that contains s , and $NC(s)$ is the number of words appearing before s . Note that this formula gives the first sentence the highest score and the last one the lowest score.

(2) f_2 : Avg. TF-IDF weight

In general, a term with a high TF-IDF weight is usually important, implying that a sentence with a high average of all TF-IDF weights of its constituent terms tends to be an important sentence. This feature is defined by Eq. (4.12).

$$Score_{f_2}(s) = \underset{t \in s \text{ \& } t \text{ is significant}}{\text{Avg.}} w(t, s) \quad (4.12)$$

where $w(t, s)$ is the same as $w_{t,s}$ in Eq. (4.2). A term t is significant if it satisfies the criterion specified in Eq. (4.13).

$$u + 0.5\sigma \leq w(t) \quad (4.13)$$

where $w(t) = \sum_i w(t, s_i)$, u is the mean and σ is the standard deviation of $w(t_j)$ for all t_j .

(3) f_3 : Similarity with title

The title of a document always sums up the main point mentioned in the document. Hence, the more overlap with the title that a sentence has, the more important it is likely to be. This feature is computed, by Eq. (4.14), as the cosine of the angle between the vectors of \vec{s} and \vec{s}_{Title} .

$$Score_{f_3}(s) = sim(s, s_{Title}) = \frac{\vec{s} \cdot \vec{s}_{Title}}{|\vec{s}| \times |\vec{s}_{Title}|} \quad (4.14)$$

where s_{Title} is the title of the document in which s lies.

(4) f_4 : Similarity with document centroid

If a sentence contains more concepts identical to those of other sentences in the same document, it tends to be more significant. This feature measures the centrality of a sentence in a document, which is specified, by Eq. (4.15), as the similarity between the sentence and the centroid of the document.

$$Score_{f_4}(s) = sim(s, D_{Cent}) = \frac{\vec{s} \cdot \vec{D}_{Cent}}{|\vec{s}| \times |\vec{D}_{Cent}|} \quad (4.15)$$

where \vec{D}_{Cent} is the average vector of all \vec{s}_i in the same document where s lies.

(5) f_5 : Similarity with topic centroid

Similar to f_4 , this feature estimates the similarity of a sentence with the

centroid of the document cluster. This feature is obtained by Eq. (4.16).

$$Score_{f_5}(s) = sim(s, T_{Cent}) = \frac{\vec{s} \cdot \vec{T}_{Cent}}{|\vec{s}| \times |\vec{T}_{Cent}|} \quad (4.16)$$

where \vec{T}_{Cent} is the average vector of all \vec{s}_i in the whole document cluster.

(6) f_6 : Num. of named entities

This feature is particularly related to producing a specific summary, which is supposed to describe and name specific instances of events, people, places, organizations, etc. We assume that the more named entities (NE) a sentence has, the more specific it is. In the implementation, three types of NEs are chosen, including <Person>, <Organization>, and <Location>, and this feature is measured, by Eq. (4.17), as the number of NEs in s .

$$Score_{f_6}(s) = Num(<PER>, s) + Num(<ORG>, s) + Num(<LOC>, s) \quad (4.17)$$

where $Num(<PER>, s)$ is the number of <Person> in s , $Num(<ORG>, s)$ is the number of <Organization> in s , and $Num(<LOC>, s)$ is the number of <Location> in s .

4.2.3 Sentence scoring

The score of a sentence indicates the importance of the sentence and is employed as the judgment to determine whether the sentence should be extracted into the summary. The scoring function that we propose in this study is a weighted function, which takes into account: (1) the degree of relevance of a sentence to the query, and (2) its feature scores. While a general summary is expected, features, except feature f_6 (i.e., the

number of named entities), are considered in the function. All features, instead, are taken into account in the function for producing a specific summary.

To produce a *general* summary, for a sentence s , it is scored using a scoring function, as shown in Eq. (4.18).

$$Score(s) = \lambda \times sim(s, q) + \sum_{i,j=1..5} w_i \times Score_{f_i}(s) \quad (4.18)$$

where $sim(s, q) \in \{sim_1(s, q), sim_2(s, q), sim_3(s, q)\}$ and $Score_{f_i}(s)$ is the score of feature f_i of s .

To produce a *specific* summary, on the other hand, s is scored using a scoring function, as shown in Eq. (4.19).

$$Score(s) = \lambda \times sim(s, q) + \sum_{i,j=1..5} w_i \times Score_{f_i}(s) + w_6 \times Score_{f_6}(s) \quad (4.19)$$

where $sim(s, q) \in \{sim_1(s, q), sim_2(s, q), sim_3(s, q)\}$ and $Score_{f_i}(s)$ is the score of feature f_i of s .

It can be seen that the difference between Eq. (4.18) and Eq. (4.19) is the further consideration of feature f_6 in Eq. (4.19). The intuition behind this design is that a specific summary is supposed to describe and name specific instances of events, people, places, etc., and hence the more named entities a sentence has, the more specific it is. In the current implementation, parameters in Eq. (4.18) and Eq. (4.19) are set heuristically: $\lambda = 0.7$; $w_1 = 0.4$; $w_2 = 0.15$; $w_3 = 0.15$; $w_4 = 0.25$; $w_5 = 0.5$; $w_6 = 0.15$.

4.2.4 Sentence extraction

Once sentences are scored, one simply way towards sentence extraction is to rank sentences according to their scores and to extract the topmost sentences into the summary. In such a way, however, the summary may have redundant information since the degree of redundancy in information contained in a cluster of topically-related documents is much high, which is due to the reason that each document in the cluster is apt to describe the main points as well as necessary shared background [44].

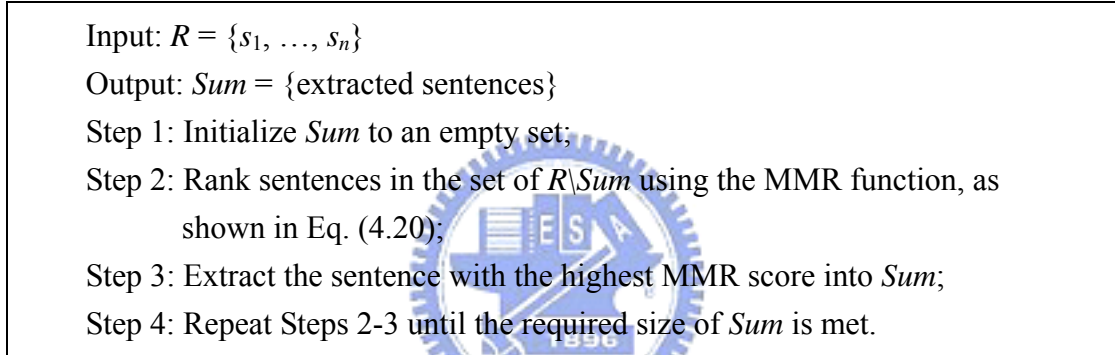


Fig. 4.2. The process of sentence extraction using MMR

One common approach to minimizing redundancy in summarization is *maximal marginal relevance* (MMR), proposed in [20]. Fig. 4.2 outlines the process of sentence extraction based on MMR and the MMR function is defined in Eq. (4.20).

$$\text{MMR} = \underset{s \in R \setminus Sum}{\overset{\text{def}}{\text{Arg max}}} [\gamma \times \text{SIM}_1(s, q) - (1 - \gamma) \times \max_{s_i \in Sum} \text{SIM}_2(s, s_i)] \quad (4.20)$$

where R is the set of sentences, Sum is the subset of sentences in R already extracted, $R \setminus Sum$ is the set difference (i.e., the set of as yet unselected sentences in R), SIM_1 is the similarity metric used in relevance ranking between sentences and the query, and SIM_2 can be the same as SIM_1 or a different similarity metric.

To sum up, while extracting a sentence, MMR follows the criteria: (1) the

maximum relevance of the sentence to the query, and (2) the minimum similarity of the sentence to previously extracted sentences in the summary. Therefore, only high-scoring sentences with less redundant information than others are extracted into the summary.

However, one shortcoming of the original MMR is that it does not take into account the feature scores of a sentence, which have been profitably used in summarization. To address this problem, we propose a modified version of MMR, as presented in Eq. (4.21).

$$\begin{aligned}
 \text{ModifiedMMR} &= \underset{s \in R \setminus \text{Sum}}{\text{Arg max}}^{\text{def}} [Score(s) - \delta \times \max_{s_i \in \text{Sum}} sim(s, s_i)] & (4.21) \\
 &= \underset{s \in R \setminus \text{Sum}}{\text{Arg max}} [\lambda \times sim(s, q) + \sum_{j=1..5 \text{ or } j=1..6} w_j \times Score_{f_j}(s) \\
 &\quad - \delta \times \max_{s_i \in \text{Sum}} sim(s, s_i)]
 \end{aligned}$$

where $sim(s, q) \in \{sim_1(s, q), sim_2(s, q), sim_3(s, q)\}$, $Score(s)$ is as that defined in Eq. (4.18) and Eq. (4.19), the similarity metric for $sim(s, s_i)$ is the same as that for $sim(s, q)$. In the current implementation, δ is empirically set to 0.3.

4.2.5 Sentence ordering

The extracted sentences are concatenated in chronological order to form the output summary. The following two criteria are applied to ensure the coherence of the summary: (1) if two sentences are extracted from different documents, they are ordered chronologically, and (2) if two sentences come from the same documents, their order remains the same as they are in the original document.

4.3 Evaluation

This section describes the data set, evaluation method, and the experimental results.

4.3.1 Date set and experimental setup

The DUC 2005 data set from DUC (Document Understanding Conferences) [34] was tested to examine the effectiveness of the proposed summarization method. The data set, created by NIST assessors, consists of 50 English news clusters. Each cluster has 25-50 documents, coming with a query statement and a user profile. The documents were collected from either Financial Times of London or Los Angeles Times. A query statement has a query title and a query narrative, which is consisted of a set of query questions to explicitly reflect the specific interests of a potential user in a task context. A user profile, stating either general or specific, was specified by the assessors to define the desired granularity of the summary. The task at DUC 2005 is to create, for each cluster, a brief, well-organized, fluent summary of roughly 250 words in length, which either answers the need for information expressed in the query statement or explains the query, at the level of granularity specified in the user profile.

Following the guideline of DUC 2005, several NIST assessors were each asked to read all the documents and to write a summary for each cluster. 30 of the clusters each has 4 human summaries and the remaining 20 of the clusters each has 9 or 10 human summaries. The manually-created summaries are treated as gold-standard summaries to evaluate the qualities of machine-generated summaries.

4.3.2 Evaluation method and metric

The machine-generated summaries were evaluated by means of ROUGE

(Recall-Oriented Understudy for Gisting Evaluation, alias RED) automatic n -gram matching [79]. ROUGE is a recall-oriented scoring metric for fix-length summaries, which adopts ideas from BLEU (BiLingual Evaluation Understudy) [104] to determine the quality of a summary. It generally counts as a performance indicator the number of co-occurrences between machine-generated and ideal summaries in different word units, such as n -gram, word sequences and word pairs.

Table 4.1. ROUGE runtime arguments for DUC 2005

<p>ROUGE -n 2 -x -m -2 4 -u -c 95 -r 1000 -f A -p 0.5 -t 0 -d</p> <p>-n 2: compute ROUGE-1 and ROUGE-2 -x: do not calculate ROUGE-L -m: apply Porter stemmer on both models and peers -2 4: compute Skip Bigram (ROUGE-S) with a maximum skip distance of 4 -u: include unigram in Skip Bigram (ROUGE-S) -c 95: use 95% confidence interval -r 1000: bootstrap resample 1000 times to estimate the 95% confidence interval -f A: scores are averaged over multiple models -p 0.5: compute F-measure with $\alpha = 0.5$ -t 0: use model unit as the counting unit -d: print per-evaluation scores</p>
--

Following the guideline to apply ROUGE for evaluation, all machine-generated summaries need to be truncated before evaluation, if the summary length is beyond the target length. Hence, we produced summaries with exactly 250 words, in order to have a fair evaluation. Model summaries (i.e., manual summaries) were truncated to 250 words before evaluation. Jackknifing was implemented so that human and system scores can be compared. ROUGE v. 1.5.5 was used and the runtime arguments of ROUGE for evaluation are listed in Table 4.1.

In this section, we only report the ROUGE-2 and ROUGE-SU4 scores since they are the basic official ROUGE scores at DUC 2005. Note that it has been found ROUGE-2 has high correlation (Spearman: 0.951; Pearson: 0.972) and ROUGE-SU4

has high correlation (Spearman: 0.942; Pearson: 0.958), when compared with human evaluation of the summaries for responsiveness [28].

4.3.3 Results

Table 4.2 summarizes the settings of different models that we evaluated in the experiments. Table 4.3 and Table 4.4 list the ROUGE-2 and ROUGE-SU4 scores of different experiments, respectively. In the tables, there is also one baseline, *NIST Baseline*, presented for comparison. The baseline, which is the official baseline at DUC 2005, simply outputs the first 250 words of the most recent document.

Table 4.2. Settings of different models

Models	Query Relevance	Use Feature?	Extraction
M1	sim_1	No	MMR
M2	sim_1	Yes	Modified-MMR
M3	sim_2	No	MMR
M4	sim_2	Yes	Modified-MMR
M5	sim_3	No	MMR
M6	sim_3	Yes	Modified-MMR

Table 4.3. ROUGE-2 scores obtained in different experimental settings

Models	ROUGE-2
M1	0.06503
M2	0.07071
M3	0.06609
M4	0.07265
M5	0.06640
M6	0.07256
NIST Baseline	0.04026

Table 4.4. ROUGE-SU4 scores obtained in different experimental settings

Models	ROUGE-SU4
M1	0.11700
M2	0.12310
M3	0.11935
M4	0.12568
M5	0.11866
M6	0.12594
NIST Baseline	0.08716

Several interesting results are found. First, M3 outperforms M1, implying that a better result can be obtained when latent semantic analysis (LSA) is employed. As mentioned in Section 4.2.1, LSA can extract the inherent latent structure in word usage and hence relates sentences and the query semantically, which leads to a higher recall when compared to the vector space model. As for M5, which utilizes the hybrid similarity measure, this model, however, does not perform as expected. The result is only as good as that of M3. Second, for models, M2, M4, and M6, it is observed that a scoring function, which takes into account both the degree of relevance of sentences to the query and the feature score of the sentences, can improve the performance of query-focused summarization. This suggests that it is reasonable for query-focused summarization to prior extract sentences with high relevance to the query and high feature scores. Finally, M2, M4, and M6 are superior to M1, M3, and M5, respectively. Recall that the modified MMR function (see Section 4.2.4) is designed to extract the sentence, which has high relevance to the query and high feature scores but has low redundancy with sentences already extracted in the summary. The results give the idea that the modified MMR is a suitable method for query-focused multidocument summarization.

Table 4.5 and Table 4.6, respectively, show the official ROUGE-2 and ROUGE-SU4 scores of human assessors and the top 5 systems at DUC 2005. In these tables, *SYSID* signifies the peer codes of participants: letters stand for human assessors, and numbers represent machine systems. Overall, the proposed summarization method is found to perform well with competitive results. Our best model (i.e., M4) has a ROUGE-2 score of 0.07265 and a ROUGE-SU4 score of 0.12568. The results are competitive to that of the best systems (see System 15 and System 17) at DUC 2005. Fig. 4.3 and Fig. 4.4 show the different ROUGE scores of

system and human peers at DUC 2005.

Table 4.5. Part of the official ROUGE-2 scores at DUC 2005

SYSID	ROUGE-2
C	0.11796
A	0.11777
E	0.10548
D	0.10062
B	0.10039
F	0.10025
J	0.09983
I	0.09833
G	0.09718
H	0.08859

15	0.07251
17	0.07174
10	0.06984
8	0.06963
4	0.06858

1 (NIST Baseline) (Rank: 31/32)	0.04026

Best machine (SYSID = 15)	0.07251
Median machine (SYSID = 24)	0.05967
Worst machine (SYSID = 23)	0.02564

Avg. of human assessors	0.10264

Table 4.6. Part of the official ROUGE-SU4 scores at DUC 2005

SYSID	ROUGE-SU4
C	0.17816
A	0.17618
D	0.16187
B	0.16113
J	0.16058
I	0.16030
G	0.15991
E	0.15937
F	0.15872
H	0.14843

15	0.13163
17	0.12973
8	0.12795
4	0.12773
10	0.12525

1 (NIST Baseline) (Rank: 31/32)	0.08716

Best machine (SYSID = 15)	0.13163
Median machine (SYSID = 3)	0.11666
Worst machine (SYSID = 23)	0.05569

Avg. of human assessors	0.16247

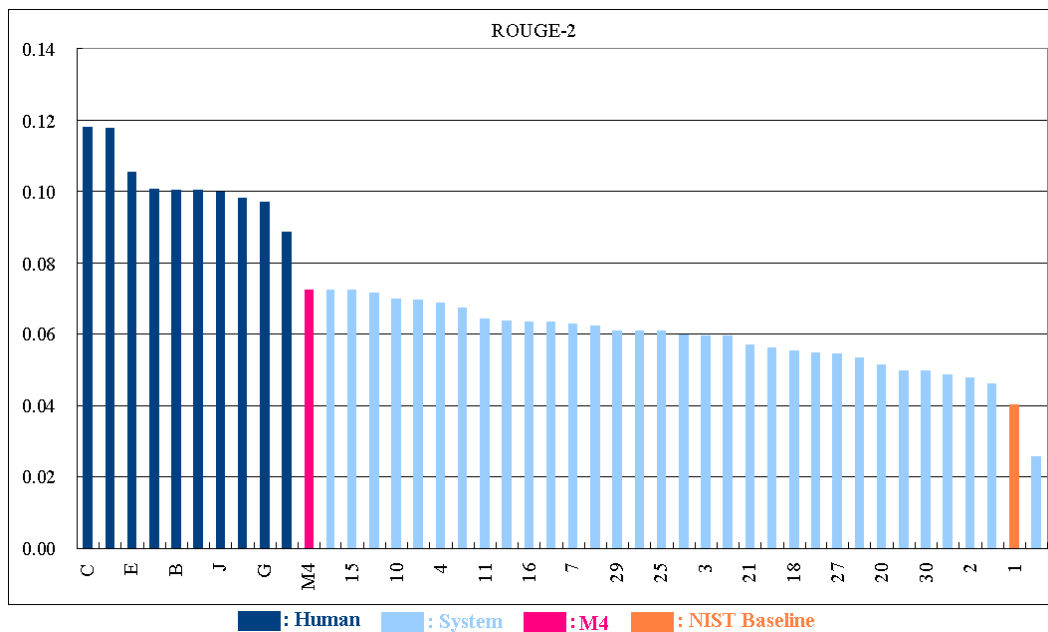


Fig. 4.3. ROUGE-2 scores of system and human peers at DUC 2005

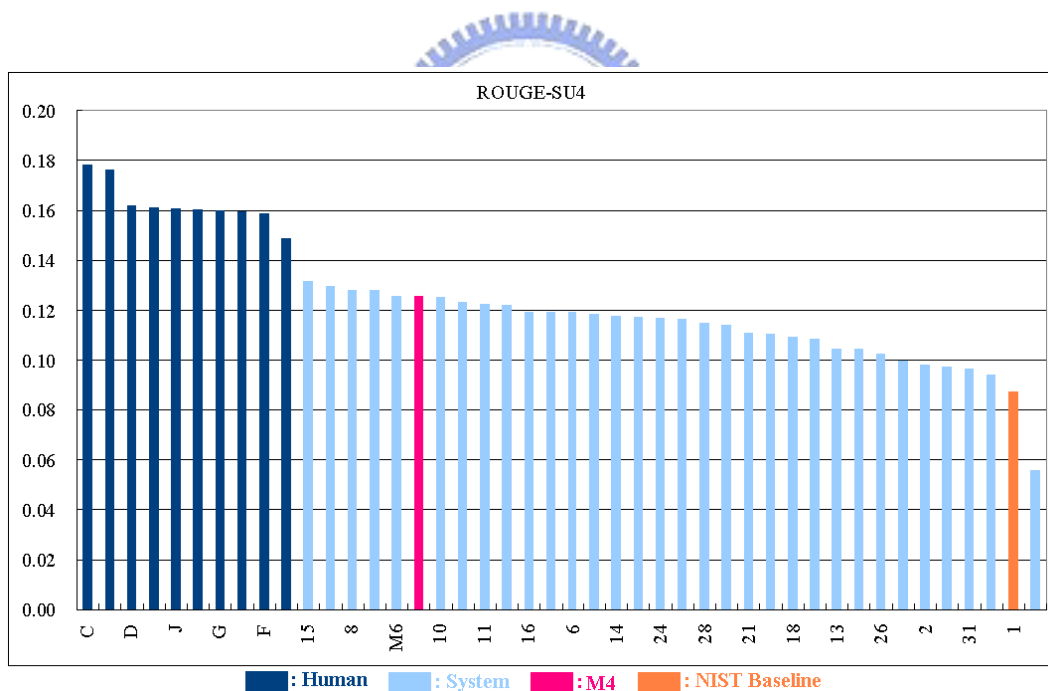


Fig. 4.4. ROUGE-SU4 scores of system and human peers at DUC 2005

4.3.4 Example output

Fig. 4.5 provides the ROUGE-2 scores of our best model, M4, for 50 clusters. The best ROUGE-2 score is 0.14298 for set d357i, the median ROUGE-2 score is 0.06963

for set d389h, and the worst ROUGE-2 score is 0.01192 for set d436j.

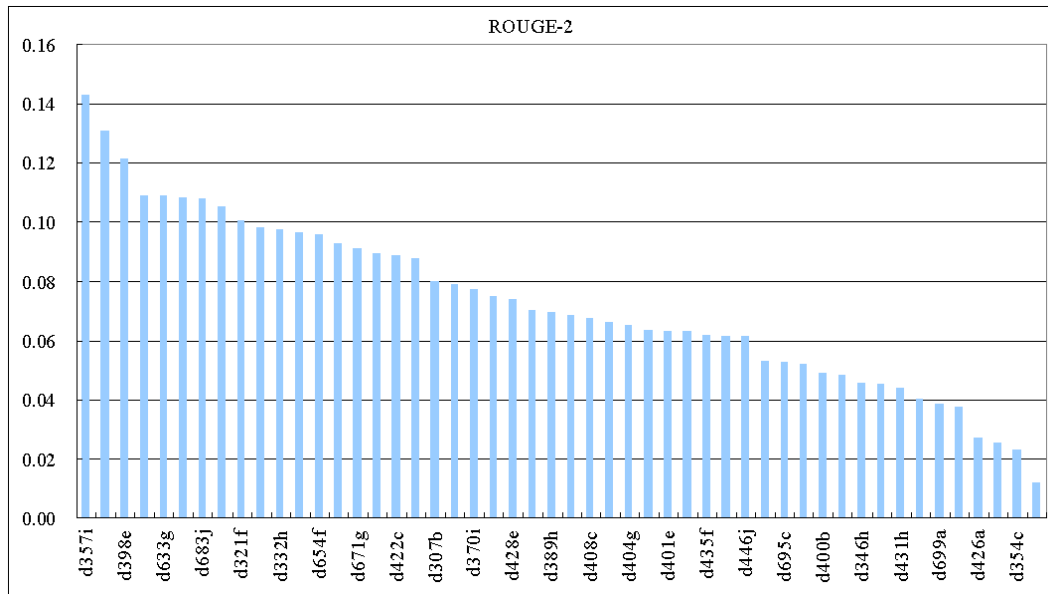


Fig. 4.5. ROUGE-2 scores of M4 for 50 clusters

Fig. 4.6 provides the ROUGE-SU4 scores of our best model, M4, for 50 clusters. The best ROUGE-SU4 score is 0.19789 for set d357i, the median ROUGE-SU4 score is 0.12312 for set d404g, and the worst ROUGE-SU4 score is 0.06281 for set d436j.

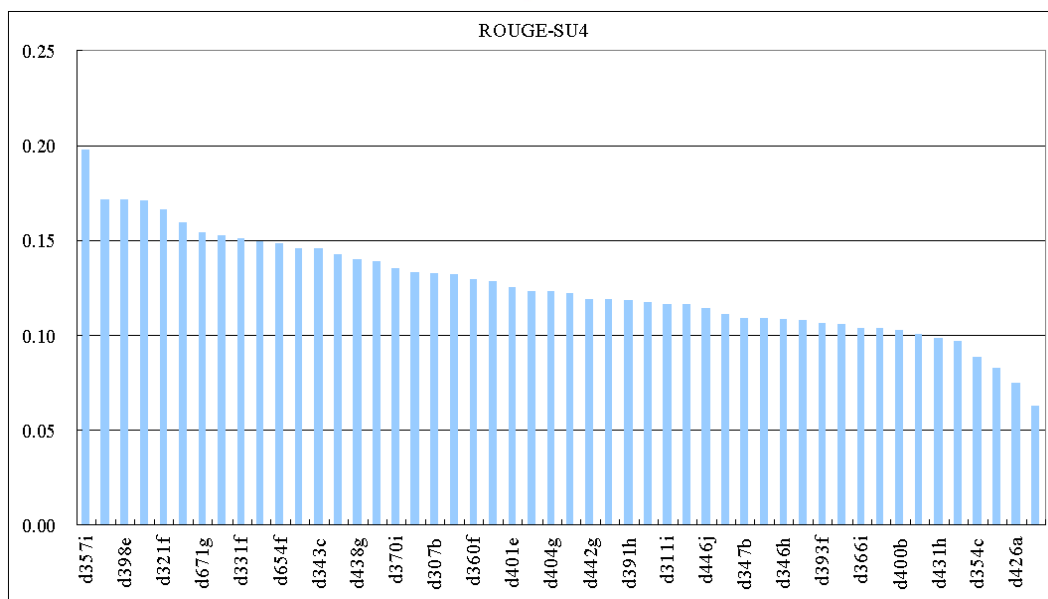


Fig. 4.6. ROUGE-SU4 scores of M4 for 50 clusters

Fig. 4.7 gives the query statement for set d357i, which asks for a *specific*

summary, Fig. 4.8 shows the system output, and Fig. 4.9 to Fig. 4.12 provide the human summaries. In this example, we obtained a *good* result (ROUGE-2 score of 0.14298 and ROUGE-SU4 score of 0.19789): when compared to model summaries (E, F, I), the ROUGE-2 score is 0.15180 and the ROUGE-SU4 is 0.20375; when compared to model summaries (D, F, I), the ROUGE-2 score is 0.12844 and the ROUGE-SU4 is 0.18646; when compared to model summaries (D, E, I), the ROUGE-2 score is 0.12299 and the ROUGE-SU4 is 0.18102; when compared to model summaries (D, E, F), the ROUGE-2 score is 0.16869 and the ROUGE-SU4 is 0.22034.

Query Statement

```
<topic>
  <num> d357i </num>
  <title> Boundary disputes involving oil </title>
  <narr>
    What countries are or have been involved in land or water boundary disputes
    with each other over oil resources or exploration? How have disputes been
    resolved, or towards what kind of resolution are the countries moving? What
    other factors affect the disputes?
  </narr>
  <granularity> specific </granularity>
</topic>
```

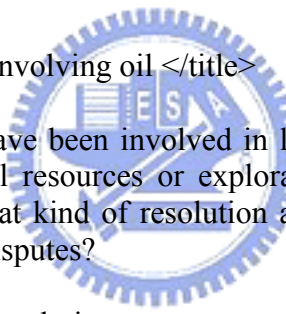


Fig. 4.7. Query statement for d357i

System Summary (~250 words)

China yesterday denounced Vietnam's contract with US oil company Mobil as a violation of Beijing's sovereignty in the South China Sea, but has pledged to settle all disputes peacefully, Reuter reports from Beijing. Britain and Argentina are to hold talks in July on joint oil exploration in waters surrounding the Falkland Islands, two years after a previous round of oil talks collapsed. VIETNAM has accused China of landing troops on a reef of the disputed Spratly islands in the South China Sea and planting a territorial marker. CHINA'S recent reaffirmation of its claim to the disputed Spratly islands in the South China Sea overshadowed the start of a meeting of the Association of South-East Asian Nations (Asean) in Manila yesterday. THE foreign ministers of Vietnam and China gave assurances yesterday that their dispute over the Spratly islands in the South China Sea would not escalate into armed conflict, Reuter reports. THE FALKLAND Islands government has chosen two companies for a multi-million pound seismic study of territorially disputed waters in the South Atlantic from October, confirming its decision to exclude Argentina from the initial search for oil. British Gas and YPF, Argentina's privatised oil company, have been discussing the formation of a joint venture to explore for hydrocarbons in disputed South Atlantic waters. China and Vietnam agreed yesterday to work peacefully to resolve their disputes over territorial and maritime boundaries, and are to set up a group of experts to discuss their rival claims to the Spratly atolls and the waters

Fig. 4.8. System summary for d357i

Model Summary: D

From 1990 to 1992 there were three major boundary disputes occurring. The first, a dispute over the Spratley Islands involving China and Viet Nam and to a lesser extent Malasia, Brunei, Taiwan, and the Philippines. These islets became an issue when China granted an oil exploration concession to a US company in an area claimed by Viet Nam. This move threatened the oil industry of Viet Nam. Ten years after the conclusion of war, Britain and Argentina are again in dispute over the waters around the Falkland Islands. Britain has begun seismic studies in hopes of drilling for oil in the waters surrounding the islands. Argentina wants results of the survey negotiations. Britain has agreed to cooperate with Argentina because they will need to use Argentinian land for the supporting the oil industry. There also been disputes between the two countries over fishing rights. A third dispute over territorial rights involving oil is between Greece and Turkey regarding the continental shelf under the Aegean Sea. Greece has opened the way for oil exploration, but has agreed to international arbitration regarding seabed mineral rights. They have rejected Turkey's proposals for joint exploration of the area. A territorial dispute has been settled between Denmark and the Faroe Islands. The Danish government has agreed to hand over all mineral rights to the Faroese government. Another potential dispute could be over oil rights in Crimea in the Black Sea currently controlled by Britain, but with interest from the Ukraine, American, Dutch, and Norwegian companies.

Fig. 4.9. Model summary, created by D, for d357i

Model Summary: E

China and Vietnam both claim the Spratly Islands in the South China Sea that are believed to have oil and gas deposits under the seabed. Taiwan, Malaysia, the Philippines, and Brunei also claim to some of Islands. China has signed an agreement with a US energy company to begin exploration for oil in that area. China has also entered into an area in the Gulf of Tonkin claimed by Vietnam. The six countries laying some claim to the disputed area have proposed a regional settlement. Later, China and Vietnam agreed to work peacefully to resolve territorial and maritime boundaries and consider joint development. Argentina was initially excluded from oil exploration around the Falkland Islands by Britain. Fishing rights were also at issue. Eventually, Britain and Argentina agreed to hold talks on joint oil exploration in that area. A United Nations Commission settled the border issue between Kuwait and Iraq over oilfields after the Gulf War. An international court of arbitration settled a dispute between France and Canada over territorial water around two French islands in the Atlantic. Canada was awarded oil rights. Greece and Turkey have both claimed oil exploration rights in the Aegean. Arguments between Russia and Ukraine over the Crimea have affected oil exploration there. The Danish government has agreed to give the Faroe Islands mineral rights. Evidence of oil may enter into the dispute between Britain and Iceland over fishing rights in the waters around Rockall.

Fig. 4.10. Model summary, created by E, for d357i

Model Summary: F

China and Vietnam agreed, in 1994, to work peacefully to resolve their disputes over the Spratly atolls and waters of the South China Sea. Both countries had awarded oil exploration contracts to U.S. oil companies in the area. Five other nearby countries also claimed the Spratlys, and spokesmen for some of them accused China of "bullying" them in an effort to fill the power vacuum left by the reduction of US troops and Russia's retreat from the region. China's foreign minister said the two sides could start "joint development" if talks failed. In 1994, British Gas and YPF, Argentina's privatized oil company, discussed a joint venture to explore for hydrocarbons in South Atlantic waters between Argentina and the Falkland Islands. But talks stalled because of disagreement between Britain and Argentina on sovereignty over the Falklands. Although Argentina was defeated in the 1982 war with Britain over the Falklands, it still claimed the islands. Tensions flared in 1994 between Greece and Turkey when the UN's Law of the Sea went into effect allowing nations to claim territorial waters and their resources, 12 miles out. Turkey declared it was ready to take up arms to prevent Greece from claiming Aegean waters 12 miles from its shoreline. Other disputes involving oil rights included that between Denmark and Great Britain over the demarcation line between Denmark's Faroe Islands and Britain's Shetlands; Iceland and Denmark versus Britain, claiming the small island of Rockall; and the unsettled Ukrainian-Russian dispute over Crimean waters.

Fig. 4.11. Model summary, created by F, for d357i

Model Summary: I

Britain and Argentina both claim Falkland Island territorial waters, suspected of oil reserves. Argentina used a fisheries dispute to press for participation in development. Britain and Argentina discussed joint-venture exploration. Vietnam and China have battled over undersea oil and gas exploitation around the Spratly Islands, partially claimed also by Taiwan, Malaysia, the Philippines and Brunei. China claimed sovereignty over the area and drilled in Vietnamese waters. Vietnam drilled nearby. ASEAN successfully urged restraint and suggested a joint development zone similar to Thailand and Malaysia's. China tentatively agreed. Greece and Turkey dispute offshore oil rights in the Aegean. Greece rejected Turkish proposals for joint Aegean exploration. An international court of arbitration lifted a drilling moratorium on waters around Saint-Pierre and Miquelon islands, disputed by France and Canada, allowing Canada to award oil contracts. Gas and oil exploration around the Faroe Islands was blocked by Denmark's dispute with the Faroese over mineral rights and its claim that Britain's demarcation line was too close. Denmark threatened to go to the International Court. Denmark ceded mineral rights to the Faroese. Denmark, Britain and Iceland all claim Rockall Island, suspected of containing oil. An unsettled Ukrainian-Russian territorial dispute threatened the first deep Black Sea offshore oil well. The 1994 UN LOS convention encourages resource exploitation compromise in unresolved maritime disputes, citing an Australian-Indonesian accord over Timor's waters. One party cannot exploit territory-straddling oil deposits without damaging another's interests. After Iraq's invasion of Kuwait and its oilfields, a UN Commission proposed giving Kuwait several Iraqi wells.

Fig. 4.12. Model summary, created by I, for d357i

Fig. 4.13 gives the query statement for set d694j, which asks for a *specific* summary, Fig. 4.14 shows the system output, and Fig. 4.15 to Fig. 4.18 provide the human summaries. In this example, we obtained a *bad* result (ROUGE-2 score of 0.02543 and ROUGE-SU4 score of 0.08266): when compared to model summaries (H, I, J), the ROUGE-2 score is 0.02219 and the ROUGE-SU4 is 0.07118; when compared to model summaries (G, I, J), the ROUGE-2 score is 0.02749 and the ROUGE-SU4 is 0.09157; when compared to model summaries (G, H, J), the ROUGE-2 score is 0.03003 and the ROUGE-SU4 is 0.08717; when compared to model summaries (G, H, I), the ROUGE-2 score is 0.02199 and the ROUGE-SU4 is 0.08073.

Query Statement

```
<topic>
  <num> d694j </num>
  <title> Fertile Fields </title>
  <narr>
    Discuss making and using compost for gardening.  Include different types of
    compost, their uses, origins and benefits.
  </narr>
  <granularity> specific </granularity>
</topic>
```

Fig. 4.13. Query statement for d694j

System Summary (~250 words)

The Christmas tree which stood outside Number 10 Downing Street was removed yesterday to be recycled as compost for a children's nature garden in south London. Our mission was a relatively simple one: retrieve discarded Christmas trees and bring them to nearby Balboa Park where the Los Angeles City Bureau of Sanitation was sponsoring a tree recycling program to turn used trees into compost and mulch. Sir, Peat producers hampered by wet weather this year would suffer less from the vagaries of the British climate if they switched to making peat-free composts in future ('Scottish peat profits bogged down by squelchy weather', June 29). Baltimore officials have agreed to take back a trainload of sewage sludge after authorities in Louisiana and Mississippi refused to let a disposal company unload the waste in their states. THE LATEST development in the seed and potting compost controversy is that Fisons, connected closely with the development of peat composts for 25 years, is about to market a non-peat product in direct competition with its own well-known Levington Multipurpose. For the second consecutive year, the city of Carson has teamed with two private firms to recycle Christmas trees. Becker said the Orange County program is modeled after one in Los Angeles County, where sanitation districts use a 20-acre compost site in Carson and work with a contractor, Kellogg's Nitrohumus Co., to distribute the material. Fisons started seed and potting compost research in 1956 when it moved to Levington, near Ipswich, Suffolk, but the research was

Fig. 4.14. System summary for d694j

Model Summary: G

Good gardening compost is commonly made from a variety of sources: vegetable and fruit waste, plant waste, tree bark, grass clippings, peat, chicken waste and horse manure. Peat has the longest history of use as garden compost but ecological concerns about exhausting peat sources have led to development of other material. Many prefer good topsoil from the garden or vermiculite and perlite. Special seed compost can be purchased in bags but it often becomes lumpy when watered. Research has led to development of special peat-free composts for either potting seed, seed germination, or for growing-on mature plants. Now there is a single mixture for all purposes in the home garden. There is also a peat-free product made from softwood bark which has no toxic or other elements harmful to the garden. Composted horse manure, which takes about 19 days to degrade, has only 3% salt and is usually mixed with five parts grass and leaves. A growing concern about the recycling ethic and water conservation has led to an increase in backyard composting. It allows the gardener to keep a steady supply of compost for use around the yard. It can be made from green waste, leaves, and fruit and vegetable waste, kept in a homemade wire or screen cage, and covered with newspaper or plastic to retain heat. It is turned and mixed every few months until it decays. The city-dump-is-going-to-overflow crisis is aided by the use of sewer sludge mixed with sawdust and the composting of Christmas trees.

Fig. 4.15. Model summary, created by G, for d694j

Model Summary: H

Composts hold water and keep soil aerated; some provide nutrients. Pure sphagnum moss peat composts from Ireland, England, U.S. and elsewhere are best for seeding and potting. They are clean, light, easy to handle, highly absorbent, moderately acid, porous, and sterile. They provide very little plant food, so plant-specific nutrients can be added. Original British "John Innes" style soil-based composts consist primarily of sphagnum moss peat and high quality loam. Low quality brands contain too little humus and become impervious to air and water. Multi-purpose non-soil peat composts were developed as high quality loam became scarce. Some contain chemicals and wetting agents. Multi-purpose non-peat composts were developed as high-quality peat became scarce, initially using non-toxic softwood bark from timber waste. Many commercial composts use recycled materials. Dehydrated and limed sewer sludge is mixed and oxidized with bark sawdust, as well as organic household waste and plant waste. Some commercial composts add chemical nitrogen or animal waste. Home gardening composts can be any garden (leaves, grass) or kitchen refuse (fruits, vegetables, husks, egg shells, coffee grounds, canning waste, not meat), on top of twigs or chopped corn stalks to aid aeration and drainage. Top with grass clippings or manure (20%). Keep moist and stir monthly to aerate. Cover loosely to retain heat. In apartments use a plastic bag, but aerate frequently. Composting kits add helpful organic compounds. Compost can also be used as top-cover mulch to hold in moisture. Fungus composted wood chips are used to decontaminate organic compounds from soil.

Fig. 4.16. Model summary, created by H, for d694j

Model Summary: I

In composting, bacteria and fungi turn organic refuse into a moist black soil conditioner. Composting reduces waste. Some compost removes toxic soil residue. Compost mulch protects from frost, controls weeds, and keeps soil moist. Traditional topsoil, leafmold, and sand composts were lumpy, impervious, and too-finely textured. Use declined with loam shortages and peat-based compost popularity. Peat is absorbent, porous, light, clean, and convenient, but peat bogs shelter rare species and need to be conserved. Substitutes include plant waste, ground-up Christmas trees, coconut fiber, softwood bark, and shredded newspaper. Mushroom-growing composts may contain horse manure, straw, millet, grape residue, chicken waste, and cottonseed meal. They are sold to gardeners after harvest. Pine needle or redwood sawdust (containing natural fungicides) is mixed with sand, perlite or soil for growing bulbs. In a home compost frame of 2-by-4s and chicken wire, pile vegetarian kitchen refuse (fruit and vegetable waste, coffee grounds, egg shells) over a base of twigs or chopped corn stalks. Add nitrogen-rich material (manure, grass clippings, hay, green weeds, fertilizer, aluminum sulfate), coarser material for aeration, and turn occasionally, wetting when dry. Cover to retain heat. Or, partially fill a black plastic bag with kitchen and plant refuse, add a nitrogen product and water. Set in sunlight, poke drainage holes, and kick occasionally. Communities have composted commercial fruit and vegetable processor waste or horse manure with grass, shredded leaves, and tree trimmings. Sewage sludge has been mixed in computer-controlled processes with bark, sawdust, leaves, and household waste to become agricultural compost.

Fig. 4.17. Model summary, created by I, for d694j

Model Summary: J

British gardeners began by using a combination of topsoil, leaf mold and sand for a growing medium. This was generally replaced by sphagnum peat moss extracted from bogs. It was absorbent, porous and virtually sterile. Because of the depletion of the bogs, it fell from favor and peat-free composts were devised using softwood bark, plant waste or coconut fiber. These were pleasant to handle and moist and crumbly. Backyard composting became very popular. A free fertilizer is created from non-meat kitchen waste and garden waste, moistened and aerated until it decays. Cities began composting yard waste to save landfill space. Christmas trees are composted from California to Number 10 Downing Street. Horse manure, which degrades in only 4 weeks, is a good addition to this compost, as is shredded newspaper. Cities are trying mixing treated sewer sludge with leaves, sawdust, or bark to create a relatively odorless compost. Mushrooms are grown in compost made of horse manure with millet added and covered with peat, or a combination of grape residue, chicken waste, cottonseed meal, and straw. Calochortus grow best in a combination of sand, perlite, and pine needles or of topsoil, sand, and sawdust. Cover dahlia tubers with soil and peat or compost. Vegetables need yard compost with chicken manure and fish emulsion added. Amend soil for flowers with redwood compost, KRA organic amendment, or Bandini soil builder. One special compost combines wood chips and fungi to breakdown toxic organic compounds. Another adds nematodes to kill harmful insect larva.

Fig. 4.18. Model summary, created by J, for d694j

Fig. 4.19 gives the query statement for set d376e, which asks for a *general* summary, Fig. 4.20 shows the system output, and Fig. 4.21 to Fig. 4.29 provide the human summaries. In this example, we obtained a *good* result (ROUGE-2 score of 0.13087 and ROUGE-SU4 score of 0.17087): when compared to model summaries (B, C, D, E, G, H, I, J), the ROUGE-2 score is 0.12783 and the ROUGE-SU4 is 0.16838; when compared to model summaries (A, C, D, E, G, H, I, J), the ROUGE-2 score is 0.13267 and the ROUGE-SU4 is 0.17314; when compared to model summaries (A, B, D, E, G, H, I, J), the ROUGE-2 score is 0.12882 and the ROUGE-SU4 is 0.17107; when compared to model summaries (A, B, C, E, G, H, I, J), the ROUGE-2 score is 0.12437 and the ROUGE-SU4 is 0.16543; when compared to model summaries (A, B, C, D, G, H, I, J), the ROUGE-2 score is 0.12786 and the ROUGE-SU4 is 0.16836; when compared to model summaries (A, B, C, D, E, H, I, J), the ROUGE-2 score is 0.13179 and the ROUGE-SU4 score is 0.16923; when compared to model summaries (A, B, C, D, E, G, I, J), the ROUGE-2 score is 0.13593 and the ROUGE-SU4 score is 0.17348; when compared to model summaries (A, B, C, D, E, G, H, J), the ROUGE-2 score is 0.13498 and the ROUGE-SU4 score is 0.17475; when compared to model summaries (A, B, C, D, E, G, H, I), the ROUGE-2 score is 0.13357 and the ROUGE-SU4 score is 0.17397.

<p>Query Statement</p> <pre> <topic> <num> d376e </num> <title> World Court </title> <narr> What is the World Court? What types of cases does the World Court hear? </narr> <granularity> general </granularity> </topic> </pre>

Fig. 4.19. Query statement for d376e

System Summary (~250 words)

The International Court of Justice in The Hague ruled in Chad's favour in a 20-year border dispute with Libya which has caused two wars. The World Court yesterday rejected Libya's plea that it should bar the US and Britain from taking punitive measures to oblige Tripoli to hand over two men suspected of the 1988 bombing of a Pan Am airliner over Lockerbie in Scotland. THE presidents of Honduras and El Salvador were set yesterday to begin two days of talks in San Salvador, amid continuing tension in border areas following a judgment on their common frontier this month by the International Court in the Hague. LIBYA WENT to the International Court of Justice yesterday seeking an emergency injunction to restrain the US and Britain from using force or imposing sanctions in their campaign to get Tripoli to surrender two Libyans alleged to have been involved in the Lockerbie bombing. THE US, Britain and France have agreed on a draft United Nations resolution imposing sanctions on Libya, including a ban on all air links with the country, for its alleged part in the bombing of civil airliners over Scotland and Niger. An international war crimes tribunal covering the former Yugoslavia formally opens in The Hague today with a request for the extradition from Germany of a Bosnian Serb alleged to have killed three Moslem prisoners. AN INTERNATIONAL tribunal to judge war crimes committed since 1991 in former Yugoslavia should be set up under the mandatory enforcement procedures of the UN

Fig. 4.20. System summary for d376e

Model Summary: A

Seated in The Hague, in the Netherlands, the International Court of Justice, also known as the World Court, is the main judicial arm of the United Nations. As such, it decides on legal disputes between states. One type of case the Court deals with concerns questions of international law. The Court found the US guilty of violating international law when it mined the harbors in Nicaragua. In the case of two Libyans accused of blowing up a Pan Am flight over Scotland, the US and Britain wanted Libya to turn the suspects over to them. Libya claimed it had the right to try them and asked the court to restrain the US and Britain from using force or imposing sanctions against Libya for failing to turn over suspects. A second type of case involves violations of international conventions, such as genocide and war crimes. The Court considered the claim by Bosnia that Serb carried out a campaign of ethnic cleansing and genocide against them. They also considered a similar claim by Serbia of genocide carried out by Moslems. A third type concerns interpretations of treaties and accords. Portugal, the recognized administrator of East Timor, challenged Australia's accord with Indonesia in the Court over waters south of Timor. Additional types of cases are territorial disputes, such as South Africa's control over Namibia; border disputes, such as between Honduras and El Salvador; and reparation cases, such as Iran's request for compensation from the US for downing an Iranian Airbus with a missile.

Fig. 4.21. Model summary, created by A, for d376e

Model Summary: B

The World Court or International Court of Justice (ICJ), located in The Hague, Netherlands, is made up of fifteen permanent judges plus two judges nominated by the parties involved in the dispute under consideration. The court has no power to enforce its orders but its decisions have traditionally carried some diplomatic weight. The ICJ usually rules on cases brought by one nation against another. One of the earliest and longest standing decisions of the ICJ involved the Corfu Channel incident in 1946 when two British destroyers struck Albanian-laid mines while exercising their right of passage through Corfu Channel. One was sunk with loss of 44 lives and the other scuttled. The ICJ ruled in 1948 that Albania should pay Britain 843,947 pounds. Albania paid up 1.1 million pounds in May 1992. The ICJ has settled Border disputes. A 20-year dispute between Libya and Chad was settled in Chad's favor in 1994. A 23-year dispute between Honduras and El Salvador was settled by a compromise devised by the ICJ in 1992. On the other hand, the ICJ ordered Serbia and Bosnia to cease committing genocide in 1993 without effect. In 1951 a company, British Petroleum, sued Iran before the ICJ over the nationalization of all foreign oil interests in Iran. The ICJ also served as the venue for the international war crimes tribunal established by the United Nations in 1993.

Fig. 4.22. Model summary, created by B, for d376e

Model Summary: C

The World Court, or International Court of Justice in The Hague, is made up of 15 permanent judges, plus a further two nominated by the parties involved in a case. It hears cases involving international disputes. Although, it does not have powers to enforce its orders, its decisions have traditionally carried some diplomatic weight. The types of cases it hears would include disputes over jurisdiction or extradition of criminal suspects, such as two Libyan suspects in the Pan Am Lockerbie bombing. Another type would be cases of long-standing border disputes over which wars may have been fought. The World Court hears cases about embargoes imposed by one country against another, and disagreements over territorial ocean waters. It hears cases involving questions of sovereignty, illegal mining of harbor waters, and illegal nationalization of foreign oil interests. Many cases which come before the World Court involve accidents of war, such as the post-World War II Corfu Channel incident in 1948 and the 1988 Iran Air Flight 655 incident in the Strait of Hormuz. The World Court considered accusations from both sides fighting in the former Yugoslavia since 1991. It ordered both Bosnia and Serbia to stop acts of genocide. The UN Security Council convened an International War Crimes Tribunal to be held at The Hague, where the International Court of Justice is located. The UN-sponsored tribunal to deal with war crimes committed by Serbia and Bosnia was sworn in for a four-year period.

Fig. 4.23. Model summary, created by C, for d376e

Model Summary: D

The International or World Court located in The Hague, Netherlands is made up of 15 permanent judges plus two others nominated by the parties involved in a dispute. Its purpose is to handle disputes between nations and sometimes acts in conjunction with war crimes tribunals. The court has no power to enforce its orders but its decisions often carry diplomatic weight. For example, when it found the United States guilty of illegally mining the harbors of Nicaragua, the US ignored its verdict, but in another case when it sided with the United States, the US worked to enforce the Court's decision. It is up to the discretion of the parties involved whether to submit to the judgment of the Court. The following are cases addressed by the World Court between 1989 and 1993. Libya brought a case against Britain and the US disputing their demand for extradition of Libyans charged with blowing up a Pan Am jet over Lockerbie, Scotland. The Court rejected Libya's plea. The presidents of Honduras and El Salvador entered a dispute over their borders. BP oil of Britain entered a case against Iran when Iran abruptly nationalized all foreign oil interests. The Court ruled in a border dispute between Chad and Libya in Chad's favor. An international tribunal at The Hague convened to judge war crimes committed in the former Yugoslavia. The court ordered Serbia and Bosnia to stop acts of genocide. The Court made decisions regarding South Africa's control of the country of Namibia

Fig. 4.24. Model summary, created by D, for d376e

Model Summary: E

The International Court of Justice, also known as the World Court, is the judicial arm of the United Nations. It is located in the Hague, in The Netherlands and has 15 permanent judges plus a further two nominated by parties going to the court. The World Court does not have any powers to enforce its orders, but its decisions carry diplomatic weight. The World Court has pronounced judgments on border disputes between Honduras and El Salvador and between Chad and Libya. The Court ruled that Albania should compensate Britain for a destroyer sunk in the Corfu Channel shortly after the end of World War II. When Iran nationalized foreign oil interests in 1951, British Petroleum went to the World Court. The World Court overruled a UN decision that declared South African control over Namibia illegal. Libya has brought a case before the World Court against Britain and the US concerning their demand for extradition of Libyans charged with blowing-up a Pan Am jet of Scotland in 1988. An international tribunal to judge war crimes in the former Yugoslavia has been proposed and would sit in the Hague where the World Court is located. The World Court had ordered Serbia and Bosnia to stop act of genocide in Bosnia. The World Court found the US guilty of violating international law by mining harbors in Nicaragua. The US ignored its verdict.

Fig. 4.25. Model summary, created by E, for d376e

Model Summary: G

The International Court of Justice at The Hague, acts as a world court. It issues rulings on international law disputes submitted to it by any state. It is made up of fifteen permanent judges, plus two more nominated by the parties involved. It does not have any powers to enforce its orders, although its decisions have traditionally carried some diplomatic weight. Its rulings sometimes fuel wider diplomatic debates. The Court may also appoint an independent tribunal such as the international war crimes tribunal in 1991, charged with attempting to prosecute perpetrators of murder, rape, and enforced expulsions in former Yugoslavia. Eleven judges of the tribunal were sworn in at the International Court of Justice. The types of international law cases heard by the Court are varied. Libya appealed unsuccessfully to the Court to halt Britain and US demands for punitive action and extradition of the Libyans alleged to have carried out the 1998 bombing of a Pan Am transport over Lockerbie, Scotland. The Court persuaded Britain and Albania to settle long standing diplomatic and legal disputes over the loss of two destroyers and 44 lives when Albanian-laid mines struck the ships. It has also ruled on disputes over borders and territorial waters, citizenship rights, the placement of embargoes, and the liability of a state using its military facilities to attack another state's commercial aircraft. The Court allowed British Petroleum to argue its case when Iran nationalized foreign oil interests and overruled a UN decision declaring Namibia under South African control.

Fig. 4.26. Model summary, created by G, for d376e

Model Summary: H

The International Court of Justice in the Hague, Netherlands, also known as "The World Court" is made up of 15 permanent judges and a further two nominated by the parties involved. It has no enforcement powers, although its decisions traditionally carry some diplomatic weight. In 1948 the court ordered Albanian compensation for two British warships destroyed by mines in Corfu Channel. The court heard mutual Serbian and Bosnian accusations, and ordered both to stop genocide and ethnic cleansing. The court resolved border disputes between Chad and Libya, and Honduras and El Salvador. The court overruled a 1960 U.N. resolution refuting South African control over Namibia, and reversed itself in 1967. Libya asked the court to make Britain and the U.S. hand over evidence incriminating Libya in airliner bombings over Scotland and Niger. It rejected Libya's requested injunction against Britain and the U.S. "taking any action against Libya calculated to coerce or compel Libya to surrender the accused individuals to any jurisdiction outside of Libya", and endorsed U.N. sanctions against Libya. The court heard a case by British Petroleum against Iran's nationalization of all foreign oil interests in 1951. In 1989 Iran asked the court to declare unlawful the 1988 U.S. shoot-down of Iran Air 655 and make the U.S. compensate Iran directly, not individual Iranians. In 1979, the court ordered Iran to release the American hostages in Tehran. Nicaragua asked the court to rule U.S. aid to the Contras illegal. The court ruled illegal the U.S. mining of Nicaraguan harbors.

Fig. 4.27. Model summary, created by H, for d376e

Model Summary: I

The World Court is another term for the International Court of Justice in The Hague. It has 15 permanent judges, plus a further two nominated by the parties involved. The Peace Palace is the court's seat. The World Court has no enforcement powers. Its decisions traditionally carry some diplomatic weight, although some countries refuse to recognize its authority. It provided a venue for the UN-sponsored Yugoslav war crimes tribunal in 1993. The World Court hears and rules on disputes between countries and violations of international law, such as the British/US-Libyan dispute over extradition of Libyans charged with bombing the Pan Am jet blown up over Lockerbie. The court also considered reciprocal Bosnian-Serbian accusations of genocide. It ruled on border disputes between Honduras and El Salvador and between Chad and Libya. It ordered Albania to compensate Britain for losses when British destroyers struck Albanian-laid mines in the Corfu Channel, heard Iran's suit that the US compensate Iran for an Iranian airliner mistakenly shot down by the US, and heard British Petroleum's arguments when Iran's Massadeq nationalized foreign oil interests. It heard Portugal's challenge of Australia's right to enter an accord with Indonesia over Timor. It overruled a UN resolution condemning South African control over Namibia but reversed itself 7 years later. It heard US appeals for release of hostages held by Iran. It heard Nicaragua's claim that the US illegally aided the Contras and condemned the US embargo and harbor-mining of Nicaragua, prompting the US to walk out in protest.

Fig. 4.28. Model summary, created by I, for d376e

Model Summary: J

The World Court, also known as the International Court of Justice, is headquartered in The Hague, Netherlands. Sixteen permanent judges preside in the Peace Palace. The Court does not have the powers to enforce its decisions, but they usually carry diplomatic weight. In the early 1990's, the Court also hosted the UN sponsored international war crimes tribunal, trying those accused of murder and other atrocities in the former Yugoslavia. The Court hears cases involving disagreements between and among nations. Military disputes are very common cases. Albania was fined when British destroyers hit Albanian mines in the Corfu Channel in 1948. Iran tried to sue the US for downing an Iranian airliner with a missile in 1988. The court has settle border disputes, including the El Salvador-Honduras dispute begun in 1969 and a 20-year dispute between Chad and Libya. Actions by world powers against smaller nations also have been tried. These include USSR embargos against Lithuania and US support of the Contras, including mining Nicaraguan harbors. They also hear disagreements over rightful leadership such as who should head Namibia in 1960, the legal administration of Timor in the early 1990's, and Bosnian claims of Serbian annexation attempts in 1993. They did not accept a case on the coup against Noriega of Panama in 1989. Other notable cases include the Iranian nationalization of foreign oil interests in 1951, the US petition for release of the embassy hostages in Iran, and Libyan efforts to circumvent prosecutions of Libyan airline bombers in the mid-1980's.

Fig. 4.29. Model summary, created by J, for d376e

Fig. 4.30 gives the query statement for set d436j, which asks for a *general* summary, Fig. 4.31 shows the system output, and Fig. 4.32 to Fig. 4.35 provide the human summaries. In this example, we obtained a *bad* result (ROUGE-2 score of 0.01192 and ROUGE-SU4 score of 0.06281): when compared to model summaries (H, I, J), the ROUGE-2 score is 0.01064 and the ROUGE-SU4 is 0.06024; when compared to model summaries (G, I, J), the ROUGE-2 score is 0.01189 and the ROUGE-SU4 is 0.06627; when compared to model summaries (G, H, J), the ROUGE-2 score is 0.01060 and the ROUGE-SU4 is 0.06200; when compared to model summaries (G, H, I), the ROUGE-2 score is 0.01453 and the ROUGE-SU4 is 0.06272.

```
Query Statement
<topic>
  <num> d436j </num>
  <title> Reasons for Train Wrecks </title>
  <narr>
    What causes train wrecks and what can be done to prevent them? Train wrecks
    are those events that result in actual damage to the trains themselves not just
    accidents where people are killed or injured.
  </narr>
  <granularity> general </granularity>
</topic>
```

Fig. 4.30. Query statement for d436j

System Summary (~250 words)

An Amtrak passenger train crash that killed three people and injured dozens on a foggy day last December was caused by the driver of a speeding truck that smashed through warning gates and hit the train, investigators said Tuesday. A crowded commuter train slammed into the rear of another early Friday, killing five people and injuring 70 others, a Brazil railway spokeswoman said. Two passenger trains collided head-on in thick morning fog near Hanover, killing three people and injuring 37. DRIVER ERROR was to blame for the Cannon Street rail crash which killed two people and injured more than 500 according to the official report published yesterday. A high-speed train plowed into the back of a slow-moving train in thick fog on the outskirts of Warsaw today, killing 16 people and injuring more than 50 officials said. An engine pulling freight cars rammed an Amtrak train that was stopped at a station Tuesday, injuring 31 people, officials said. A TRAIN driver was yesterday blamed for the rail crash in which he and three others were killed and 22 were injured at Newton, near Glasgow, last year. Two people were killed and 44 injured when two commuter trains collided head-on in Britain's second fatal rail crash in three days. Federal investigators began to probe the charred and twisted wreckage Wednesday in their effort determine why a truck drove onto the tracks in front of a high-speed passenger train in a crash that killed three and injured at least 55. A 20-year

Fig. 4.31. System summary for d436j

Model Summary: G

Train wrecks are caused by a number of factors: human, mechanical and equipment errors, spotty maintenance, insufficient training, load shifting, vandalism, and natural phenomenon. The most common types of mechanical and equipment errors are: brake failures, signal light and gate failures, track defects, and rail bed collapses. Spotty maintenance is characterized by failure to consistently inspect and repair equipment. Lack of electricians and mechanics results in letting equipment run down until someone complains. Engineers are often unprepared to detect or prevent operating problems because of the lack of follow-up training needed to handle updated high technology equipment. Load shiftings derail trains when a curve is taken too fast or there is a track defect. Natural phenomenon such as heavy fog, torrential rain, or floods causes some accidents. Vandalism in the form of leaving switches open or stealing parts from them leads to serious accidents. Human errors may be the most common cause of train accidents. Cars and trucks carelessly crossing or left on tracks cause frequent accidents. Train crews often make inaccurate tonnage measurements that cause derailments or brake failures, fail to heed single-track switching precautions, make faulty car hook-ups, and, in some instances, operate locomotives while under the influence of alcohol or drugs. Some freak accidents occur when moving trains are not warned about other trains stalled on the tracks. Recommendations for preventing accidents are: increase the number of inspectors, improve emergency training procedures, install state-of-the-art warning, control, speed and weight monitoring mechanisms, and institute closer driver fitness supervision.

Fig. 4.32. Model summary, created by G, for d436j

Model Summary: H

Causes of train accidents found by investigations fall into these categories. Faulty or damaged equipment include track, wheels, breaks, signaling equipment and cargo braces. Human factors include driver errors; not complying with light signals; signaling procedure errors; improper coupling of cars and engines; improper breaks connections; inaccurate cargo manifests; poor communication among train crews; poor communication between police and railway operators; improper switch operation; vehicle error at crossings; and improper maintenance. Substance abuse includes operating a train under the influence of drugs or alcohol. Management errors include failure to train drivers in difficult maneuvers and emergency procedures, and failure to systematically implement existing policies for identification, management and monitoring of hazards. Under-funding delayed implementation of safety measures, such as installation of "Automatic Train Protection" systems. Natural causes include floods and ice. Other causes include poor track design, such as conversion of double track to single track for bidirectional travel; vandalism; and pedestrians on tracks, including suicides. Formal recommendations for avoiding train accidents include replacing older trains with new ones; putting data recorders on all trains; improving driver emergency training, especially regarding 1) past accident scenarios, 2) light signal compliance, and 3) emergency braking; improving accuracy of cargo weight listings; improving communications among crews members; installing "Automatic Train Protection" systems; redesigning buffer stops; supervising drivers' fitness for work; making it illegal for railway workers with safety responsibilities to be impaired by alcohol or drugs; allowing drug testing after incidents; and investigating problems of sudden light changes between outdoors and illuminated stations.

Fig. 4.33. Model summary, created by H, for d436j

Model Summary: I

Human errors that cause train wrecks occur in coupling, braking, setting or working on switches, and misreading or disobeying signals. Marijuana or alcohol can be a factor. Collisions with other trains result from runaway cars, switching or signal errors, or slow-moving or stalled trains. Collisions with vehicles at crossings result from drivers disobeying or not seeing warning signals, vehicles getting stuck on tracks, or suicides. Emergency dispatchers may fail to notify railroads of obstructed track. Equipment problems causing wrecks include malfunctioning brakes, signal failures, warped or cracked track, faulty cross-braces, and defective wheel systems. Management practices contributing to train wrecks include insufficient attention to maintenance, training, regulation enforcement, and the possibility of an emergency. Risky cost-cutting measures, corruption, and bad employee morale also contribute. Other factors contributing to train wrecks include fog, railbeds and supports weakened by tunneling or rains, too few railroad inspectors, aging equipment, and vandalism. A systematic approach to identifying, managing and monitoring hazards could check errors and prevent accidents. Training is needed to keep people current on new equipment, safety and communication requirements, and all aspects of train handling. Supervisors need to ensure employees are fit for work. Equipment can be updated and crossings and buffer stops redesigned. More inspectors can be hired. Computerized systems such as Automatic Train Protection can take over if sensing a wrong decision or non-response to signals, and can adjust a train's speed on entering a station. A special switching mechanism in a trainyard can automatically derail loose cars.

Fig. 4.34. Model summary, created by I, for d436j

Model Summary: J

Multiple reasons cause train wrecks. Weather, including fog and torrential rains, can be a cause. Human factors are a common cause. Vehicle drivers ignore or try to out run crossing signals or break through crossing gates. Engineers ignore or fail to see signals directing them to stop their train. Both operators can be under the influence of alcohol or marijuana. Cars can be improperly coupled and break free. Mechanical and equipment failures are often the cause. Brakes can be faulty or disconnected, tracks can be warped or cracked; wheels can be faulty; individual cars may have faulty braces; or switches may malfunction. Signals may fail to operate properly, especially dangerous during single-track operations. In South Korea, accidents were blamed on lax safety standards, non-enforcement of regulations, and corruption. Combining modern faster equipment with older slower trains can result in rear end collisions. Vandalism continues to be a problem. Solutions include replacing older equipment, installing data recorders, spreading passengers out through trains, and improving buffering in stations. Drivers' training should be reviewed and they should be given follow-on training including practice in emergency situations. Drivers' fitness should be determined and alcohol and drug impairment should be illegal and tested for after an incident. Regulations to improve crew communication were called for. British Rail proposes installing an Automatic Train Protection system to take over for the engineer if signals or speed directions are not followed. Amtrak was planning special mechanisms to derail runaway cars before collisions. Track inspections remain vital.

Fig. 4.35. Model summary, created by J, for d436j

4.4 Discussion

This section provides general discussion on the proposed summarization approach.

4.4.1 Query relevance analysis by latent semantic analysis

Latent semantic analysis (LSA) is a technique to extract the inherent latent structure in word usage, based on the dependencies between terms [32]. In practice, it uses singular value decomposition (SVD) to discover the inter-relationships between terms and creates a reduced semantic space in which words that occur in similar contexts are near each other. Thus, it is possible to retrieve a passage (i.e., a sentence in this study), even if the query and the passage share no words in common.

The major advantages of LSA are twofold, as outlined in [81]. First, it deals with synonymy automatically without the use of any external dictionaries, thesauri, or knowledge base. Second, the learned reduced semantic space (constructed from term inter-relationships) is specific to the domain of interest.

LSA has recently proven profitable to in information retrieval (e.g., [19], [32], [35]). However, there are still a number of drawbacks related to LSA. First, even though LSA is statistically based on the co-occurrences of terms in the data, the resulting semantic space can only be justified on the mathematical level, but has no interpretable meaning in natural language, due to its unsatisfactory statistical foundation. Second, the purpose of dimension reduction is to relate words (or sentences) to k latent semantics. However, there is no obvious way to suggest a good value of k since the choice of k dimensions of the reduced space depends much on different applications and can only determined empirically.

4.4.2 The use of sentence-specific features

In this study, we had tested six surface-level features to assess the strength of representative power (or the informativeness) of each sentence, so as to help improve the performance of query-focused summarization. These features include: *position*, *avg. TF-IDF weight*, *similarity with title*, *similarity with document centroid*, and *similarity with topic centroid*, and (6) *num. of named entities*. It could be observed from the evaluation results, as shown in Table 4.3 and Table 4.4, that a scoring function, which takes into account both the degree of relevance of sentences to the query and the feature score of the sentences, can improve the performance of query-focused summarization. Nevertheless, it is worth studying to discover more features that are advantageous to the task. This issue is left as an open question, since to examine the whole feature space is not straightforward. In addition, it is also interesting to investigate the relations between different features for feature selection.

4.4.3 MMR vs. Modified MMR

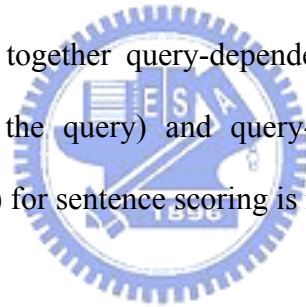
Maximal marginal relevance (MMR) is originally proposed to combine query-relevance with information-novelty in information retrieval [20]. The method tries to reduce redundancy while maintaining query relevance in re-ranking retrieved documents. In text summarization, passages (e.g., sentences) are re-ranked based on not only the relevance of passages to the query but also the redundancy among passages. Note that MMR is specific to query-focused summarization, in contrast to cross-sentence informational subsumption (CSIS) [111], which is designed for query-independent generic summaries.

The proposed modified MMR in Section 4.2.4 takes MMR one step further by enhancing it with the consideration of feature scores of a sentence, which have been

profitably used to determine the informativeness of sentences in summarization. The evaluation results, as shown in Table 4.3 and Table 4.4, roughly give the idea that the modified MMR is a suitable method for query-focused multidocument summarization.

4.4.4 The proposed summarization method

The combination of (1) the degree of relevance of a sentence to the query, and (2) the informativeness of a sentence, to measure the likelihood of sentences of being part in the summary, has shown promising according to the evaluation results. This suggests that it is reasonable for query-focused summarization to prior extract sentences with high relevance to the query and high feature scores. In fact, the proposed scoring model to takes into account together query-dependent feature (e.g., the degree of relevance of a sentence to the query) and query-independent feature (e.g., the informativeness of a sentence) for sentence scoring is relatively new in the field.



Chapter 5

Conclusions

This thesis discusses work on multidocument summarization (see Chapter 3) and query-focused multidocument summarization (see Chapter 4). The first is to produce a generic summary of a set of topically-related documents. The second, a particular task of the first, is, given a user query, to generate a query-focused summary which reflects particular points that are relevant to the user's desired topic(s) of interest. Both tasks are addressed in this thesis using the most common technique for summarization, namely sentence extraction: important sentences are identified and extracted verbatim from documents and composed into an extractive summary.

In this chapter, we summarize the extraction-based summarization framework proposed in this study, describing the benefits and limitations, present our contributions to the field, and finally provide possible directions for future work.

5.1 Multidocument Summarization Framework

This thesis has proposed an extraction-based summarization framework, as shown in Fig. 5.1, for the creation of generic and query-focused summaries of multiple documents. Note that Fig. 5.1 is the union of Fig. 3.1 and Fig. 4.1. In the figure, the “ * ” symbol indicates that the input/output or the module is specific to multidocument summarization, while the “ † ” symbol denotes that the input/output or the module is designed for query-focused multidocument summarization. The whole summarization process can be decomposed into three phases: (1) the *preprocessing* phase preprocesses the input documents and the query statement if given, (2) the *sentence scoring/ranking* phase scores sentences and ranks them according to their

likelihood of being part of the summary, and (3) the *summary production* phase extracts important sentences to create a summary. The details are presented and discussed in Chapter 3 and Chapter 4.

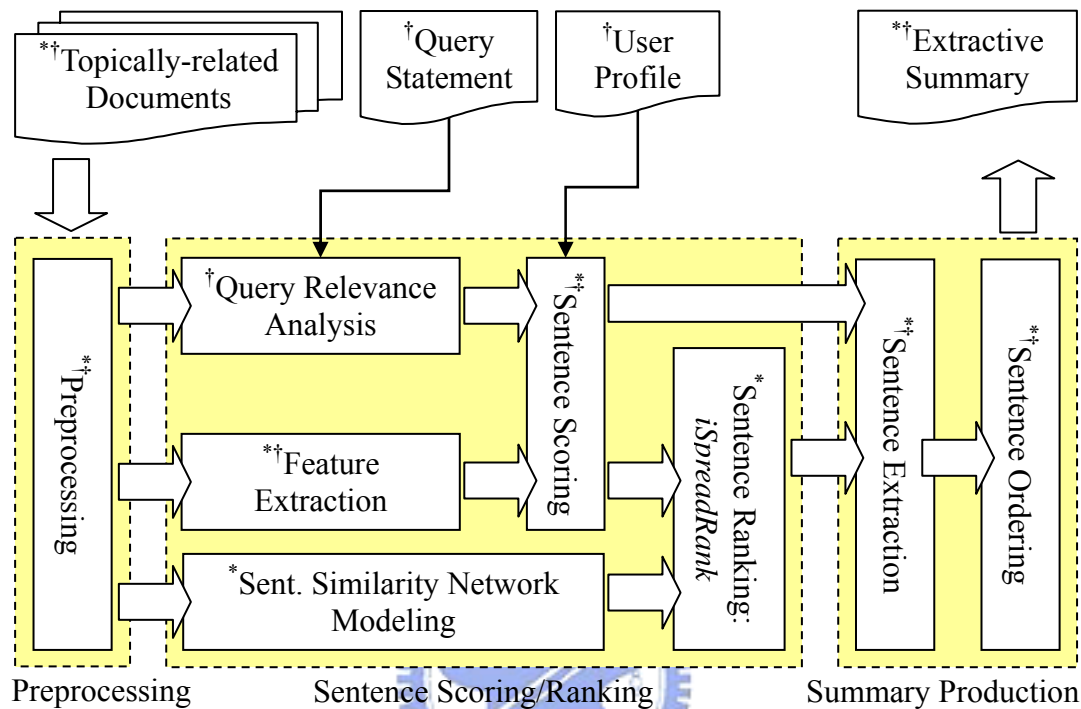
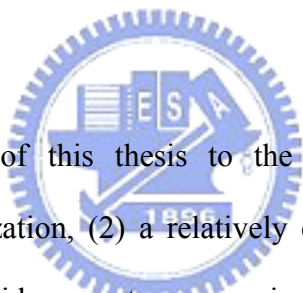


Fig. 5.1. Proposed framework for extraction-based multidocument summarization

The proposed summarization framework has several benefits. First, it is in an unsupervised manner, and therefore no training data is required. Second, it is domain- and language-independent since it takes into account neither domain-specific knowledge nor deep linguistic analysis particular to languages. Hence, it is relatively easy to use the summarization approach as a base prototype in any domains and for documents in any languages. Third, it is flexible and extensible due to the underlying modulization design. For instance, other surface-level features can be added to help measure the importance of sentences. Finally, the core module of sentence scoring/ranking makes it adaptive to produce either short or long summaries in different sizes, based on a ranking over all sentences.

There exist some limitations, even though the proposed summarization framework has proven successful to a degree by the evaluation on the DUC 2004 and DUC 2005 data sets. It is essentially a surface-level approach based on the use of features to recognize important sentences (see Section 1.1.3 for the categorization of summarization techniques). Hence, there is neither deep analysis of natural language processing performed, discourse structure considered, nor domain-specific knowledge involved in summarization, leading to the bad understanding of the input texts. On the other hand, the strategy of sentence extraction may include good content in the summaries. However, it does not guarantee good summary quality in terms of coherence, cohesion, and overall organization.

5.2 Contributions



The principal contributions of this thesis to the field include: (1) an overall introduction to text summarization, (2) a relatively complete survey of the current state of the art in multidocument summarization, (3) a general-purpose extraction-based summarization framework for producing generic and query-focused summaries of multiple documents, (4) a discussion on the proposed summarization framework in characteristics, benefits and limitations, and (5) case studies of the proposed summarization framework on the DUC 2004 and DUC 2005 data sets.

In the following, we outline the contributions with respect to the proposed summarization methods.

(1) Multidocument summarization:

Chapter 3 proposes a novel graph-based sentence ranking method, iSpreadRank, to rank sentences according to their likelihood of being part of the

summary. The input set of documents are modeled as a sentence similarity network. A feature profile is created to capture the values of surface-level features of all the sentences and the feature scores serve as the initial importance of nodes in the network. To reason the relative importance of sentences, iSpreadRank practically applies spreading activation to iteratively re-weight the importance of sentences by collecting the importance propagated from their connected nodes as a function of the importance of the connected nodes and the strength of relationships between nodes. iSpreadRank, in fact, operates like a semi-supervised learning process in which the initial labeling of every sentence is determined by its feature score, and the final labeling of sentences is based on the feature scores of sentences and the relationships between them.

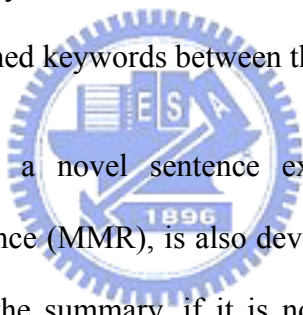
For summarization, a sentence extraction method, based on cross-sentence informational subsumption (CSIS) for redundancy filtering, iteratively extracts one sentence at a time into the summary, which not only has high importance but also has less redundancy than the other sentences extracted prior to it. Finally, a sentence ordering policy, which considers together topical relatedness and chronological order between sentences, is employed to organize extracted sentences into a coherent summary.

The proposed summarization method is evaluated in a case study with the DUC 2004 data set and found to perform well in various ROUGE measures. Experimental results show that the proposed method performs competitively to the top systems at DUC 2004.

(2) Query-focused multidocument summarization:

Chapter 4 proposes a new scoring method, which combines (1) the degree

of relevance of a sentence to the query, and (2) the informativeness of a sentence, to measure the likelihood of sentences of being part in the summary. The degree of query relevance of a sentence is assessed as the similarity between the sentence and the query computed in a latent semantic space, and the informativeness of a sentence is estimated using surface-level features. While most research works have mainly focused on the identification of query-biased sentences, our idea to takes into account together query-dependent feature (e.g., the degree of relevance of a sentence to the query) and query-independent feature (e.g., the informativeness of a sentence) is relatively new. Furthermore, the proposed use of latent semantic analysis (LSA) can potentially relate a sentence and the query semantically and hence obtains a better estimate of the similarity, even the number of matched keywords between them is not significant.



For summarization, a novel sentence extraction method, inspired by maximal marginal relevance (MMR), is also developed to iteratively extract one sentence at a time into the summary, if it is not too similar to any sentences already extracted. In one iteration, all the remaining unselected sentences are re-scored and ranked using a modified MMR function, so as to extract the sentence with the highest score. Finally, the extracted sentences are concatenated in chronological order to form the output summary.

The proposed summarization method is evaluated in a case study with the DUC 2005 data set and found to perform well in various ROUGE measures. Experimental results show that the proposed method performs competitively to the top systems at DUC 2005.

5.3 Future Work

There is still much work that should be done in the future. This section outlines future work that could extend the applicability and performance of the proposed summarization framework, including:

- (1) The employment of natural language processing (NLP) techniques:

This thesis does not have much NLP techniques to help understand and analyze the input texts. We suppose that the use of NLP techniques, for example, information extraction, sentence parsing, lexical chains, co-reference chains, etc., can directly benefit the identification of text entities and their relationships, and hence leads to better understanding of texts and content selection.

- (2) The utilization of domain knowledge and external resources:

This thesis uses no domain knowledge in the summarization process since it targets at general-purpose summarization of documents in public domains. Such a summarization framework would apparently not work well in all domains. It is expected that domain knowledge and external resources, such as patient record and disease information in medical domains, and terminologies in financial and sports areas, should improve the analysis of texts in particular domains.

- (3) The consideration of language properties:

This thesis performs no deep linguistic analysis particular to languages and thus the proposed framework can be practically applied to documents in any languages, for which the decomposition of texts into word units in preprocessing can be done. It is believed that the framework is applicable to

multilingual/cross-lingual multidocument summarization as well, provided that machine translation modules are available, or the relevance of text portions in different languages can be determined.

(4) The investigation of new similarity measures:

This thesis exploits the cosine similarity metric (in vector space model and in latent semantic space) to measure the relations between each pair of sentences, as well as the relevance between a sentence and the query. We expect that more advanced techniques of assessing similarity, which incorporate word semantics and relations, will be easily integrated in the summarization model. In addition, using words or phrases with similar meanings to expand the user query will obviously profit the identification of query-biased sentences.

(5) The exploration of new surface-level features:

This thesis examines a subset of surface-level features and various combinations of them to determine the informativeness of sentences (or the likelihood of sentences of being part of the summary) in summarization. Nevertheless, it is worth studying to discover other effective features, to identify the effect of a feature to summarization, as well as to investigate the relations between different features for feature selection.

(6) The application of machine learning techniques:

This thesis combines different features in an unsupervised manner to yield a sentence scoring function, for which parameters are tuned empirically. As more and more standard collections for training and test on evaluation of

summarization methods have been established recently, we intend to apply machine learning techniques to automatically learn an effective sentence scoring model from training data.

(7) The improvement of the summary quality:

This thesis adopts the most common technique in summarization, namely sentence extraction, to create extractive summaries. However, this strategy does not guarantee good summary quality in terms of coherence, cohesion, and overall organization, even though it may include good content in the summaries. Fortunately, techniques to improve the quality of summaries, such as, information fusion and reformulation by natural language generation to produce abstractive summaries, passage simplification/compression to remove parts of, for example, a sentence without disturbing its understandability or underlying meaning, information ordering to yield coherent summaries, and anaphora resolution and time annotation to produce summaries with good readability, have proven successful in some degree.

(8) The use of different strategies for different types of the input document clusters:

This thesis uses the same strategy to deal with different types of the input document clusters. Such a single strategy has shown promising in evaluation. However, we believe that a first step to examine the types of the document clusters, and then to process the documents using different strategies will probably generate better summaries. For instance, news articles can be classified into on the same event, on topically related but different events, natural disaster, biography, etc., for which different summarization strategies should be decided.

(9) The enhancement of visualization:

This thesis does not provide visualization of summaries. Obviously, it could be beneficial to the user by presenting visual information related to the content in summaries. The following gives some visualization examples in news summarization: the visualization of the spatial information, indicated in the summary, on a geographical map; a visual summary with the x-axis representing the timeline, the y-axis representing the location, and the (x, y) point labeled with keywords of news events, linking to the corresponding text summary.

(10) The addition of user interaction mechanisms:

This thesis only provides the user with simple controls, such as the length of the summary, and the summary type in generic and query-focused, over the summarization process. One shortage of such a system is the lack of dynamic response to the user's need. Therefore, future work will add user interaction mechanisms into the proposed summarization framework. For instance, the linking of a summary sentence to the original document or to the most relevant sentences in the documents; the zoom-in and zoom-out of topics of a summary in the hierarchical structure; the control to obtain preferred summaries by relevance feedback of user interests.

Bibliography

- [1] Afantenos, S., Karkaletsis, V., & Stamatopoulos, P. (2005). Summarization from medical documents: a survey. *Artificial Intelligence in Medicine*, 33(2), 157-177.
- [2] Allan, J., Wade, C., & Bolivar, A. (2003). Retrieval and novelty detection at the sentence level. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 314-321). Toronto, ON, Canada.
- [3] Amigó, E., Gonzalo, J., Peinado, V., Peñas, A., & Verdejo, F. (2004). An empirical study of information synthesis tasks. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics* (pp. 207-214). Barcelona, Spain.
- [4] Anderson, J. R. (1983). A spreading activation theory of memory. *Journal of Verbal Learning and Verbal Behavior*, 22, 261-295.
- [5] Aone, C., Okurowski, M. E., & Gorfinsky, J. (1998). Trainable, scalable summarization using robust NLP and machine learning. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics* (pp. 62-66). Montreal, QC, Canada.
- [6] Azzam, S., Humphreys, K., & Gaizauskas, R. (1999). Using coreference chains for text summarization. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics* (pp. 77-84). College Park, MD, USA.
- [7] Barzilay, R., & Elhadad, M. (1997). Using lexical chains for text summarization. In *Proceedings of the ACL'97/EACL'97 workshop on intelligent scalable text summarization* (pp. 10-17). Madrid, Spain.
- [8] Barzilay, R., Elhadad, N., & McKeown, K. R. (2002). Inferring strategies for sentence ordering in multidocument news summarization. *Journal of Artificial Intelligence Research*, 17, 35-55.
- [9] Barzilay, R., McKeown, K. R., & Elhadad, M. (1999). Information fusion in the context of multi-document summarization. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics* (pp. 550-557). College Park, MD, USA.
- [10] Berger, A., & Mittal, V. O. (2000). Query-relevant summarization using FAQs. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics* (pp. 294-301). Hong Kong, China.
- [11] Bergler, S., Witte, R., Li, Z., Khalife, M., Chen, Y., Doandes, M., & Andreevskaia, A. (2004). Multi-ERSS and ERSS 2004. In *Proceedings of the DUC 2004*. Boston, MA, USA.
- [12] Blair-Goldensohn, S. (2005). From definitions to complex topics: Columbia University at DUC 2005. In *Proceedings of the DUC 2005*. Vancouver, BC,

Canada.

- [13] Bollen, J., Vandesompele, H., & Rocha, L. M. (1999). Mining associative relations from website logs and their applications to context-dependent retrieval using spreading activation. In *Proceedings of the Workshop on Organizing Web Space*. Berkeley, CA, USA.
- [14] Boros, E., Kentor, P. B., & Neu, D. J. (2001). A clustering based approach to creating multi-document summaries. In *Proceedings of the DUC 2001*. New Orleans, LA, USA.
- [15] Brunn, M., Chali, Y., & Pinchak, C. J. (2001). Text summarization using lexical chains. In *Proceedings of the DUC 2001*. New Orleans, LA, USA.
- [16] Borko, H., & Bernier, C. (1975). *Abstracting concepts and methods*. Academic Press, NY: New York.
- [17] Bosma, W. (2005). Query-Based Summarization Using Rhetorical Structure Theory. In *Proceedings of the 15th Meeting of Computational Linguistics in the Netherlands* (pp. 29-44). Leiden, Netherlands.
- [18] Brin, S., & Page, L. (1998). The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems*, 30(1-7), 107-117.
- [19] Caid, W. R., Dumais, S. T., & Gallant, S. I. (1995). Learned vector space models for information retrieval. *Information Processing & Management*, 31(3), 419-429.
- [20] Carbonell, J., & Goldstein, J. (1998). The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 335-336). Melbourne, Australia.
- [21] Chen, H.-H., Kuo, J.-J., Huang, S.-J., Lin, C.-J., & Wang, H.-C. (2003). A summarization system for Chinese news from multiple sources. *Journal of American Society for Information Science and Technology*, 54(13), 1224-1236.
- [22] Chen, Y.-M., Wang, X.-L., & Liu, B.-Q. (2005). Multi-document summarization based on lexical chains. In *Proceedings of the 4th International Conference on Machine Learning and Cybernetics* (pp. 1937-1942). Guangzhou, China.
- [23] Collins, A. M., & Loftus, E. F. (1975). A spreading-activation theory of semantic processing. *Psychological Review*, 82(6), 407-428.
- [24] Conroy, J. M., Schlesinger, J. D., & O'Leary, D. P. (2006). Topic-focused multi-document summarization using an approximate oracle score. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics (COLING/ACL 2006)* (pp. 152-159). Sydney, Australia.
- [25] Crammer, K., and Singer, Y. (2002). PRanking with ranking. *Advances in Neural Information Processing Systems*, 14, 641-647.
- [26] Cremmins, E. T. (1996). *The art of abstracting*. Information Resources Press, VA: Arlington.

- [27] D'Avanzo, E., & Magnini, B. (2005). A keyphrase-based approach to summarization: The LAKE system at DUC-2005. In *Proceedings of the DUC 2005*. Vancouver, BC, Canada.
- [28] Dang, H. T. (2005). Overview of DUC 2005. In *Proceedings of the DUC 2005*. Vancouver, BC, Canada.
- [29] Daniel, N., Radev, D., & Allison, T. (2003). Sub-event based multi-document summarization. In *Proceedings of the HLT-NAACL'03 Workshop on Text Summarization* (pp. 9-16). Edmonton, AB, Canada.
- [30] Daumé III, H., Echihabi, A., Marcu, D., Munteanu, D. S., & Soricut, R. (2002). GLEANS : A generator of logical extracts and abstracts for nice summaries. In *Proceedings of the DUC 2002*. Philadelphia, PA, USA.
- [31] Daumé III, H., & Marcu, D. (2006). Bayesian Query-Focused Summarization. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics* (pp. 305-312). Sydney, Australia.
- [32] Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., & Harshman, R. (1990). Indexing by latent semantic analysis, *Journal of the American Society for Information Science*, 41(6), 391-407.
- [33] DeJong, G. F. (1982). An overview of the FRUMP system. In W. G. Lehnert, & M. H. Ringle (Eds.), *Strategies for natural language processing*. Hillsdale, NJ: Lawrence Erlbaum.
- [34] DUC (Document Understanding Conferences): <<http://duc.nist.gov/>>.
- [35] Dumais, S. T., Landauer, T. K., & Littman, M. L. (1996). Automatic cross-linguistic information retrieval using latent semantic indexing. In *Proceedings of SIGIR'96 Workshop on Cross-Linguistic Information Retrieval* (pp. 16-23). Zurich, Switzerland.
- [36] Edmundson, H. P. (1969). New methods in automatic extracting. *Journal of the ACM*, 16(2), 264-285.
- [37] Elhadad, N. & McKeown, K. R. (2001). Towards generating patient specific summaries of medical articles. In *Proceedings of the NAACL 2001 Workshop on Automatic Summarization*. Pittsburgh, PA, USA.
- [38] Erkan, G. (2006). Using biased random walks for focused summarization. In *Proceedings of the DUC 2006*. Brooklyn, NY, USA.
- [39] Erkan, G., & Radev, D. R. (2004). LexRank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, 22, 457-479.
- [40] Fish, S., & Roark, B. (2006). Query-focused summarization by supervised sentence ranking and skewed word distributions. In *Proceedings of the DUC 2006*. Brooklyn, NY, USA.
- [41] Fuentes, M., Alfonseca, E., & Rodríguez, H. (2007). Support vector machines for query-focused summarization trained and evaluated on Pyramid data. In

Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL 2007) (pp. 57-60). Prague, Czech Republic.

- [42] Fum, D., Guida, G., & Tasso, C. (1985). Evaluating importance: a step towards text summarization. In *Proceedings of the 9th International Joint Conference on Artificial Intelligence* (pp. 840-844). Los Angeles, CA, USA.
- [43] Ge, J., Huang, X., & Wu, L. (2003). Approaches to event-focused summarization based on named entities and query words. In *Proceedings of the DUC 2003*. Edmonton, AB, Canada.
- [44] Goldstein, J., Mittal, V., Carbonell, J., & Kantrowitz, M. (2000). Multi-document summarization by sentence extraction. In *Proceedings of NAACL-ANLP 2000 Workshop on Automatic Summarization* (pp. 40-48). Seattle, WA, USA.
- [45] Gong, Y., & Liu, X. (2001). Generic text summarization using relevance measure and latent semantic analysis. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 19-25). New Orleans, LA, USA.
- [46] Gupta, S., Nenkova, A., & Jurafsky, D. (2007). Measuring importance and query relevance in topic-focused multi-document summarization. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL 2007)* (pp. 193-196). Prague, Czech Republic.
- [47] Hachey, B., Murray, G., & Reitter, D. (2005). The Embra system at DUC 2005: Query-oriented multi-document summarization with a very large latent Semantic space. In *Proceedings of the DUC 2005*. Vancouver, BC, Canada.
- [48] Hahn, U., & Mani, I. (2000). The challenges of automatic summarization. *Computer*, 33(11), 29-36.
- [49] Harabagiu, S., & Lacatusu, F. (2005). Topic themes for multi-document summarization. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 202-209). Salvador, Brazil.
- [50] Harabagiu, S. M., & Maiorano, S. J. (2000). Acquisition of linguistic patterns for knowledge-based information extraction. In *Proceedings of the 2nd International Conference on Language Resources & Evaluation*. Athens, Greece.
- [51] Harabagiu, S. M., & Lacatusu, F. (2002). Generating single and multi-document summaries with GIXTexter. In *Proceedings of the DUC 2002*. Philadelphia, PA, USA.
- [52] Harnly, A., Nenkova, A., Passonneau, R., & Rambow, O. (2005). Automation of summary evaluation by the Pyramid method. In *Proceedings of 2005 International Conference on Recent Advances in Natural Language Processing*. Borovets, Bulgaria.
- [53] Hatzivassiloglou, V., Klavans, J. L., Holcombe, M. L., Barzilay, R., Kan, M.-Y., & McKeown, K. R. (2001). SimFinder: a flexible clustering tool for summarization. In *Proceedings of NAACL Workshop on Automatic Summarization* (pp. 41-49). Pittsburgh, PA, USA.

- [54] Hirao, T., Takeuchi, K., Isozaki, H., Sasaki, Y., & Maeda, E. (2002). NTT/NAIST's Text Summarization Systems for TSC-2. In *Proceedings of the 3rd NTCIR Workshop on Research in Information Retrieval, Automatic Text Summarization and Question Answering* (pp. 13-18). Tokyo, Japan.
- [55] Hovy, E., & Lin, C.-Y. (1997). Automated text summarization in SUMMARIST. In *Proceedings of the ACL'97/EACL'97 Workshop on Intelligent Scalable Text Summarization* (pp. 18-24). Madrid, Spain.
- [56] Hovy, E., Lin, C.-Y., & Zhou, L. (2005). A BE-based multidocument summarizer with query interpretation. In *Proceedings of the DUC 2005*. Vancouver, BC, Canada.
- [57] Hovy, E., & Marcu, D. (1998). *Tutorial on automated text summarization*. Presented at COLING-ACL'98. Montreal, QC, Canada.
- [58] Huang, Z., Chen, H., & Zeng, D. (2004). Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering. *ACM Transactions on Information Systems*, 22(1), 116-142.
- [59] Jagarlamudi, J., Pingali, P., & Varma, V. (2005). A relevance-based language modeling approach to DUC 2005. In *Proceedings of the DUC 2005*. Vancouver, BC, Canada.
- [60] Jagarlamudi, J., Pingali, P., & Varma, V. (2006). Query independent sentence scoring approach to DUC 2006. In *Proceedings of the DUC 2006*. Brooklyn, NY, USA.
- [61] Kan, M.-Y., & Klavans, J. L. (2002). Using librarian techniques in automatic text summarization for information retrieval. In *Proceedings of the 2nd ACM/IEEE-CS Joint Conference on Digital Libraries* (pp. 36-45). Portland, OR, USA.
- [62] Kan, M.-Y., McKeown, K. R., & Klavans, J. L. (2001). Domain-specific informative and indicative summarization for information retrieval. In *Proceedings of the DUC 2001*. New Orleans, LA, USA.
- [63] Kan, M.-Y., McKeown, K. R., & Klavans, J. L. (2001). Applying natural language generation to indicative summarization. In *Proceedings of the 8th European Workshop on Natural Language Generation* (pp. 1-9). Toulouse, France.
- [64] Kleinberg, J. M. (1999). Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5), 604-632.
- [65] Kuo, J.-J., Wung, H.-C., Lin, C.-J., & Chen, H.-H. (2002). Multi-document summarization using informative words and its evaluation with a QA system. In *Proceedings of the 3rd International Conference on Intelligent Text Processing and Computational Linguistics* (Lecture Notes in Computer Science, 2276) (pp. 391-401). Mexico City, Mexico.
- [66] Kupiec, J., Pedersen, J., & Chen, F. (1995). A trainable document summarizer. In *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 68-73). Seattle, WA,

USA.

- [67] Lacatusu, F., Hickl, A., Roberts, K., Shi, Y., Bensley, J., Rink, B., Wang, P., & Taylor, L. (2006). LCC's GISTexter at DUC 2006: Multi-strategy multi-document summarization. In *Proceedings of the DUC 2006*. Brooklyn, NY, USA.
- [68] Lavrenko, V., & Croft, W. B. (2001). Relevance-based language models. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2001)* (pp. 120-127). New Orleans, LA, USA.
- [69] Lehnert, W. G. (1982). Plot units: a narrative summarization strategy. In W. G. Lehnert, & M. H. Ringle (Eds.), *Strategies for natural language processing* (pp. 375-412). Hillsdale, NJ: Lawrence Erlbaum.
- [70] Lenci, A., Bartolini, R., Calzolari, N., Agua, A., Busemann, S., Cartier, E., Chevreau, K., & Coch, J. (2002). Multilingual summarization by integrating linguistic resources in the MLIS-MUSI project. In *Proceedings of the 3rd International Conference on Language Resources and Evaluation* (pp. 1464-1471). Canary Islands, Spain.
- [71] Leuski, A., Lin, C.-Y., & Hovy, E. (2003). iNeATS: Interactive Multi-document Summarization. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics* (pp. 125-128). Sapporo, Japan.
- [72] Li, W., Li, W., Li, B., Chen, Q., & Wu, M. (2005). The Hong Kong Polytechnic University at DUC2005. In *Proceedings of the DUC 2005*. Vancouver, BC, Canada.
- [73] Li, S., Ouyang, Y., Sun, B., Guo, Z. (2006). Peking University at DUC 2006. In *Proceedings of the DUC 2006*. Brooklyn, NY, USA.
- [74] Li, S., Ouyang, Y., Wang, W., & Sun, B. (2007). Multi-document summarization using support vector regression. In *Proceedings of the DUC 2007*. Rochester, NY, USA.
- [75] Li, J., Sun, L., Kit, C., Webster, J. (2007). A query-focused multi-document summarizer based on lexical chains. In *Proceedings of the DUC 2007*. Rochester, NY, USA.
- [76] Lin, C.-Y. (1999). Training a selection function for extraction. In *Proceedings of the 8th International Conference on Information and Knowledge Management* (pp. 55-62). Kansas City, MO, USA.
- [77] Lin, C.-Y., & Hovy, E. (2002). From single to multi-document summarization: a prototype system and its evaluation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics* (pp. 457-464). Philadelphia, PA, USA.
- [78] Lin, C.-Y., & Hovy, E. (2002). NeATS in DUC 2002. In *Proceedings of the DUC 2002*. Philadelphia, PA, USA.
- [79] Lin, C.-Y., & Hovy, E. (2003). Automatic evaluation of summaries using N-gram co-occurrence statistics. In *Proceedings of the 3rd International Conference on*

Human Language Technology Research and 3rd Meeting of the North American Chapter of the Association for Computational Linguistics (pp. 71-78). Edmonton, AB, Canada.

- [80] Lin, C.-Y., & Hovy, E. (2003). The potential and limitations of automatic sentence extraction for summarization. In *Proceedings of the HLT-NAACL 03 on Text Summarization Workshop* (pp. 73-80). Edmonton, Canada.
- [81] Littman, M. L., Dumais, S. T., Landauer, T. K. (1996). Automatic cross-language information retrieval using latent semantic indexing. In *Proceedings of SIGIR'96 Workshop on Cross-Linguistic Information Retrieval* (pp. 16-23). Zurich, Switzerland.
- [82] Luhn, H. P. (1958). The automatic creation of literature abstracts. *IBM Journal of Research and Development*, 2(2), 159-165.
- [83] Lund, K., & Burgess, C. (1996). Producing high-dimensional semantic spaces from lexical co-occurrence. *Behavior Research Methods, Instrumentation, and Computers*, 28, 203-208.
- [84] Maña-López, M. J., Buenaga, M. D., & Gómez-Hidalgo, J. M. (2004). Multidocument summarization: An added value to clustering in interactive retrieval. *ACM Transactions on Information Systems*, 22(2), 215-241.
- [85] Mani, I. (2001). *Automatic Summarization*. Amsterdam, Netherlands: John Benjamins Pub Co.
- [86] Mani, I., & Bloedorn, E. (1999). Summarizing similarities and differences among related documents. *Information Retrieval*, 1(1-2), 35-67.
- [87] Mani, I., & Maybury, M. T. (Eds.). (1999). *Advances in automatic text summarization*. Cambridge, MA: The MIT Press.
- [88] Mann, W. C., & Thompson, S. A. (1988). Rhetorical structure theory: Toward a function theory of text organization, *Text*, 8(3), 243-281.
- [89] Marcu, D. (2000). *The theory and practice of discourse parsing and summarization*. Cambridge, MA: The MIT Press.
- [90] Mayeng, S. H., & Jang, D. (1999). Development and evaluation of a statistically based document summarization system. In I. Mani & M. T. Maybury (Eds.), *Advances in automatic text summarization* (pp. 61 - 70). Cambridge, MA: The MIT Press.
- [91] McDonald, D. M., & Chen, H. (2006). Summary in context: Searching versus browsing. *ACM Transactions on Information Systems*, 24(1), 111-141.
- [92] McKeown, K., Barzilay, R., Chen, J., Elson, D., Evans, D., Klavans, J., Nenkova, A., Schiffman, B., & Sigelman, S. (2003). Columbia's Newsblaster: New features and future directions. In *Proceedings of the 3rd International Conference on Human Language Technology Research and 3rd Meeting of the North American Chapter of the Association for Computational Linguistics* (pp. 15-16). Edmonton, AB, Canada.
- [93] McKeown, K. R., Barzilay, R., Evans, D., Hatzivassiloglou, V., Klavans, J. L.,

- Nenkova, A., Sable, C., Schiffman, B., & Sigelman, S. (2002). Tracking and summarizing news on a daily basis with Columbia's Newsblaster. In *Proceedings of the 2nd International Conference on Human Language Technology Research* (pp. 280-285). San Diego, CA, USA.
- [94] McKeown, K. R., Chang, S.-F., Cimino, J., Feiner, S., Friedman, C., Gravano, L., Hatzivassiloglou, V., Johnson, S., Jordan, A. D., Klavans, J. L., Kushniruk, A., Patel, V., & Teufel, S. (2001). PERSIVAL, a system for personalized search and summarization over multimedia healthcare information. In *Proceedings of the 1st ACM/IEEE-CS Joint Conference on Digital Libraries* (pp. 331-340). Roanoke, VA, USA.
- [95] McKeown, K., Hirschberg, J., Galley, M., & Maskey, S. (2005). From text to speech summarization. In *Proceedings of the 30th International Conference on Acoustics, Speech, and Signal Processing* (pp. 997-1000). Philadelphia, PA, USA.
- [96] McKeown, K. R., Klavans, J. L., Hatzivassiloglou, V., Barzilay, R., & Eskin, E. (1999). Towards multidocument summarization by reformulation: progress and prospects. In *Proceedings of the 16th National Conference on Artificial Intelligence* (pp. 453-460). Orlando, FL, USA.
- [97] McKeown, K., & Radev, D. R. (1995). Generating summaries of multiple news articles. In *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 74-82). Seattle, WA, USA.
- [98] Mihalcea, R. (2004). Graph-based ranking algorithms for sentence extraction, applied to text summarization. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics* (pp. 170-173). Barcelona, Spain.
- [99] Mihalcea, R., Corley, C., & Strapparava, C. (2006). Corpus-based and knowledge-based measures of text semantic similarity. In *Proceedings of the 21st National Conference on Artificial Intelligence*. Boston, MA, USA.
- [100] Mihalcea, R., & Tarau, P. (2005). An algorithm for language independent single and multiple document summarization. In *Proceedings of the 2nd International Joint Conference on Natural Language Processing* (pp. 19-24). Jeju Island, Korea.
- [101] Moens, M.-F., Uyttendaele, C., & Dumortier, J. (1999). Abstracting of legal cases: The potential of clustering based on the selection of representative objects. *Journal of the American Society for Information Science*, 50(2), 151-161.
- [102] Noble, B., & Daniel, J. W. (1988). *Applied linear algebra*. Englewood Cliffs, NJ: Prentice Hall.
- [103] Paice, C. D. (1990). Constructing literature abstracts by computer: Techniques and prospects. *Information Processing & Management*, 26(1), 171-186.
- [104] Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2001). BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics* (pp. 311-318).

Philadelphia, PA, USA.

- [105] Pirolli, P., Pitkow, J., Rao, R. (1996). Silk from a sow's ear: extracting usable structures from the Web. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 118-125). Vancouver, BC, Canada.
- [106] Quillian, M. R. (1968). Semantic memory. In M. Minsky (Ed.), *Semantic Information Processing* (pp. 227-270). Cambridge, MA: The MIT Press.
- [107] Radev, D. (2001). Tutorial: Text summarization. Presented at the *ACM SIGIR 2001*. New Orleans, LA, USA.
- [108] Radev, D. R., Blair-Goldensohn, S., Zhang, Z., & Raghavan, R. S. (2001). NewsInEssence: A system for domain-independent, real-time news clustering and multi-document summarization. In *Proceedings of the 1st International Conference on Human Language Technology Research*. San Diego, CA, USA.
- [109] Radev, D. R., Fan, W. & Zhang, Z. (2001). WebInEssence: A personalized Web-based multi-document summarization and recommendation system. In *Proceedings of NAACL 2001 Workshop on Automatic Summarization*. Pittsburgh, PA, USA.
- [110] Radev, D. R., Hovy, E., & McKeown, K. (2002). Introduction to the special issue on summarization. *Computational Linguistics*, 28(4), 399-408.
- [111] Radev, D. R., Jing, H., Styś, M., & Tam, D. (2004). Centroid-based summarization of multiple documents. *Information Processing & Management*, 40(6), 919-938.
- [112] Radev, D., Teufel, S., Saggion, H., Lam, W., Blitzer, J., Çelebi, A., Qi, H., Drabek, E., & Liu, D. (2001). Evaluation of text summarization in a cross-lingual information retrieval framework. *Technical report*. Center for Language and Speech Processing, Johns Hopkins University.
- [113] Rath, G. J., Resnick, A., & Savage, T. R. (1961). The formation of abstracts by the selection of sentences. *American Documentation*, 12(2), 139-141.
- [114] Rau, L. F., Jacobs, P. S., & Zernik, U. (1989). Information extraction and text summarization using linguistic knowledge acquisition. *Information Processing and Management*, 25(4), 419-428.
- [115] Reimer, U., & Hahn, U. (1988). Text condensation as knowledge base abstraction. In *Proceedings of the 4th Conference on Artificial Intelligence Applications* (pp. 338-344). San Diego, CA, USA.
- [116] Saggion, H., & Gaizauskas, R. (2004). Multi-document summarization by cluster/profile relevance and redundancy removal. In *Proceedings of the DUC 2004*. Boston, MA, USA.
- [117] Saggion, H., & Lapalme, G. (2002). Generating indicative-informative summaries with SumUM. *Computational Linguistics*, 28(4), 497-526.
- [118] Salton, G., & McGill, M. J. (1983). *Introduction to Modern Information Retrieval*. New York, NY: McGraw-Hill.

- [119] Salton, G., Singhal, A., Mitra, M., & Buckley, C. (1997). Automatic text structuring and summarization. *Information Processing & Management*, 33(2), 193-207.
- [120] Schiffman, B., Nenkova, A., & McKeown, K. (2002). Experiments in multidocument summarization. In *Proceedings of the 2nd International Conference on Human Language Technology Research*. San Diego, CA, USA.
- [121] Schilder, F., McCulloh, A., McInnes, B. T., & Zhou, A. (2005). TLR at DUC: Tree similarity. In *Proceedings of the DUC 2005*. Vancouver, BC, Canada.
- [122] SEE (Summary Evaluation Environment): <<http://www.isi.edu/~cyl/SEE>>.
- [123] Seki, Y., Eguchi, K., Kando, N., & Aono, M. (2005). Multi-document summarization with subjectivity analysis at DUC 2005. In *Proceedings of the DUC 2005*. Vancouver, BC, Canada.
- [124] Sjöbergh, J., & Araki, K. (2006). Extraction based summarization using a shortest path algorithm. In *Proceedings of the 12th Annual Meeting of the Association for Natural Language Processing* (pp. 1071-1074). Yokohama, Japan.
- [125] Spärck Jones, K. (1997). Summarising: Where are we now? Where should we go? In *Proceedings of the ACL'97/EACL'97 Workshop on Intelligent Scalable Text Summarization*. Madrid, Spain.
- [126] Spärck Jones, K. (1999). Automatic summarizing: Factors and directions. In I. Mani & M. T. Maybury (Eds.), *Advances in automatic text summarization* (pp. 1-14). Cambridge, MA: The MIT Press.
- [127] Spärck Jones, K. (2007). Automatic summarising: The state of the art. *Information Processing and Management*, 43(6), 1449-1481.
- [128] Teufel, S. H., & Moens, M. (1997). Sentence extraction as a classification task. In *Proceedings of the ACL'97/EACL'97 workshop on intelligent scalable text summarization* (pp. 58-65), Madrid, Spain.
- [129] Toutanova, K., Brockett, C., Gamon, M., Jagarlamudi, J., Suzuki, H., & Vanderwende, L. (2007). The PYPHY summarization system: Microsoft Research at DUC 2007. In *Proceedings of the DUC 2007*. Rochester, NY, USA.
- [130] Vanderwende, L., Banko, M., & Menezes, A. (2004). Event-centric summary generation. In *Proceedings of the DUC 2004*. Boston, MA, USA.
- [131] Vanderwende, L., Suzuki, H., & Brockett, C. (2006). Microsoft Research at DUC 2006: Task-focused summarization with sentence simplification and lexical expansion. In *Proceedings of the DUC 2006*. Brooklyn, NY, USA.
- [132] Vapnik, V. (1995). *The nature of statistical learning theory*. New York, NY: Springer.
- [133] Wan, X., Yang, J., & Xiao, J. (2006). Using cross-document random walks for topic-focused multi-document summarization. In *Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence* (pp. 1012-1018). Hong Kong, China.

- [134] Wan, X., Yang, J., & Xiao, J. (2007). Manifold-ranking based topic-focused multi-document summarization. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence* (pp. 2903-2908). Hyderabad, India.
- [135] White, M., Korelsky, T., Cardie, C., Ng, V., Pierce, D., & Wagstaff, K. (2001). Multidocument summarization via information extraction. In *Proceedings of the 1st International Conference on Human Language Technology Research* (pp. 1-7). San Diego, CA, USA.
- [136] Ye, S., Qiu, L., Chua, T.-S., & Kan, M.-Y. (2005). NUS at DUC 2005: Understanding documents via concept links. In *Proceedings of the DUC 2005*. Vancouver, BC, Canada.
- [137] Yeh, J.-Y., Ke, H.-R., Yang, W.-P., & Meng, I.-H. (2005). Text summarization using a trainable summarizer and latent semantic analysis. *Information Processing & Management*, 41(1), 75-95.
- [138] Zha, H. (2002). Generic summarization and key phrase extraction using mutual reinformation principle and sentence clustering. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 113-120). Tampere, Finland.
- [139] Zhang, Z., Blair-Goldensohn, S., & Radev, D. R. (2002). Towards CST-enhanced summarization. In *Proceedings of the 18th National Conference on Artificial Intelligence* (pp. 439-445). Edmonton, AB, Canada.
- [140] Zhang, J., Sun, L., & Zhou, Q. (2005). A cue-based hub-authority approach for multi-document text summarization. In *Proceedings of the IEEE International Conference on Natural Language Processing and Knowledge Engineering* (pp. 642-645). Wuhan, China.
- [141] Zhou, D., Bousquet, O., Lal, T. N., Weston, J., & SchÖlkopf, B. (2003). Learning with local and global consistency. In *Proceedings of the 7th Annual Conference on Neural Information Processing Systems (NIPS 2003)*. Whistler, BC, Canada.
- [142] Zhou, Q., Sun, L., & Nie, J.-Y. (2005). IS_SUM: A multi-document summarizer based on document index graphic and lexical chains. In *Proceedings of the DUC 2005*. Vancouver, BC, Canada.
- [143] Ziegler, C.-N. & Lausen, G. (2004). Spreading activation models for trust propagation. In *Proceedings of 2004 IEEE International Conference on e-Technology, e-Commerce and e-Service* (pp. 83-97). Taipei, Taiwan.