Annotated Confirmation Report

Coherence-based Text Summarization

The Hong Kong Polytechnic University

The Hong Kong Polytechnic University
Department of Computing

Coherence-Based Text Summarization

Report for Confirmation of Registration

2010

Abstracts usually follow this pattern:

a. Background, b. Aims, c. Methods, d. Results, e. Conclusions (and Future Work)

- Describe outline of paper in introduction
- Abstracts are not written in the first person, e.g. ‘This report first describes the significance ...”

Abstract

In this report, I first describe the significance of the project for my PhD program – coherence-based summarization. I observe that informativeness-oriented summarization has been pursued for decades and it is time to shift some attention to expressiveness-oriented summarization. The latter is a largely untilled land, with promising directions in post-extractive ordering, coherence-based extraction, and coherence-based revision with coherence playing an increasingly central role and the output evolving from extractive summarization to abstractive summarization. In Section 2, I survey a massive collection of works, which serve as the background and stepping stones for achieving the goals of this project. The methodology for the proposed project is also described in Section 3, based on my past work and future plan. The preliminary results of some proposed methods are provided in Section 4, accompanied with discussions of their significance and impact. Finally, I describe the plan and scheduled progress for the unfinished parts of the project in Section 5.

Use more formal language. Use “extensive range of studies”, rather than the more informal “massive collection of...”.

Include a “Table of Contents”, including a “List of Abbreviations” and a “List of Tables and Figures”.

1. Project Description

The project for my PhD program is coherence-based text summarization, which is distinguished from general summarization research in its emphasis on coherence.

The concern with coherence is motivated by the purpose of automatic text summarization—to provide human readers with a sufficiently abridged summary of a long document or document set to facilitate efficient information processing. In this sense, the summary serves as a surrogate for the original document(s) in terms of informativeness and expressiveness. Informatively, the summary is expected to maximally reproduce the original document's essential information in a reduced space. Expressively, it is expected to convey the information in an intelligible and coherent way to human users. In the past, much emphasis has been laid on informativeness, leading to increasingly sophisticated models, algorithms, and evaluation methods. But the other side of the coin — a summary's quality of being easily understood — counts no less.

✓States why the project is important

✓Explains key concepts

A deciding factor for the expressiveness of a summary is coherence, i.e., how well textual components such as sentences are connected to each other and together in the whole text. Failure to address coherence will defeat the purpose of summarization because coherence is interrelated with informativeness. An incoherent summary, e.g., with unresolved anaphors or a disordered structure, will thwart the communication of the content to a human reader, no matter how informatively faithful it is to the original document.

✓Gives a clear example to explain key point

According to text linguistics, coherence takes effect on two levels: global and local, which involve different discourse and mental processes (Tapiero, 2007). Global coherence takes the whole text in its view and measures to what extent the pieces of a summary stand together. Local coherence is often referred to the relationship between "local" or adjacent discourse units (such as sentences) and measures their closeness by cohesive patterns (e.g., repetition) or entity continuity (Grosz et al.), connectedness. It is my goal to build a framework to capture both global and local coherence.

✓Cites sources and gives a clear definition of coherence

Include date when citing sources, i.e. Grosz et al (1995).

The above description, however, does not imply that coherence-based summarization is very different from the existent approaches and models. My primary interest is to integrate coherence-based techniques into the mainstream summarization frameworks with the mature techniques, models, algorithms, and concepts developed over decades. The integration is three-fold: 1) post-extractive ordering, 2) coherence-based extraction, and 3) coherence-based revision.

✓Explains the scope of the project

Post-extractive ordering represents the shallowest integration of coherence elements with any extraction systems. The first stage, sentence extraction, is not affected by coherence. At the second stage, the extracted sentences are ordered according to coherence-based principles and algorithms to optimize output readability.

Coherence-based extraction makes a further stride by extracting sentences with coherence-based models. The selected sentence are not only representative of the most salient information in the document(s) but are also connected in terms of event structure or content components.

✓Uses paragraphs effectively to separate each main concept

Both post-extractive and coherence-based extractions are still extractive methods of summarization. Beyond that, abstractive summarization is still a major challenge. Coherence can make its contribution at that level by revising the extracted textual units (e.g., sentences) to maximize connectedness in the output.

Most current summarization models are built for the news domain. Since the summarization techniques are needed for sundry text genres and the significance of coherence is more manifest in narrative and expository text, I will apply the coherence-based techniques to various text genres to show their real potential.

💡Be less subjective and do not use "I", i.e. "Coherence-based techniques will be applied to…"

In sum, the goal of this project is to enhance the expressiveness of automatic summaries by instilling coherence into the mainstream summarization frameworks on different levels, from the shallowest post-extractive ordering to the deepest coherence-based revision. It is my expectation that the output quality of such coherence-adapted or coherence-based systems will be significantly improved for various text genres, both news and beyond.

The rest of the report is organized as follows. Section 2 is a comprehensive review of related literature, with emphases on the history and development of text summarization and representative approaches, both non-coherence-based and coherence-based. Section 3 describes my methods, implemented or proposed, to tackle the three sub-goals of the project: post-extractive ordering, coherence-based extraction, and coherence-based revision. Some preliminary results from post-extractive ordering and coherence-based extraction are presented in Section 4, which are accompanied with discussions about their significance and impact. My plan and scheduled progress for the unfinished parts of the project are presented in Section 6.

✓Outlines the content of the rest of the report

✓Uses passive voice to give more objectivity to the report

2. Literature Review

💡For Literature Review use:

a. Past simple tense for reported studies and showing what is no longer true

b. Present simple tense to indicate study results that are still valid

Since its inception in the late 1950s, automatic text summarization has been actively pursued for more than half a century and proved to be one of the most vigorously explored frontiers in NLP applications. What was once believed to be a human-privileged creative task is now extensively automated in modern computing labs and commercial packages. The practical achievement is paralleled by theoretical advancement: the past few decades has witnessed the mushrooming of theories, models, algorithms, implemented systems, as well as our enhanced understanding of text summarization per se, ranging from taxonomies to techniques and across species of summarization. Work in this area has so flourished that the turn of the century saw two compendiums of the state-of-the-art models and techniques: (Mani and Maybury, 1999) and (Mani, 2001).

✓Gives general background to topic
✓Uses present perfect tense for background, e.g. "has witnessed"

Use brackets only for the year when citing directly i.e. ...techniques: Mani and Maybury (1999) and Mani (2001)

With reference to those and many more advanced approaches in the years that follow the compendiums, this section is intended to draw a large picture of automatic text summarization. Specifically, 2.1 will give a bird's-eye-view of the field, summarizing the taxonomies, motivations, stages, factors, and other conceptual elements as preliminaries for this project; 2.2 and 2.3 will explore more technical details, by zooming in on both non-coherence-based approaches and coherence-based approaches.

Use focal expressions, I.e. "focus on" rather than "zooming in".

2.1 An Overview of Text Summarization

In this section, I will review the founding works of the major terminological concepts about text summarization. The discussion will center on different accounts — historical vs. concurrent, analytical vs. holistic, theoretical vs. practical — in order to capture the multi-dimensional nature of text summarization.

When referring to a study, work is an uncountable noun. Alternatives include: "studies" or "research"

2.1.1 A Historical Account

Automatic summarization is historically preceded by human summarization serving various communicative needs. In fact, many defining concepts of the former are inherited from the latter, including the well-known dichotomy of **extract** and **abstract** (see 2.1.2).

✓Gives a clear topic paragraph
✓Refers to what will appear later in the paper

Systematic study of human professional summarization has shed much light on automatic summarization. Cremmins (1996) and Endres-Niggemeyer (1998) found that human summarizers tend to focus on shallow linguistic features such as cue phrases, location, key content (title, headings, etc.), which are the cradles of many shallow feature-based models in automatic summarization, such as (Edmundson, 1969) and (Kupiec et al., 1995). (Teufel and Moens, 1999) can be regarded as a direct implementation of Cremmins's (1996) findings about selecting the purpose, methods, results etc. as key textual component. Endres-Niggemeyer (1998: 146–157) divides central summarization into three subtasks: 1) document exploration; 2) relevance assessment; 3) summary production, leading to the popular tripartite stages of automatic summarization (see 2.1.3). Human-induced summarizing/abstracting patterns and strategies will continue to inform and enlighten the computational community.

Avoid using bolded letters for key words.

Do not use brackets when the research paper is the subject. The citation should be: "Teufel and Moens (1999) can be..."

Only include page numbers for direct quotations.

Luhn (1958) is commonly credited as the trailblazer in automatic summarization. He built the first automatic summarizing system, which was followed by a few similar models (Edmundson, 1963; Rush et al., 1971; Skorohodko, 1971). (Edmundson, 1969) is a milestone in the early years of text summarization, establishing the basic extraction method.

✓Groups previous research together based on similarities

The next trophy is usually attributed to (Kupiec et al., 1995) as a medium between the shallow-feature extraction (Edmundson, 1969) and the modern statistical and corpus-based approaches. In the "hiatus" (Hovy, 2005: 583) of over 20 years in between, there were a number of cognitively grounded summarizing systems or models, such as FRUMP (Dejong, 1982) and SCISOR (Jacobs and Rau, 1990). They all take semantic representation as input and incorporate complicated knowledge processing, which makes them markedly different from today's summarizing systems taking text as input and utilizing models and algorithms from AI and NLP. A more detailed introduction of such efforts can be found in (Endres-Niggemeyer, 1998: 310–330).

Do not use brackets when the research paper is discussed directly. The citation should read: "is usually attributed to Kupiec et al. (1995) as...".

Ushered in by (Kupiec et al., 1995), the age of text summarization has arrived, with upgraded technology (machine learning, statistical, corpus-based, etc.), sharpened tools (lexical cohesion, discourse structure, graph model, etc.), and extended coverage (from single-document summarization to multi-document and query-focused summarization). More details will be given in the following sections.

Do not use "etc" with lists. It is more appropriate to state "summarization has arrived with machine learning, statistical developments and other forms of upgraded technology".

2.1.2 A Taxonomical Account

The task of text summarization can be classified along various dimensions. One is the informational coverage, according to which there are **extract**, a summary of sentences or phrases verbatim from the source document(s), and **abstract**, a summary containing original sentences via content reformulation or paraphrase, which is sometimes called **non-extractive summary** (Spärck Jones, 2007: 1473). Non-extractive summarization is a typical human reserve and its computerization usually involves deep semantic / logical analysis on the discourse level and the language generation technique. FRUMP (Dejong, 1982) is an early documented non-extractive system, followed by STREAK, PLANDOC (McKeown et al., 1995), SUMMON (McKeown and Radev, 1995), SUMMARIST (Hovy and Lin, 1999), SumUM (Saggion and Lapalme, 2002), the template-filling approach (Paice and Jones, 1993), the cut-and-paste approach (Jing and McKeown, 1999), etc. However, the vast majority of implemented systems and models are targeted at extractive summarization, which has enjoyed a long history and is still the main force.

✓Classifies key concepts by highlighting differences

Only use family name when citing, i.e. "(Jones, 2007)".

Taking a functional dimension, we can divide summaries into **indicative**, **informative**, and **critical** types. Indicative summaries indicate the content of a document with no further detail; informative summaries provide such detail and can act as an abridged surrogate for the source document(s); critical summaries represent the summarizer’s attitude and opinion about the source document(s) and are thus highly human-privileged. With a few exceptions (e.g., Saggion and Lapalme, 2002), automatic summarizations are exclusively informative.

Avoid starting sentences with "Taking..." to introduce an idea. Use a subject, i.e. "The functional dimension can be divided into..."

Taking a utilitarian dimension, we identify **generic** summaries and **query-focused** summaries. Many summarizing models and systems are traditionally oriented to generic summaries, which don’t address a particular user need. On the other hand, query-focused summaries, which are produced in response to a user need or query and related closely to question-answering systems and information extraction techniques (Jurafsky and Martin, 2009: 836-838), have attracted sustained research interest in the past decade.

✓Uses evaluative language to describe strengths and weaknesses of previous research, e.g. "don’t address"’ "pioneering"

A pioneering work, (Baldwin and Morton, 1998), addresses an obvious difficulty caused by query — coreference identification of the key terms in the query. A query (or headline) term and its related terms form a coreference chain, which is used to select sentences for summary. Mani and Bloedorn (1999) report a more complicated query-based MDS system that is built on a standard “analysis-refinement-synthesis” architecture. In the analysis stage, documents are represented as graphs with words as nodes and word attributes and relations as edges. In the refinement stage, a spreading activation algorithm is used to reweight the nodes based on the user's query. Then commonalities and differences between documents are represented as a matrix for sentence extraction.

Although most query-focused models are slightly adapted from query-free models, there are also original models designed for query-focused summarization, such as Daumé III and Marcu's (2006) Bayesian model. It is built on the idea of using known relevant documents, also known as query expansion, in information retrieval.

Ever since the Document Understanding Conference (DUC) 2003 summarization track (Over and Yen, 2003), query-focused summarization has been a routine task of this annual competitive event and its successor, the Text Analysis Conference (TAC), since 2008. In those events, the queries have also evolved from keywords/phrases to narratives and predefined aspect collections.

The last dimension is the number of source documents, which can distinguish **single-document summarization** (**SDS**) from **multi-document summarization** (**MDS**). Literally, single-document summarization is operated on a single source document and is the default task undertaken by most early extractive or abstractive models (Luhn, 1958; Edmundson, 1969; Kupiec et al., 1995; Hovy and Lin, 1999, inter alia). Motivated by the information overload with the explosive growth of textual information on the Web, multi-document summarization is operated on a collection of related documents and expected to produce a collection-wide synopsis.

Avoid starting a sentence with an adverb such as "literally", place it next to the verb, i.e. "is literally operated…"

✓Uses bracketing for references when they are not directly cited, but appear as a list at the end of the sentence.

SUMMON (McKeown and Radev, 1995) is an early implemented MDS system. It is very different from most of the modern MDS models in that it is built on a language generation model with templates, instead of raw text, as input. Unlike SUMMON, most modern MDS models and algorithms address the issue of redundancy reduction by identifying document similarities and differences. One solution is the Maximal Marginal Relevance (MMR) scoring system established by Carbonell and Goldstein (1998). Another solution is the centroid method advocated by Radev et al. (2000, 2004a). (Wan and Yang, 2008) is a more complicated model of the clustering-based algorithms.

✓Explains main features of systems, but could extend the point by discussing the strengths and weaknesses.

Provide the page number with direct quotations.

✓Compares differences between approaches and highlights major strengths

There are also models working beyond redundancy, such as (Okazaki et al., 2003), and under the principle of "sentences which are relevant to ones of significance are also significant". Spreading activation is used to rank sentences. Barzilay et al.'s (1999) work focuses on reducing redundancy in MDS by identifying the similarities and differences between related single-document extracts. Their method consists of content selection based on paraphrasing rules and sentence generation realized by the sentence generator SURGE. A different path is sought by Steinberger and Kristen (2007) with their Latent Semantic Analysis (LSA)-based model, a development of its SDS version (Gong and Liu, 2002). The advantage of the LSA approach is that no domain knowledge or corpus resource is needed in order to work out the topics represented in word-sentence matrices.

2.1.3 A Compositional Account

There are different models for the general architecture of text summarization. A classic one is attributed to Mani (2001: 14), who identifies **analysis**, **transformation**, and **synthesis** as the three fundamental phases in a "high-level architecture of a summarizer". Both analysis and synthesis address some "internal representation" of a text through deep semantic and logical parsing. Transformation, however, concerns the condensation of information from the source and is thus regarded as the essential phase of summarization.

✓Gives page number for direct quotation

✓Introduces section by explaining key terms

A terminological variant of this tripartite model is adopted by Spärck Jones (1999), consisting of **interpretation** (from source text to source representation), **transformation** (from source representation to summary representation), and **generation** (from summary representation to summary text).

A more microscopic and extract-oriented model is assumed by Hovy (2005), who establishes **topic identification**, **interpretation**, and **summary generation** as three distinct stages of summarization. Topic identification corresponds to the selection of the most salient, or extract-worthy, units (e.g., sentences) by various criteria such as position, cue phrase, word frequency, etc. While topic identification is often equated with simple (extractive) summarization, interpretation and summary generation are aimed at higher-quality, abstractive, human-like output. Interpretation means the transformation of words to concepts by simulating the human understanding. Due to the underdeveloped knowledge engineering, it is a largely untilled field despite a few trials (Dejong 1978; Jacobs and Rau, 1990; Hahn and Reimer, 1999). Summary generation is a indispensable part of summarization, which aims at reducing disfluencies and improving readability of abstracts or extracts.

✓Introduces section by explaining key terms

✓Explains model using examples.

✓Explains weaknesses with the model

In early works, much effort in summary generation is made to remove syntactically subordinate constituents, such as attributions, appositives, and adverbials (Grefenstette, 1998; Jing, 2000). Simple techniques such as clause deletion are used in (Baldwin and Morton, 1998). Jing and McKeown (2000) explore sentence combination in addition to sentence reduction by studying the "cut and paste" operations in human summarization.

More recent approaches aimed at sentence compression have relied on supervised machine learning by using parallel document / summary corpora.

Exemplary methods are maximum entropy (Riezler et al., 2003), the noisy channel model (Knight and Marcu, 2000), large-margin learning (McDonald, 2006), and Integer Linear Programming (Clarke and Lapata, 2007).

Apart from sentence compression or reduction, summary generation also includes more advanced operations such as revision (Mani et al., 1999), fusion (Barzilay et al., 1999; Barzilay and McKeown, 2005), and rewriting / paraphrasing (Barzilay and Lee, 2003; Nenkova, 2008).

Jurafsky and Martin (2009: 824) identify three stages for summarization: **content selection**, **information ordering**, and **sentence realization**. Content selection and sentence realization are basically Hovy's (2005) topic identification and summary generation, with a narrower focus on sentence extraction. The intermediary information ordering concerns the ordering of the selected sentences in the output. Whereas text order is the default ordering for single-document summarization, recently sentence ordering is a hot topic for multi-document summarization (Barzilay et al., 2002; Lapata, 2003, 2006; Barzilay and Lee, 2004; Barzilay and Lapata, 2005, 2008; Karamanis et al., 2004a, 2004b, 2008; Karamanis and Mellish, 2005; Karamanis, 2006, 2007; Ji and Pullman, 2006; Soricut and Marcu, 2006; Nahnsen, 2009).

✓Groups together similar models/methods
✓Highlights similarities
✓Explains differences

Avoid clichés such as "hot topic", use more formal language, e.g. "much researched area".

In addition to these areas, the research on text summarization is not complete without evaluation of the output for content/informativeness and coherence/readability or their combination. Summarization evaluation methods are generally classified as **intrinsic** evaluation and **extrinsic** evaluation. Intrinsic methods evaluate a system for the quality of its output by doing cross-summary comparisons, so that one system-produced summary is evaluated against other system-produced summaries, simple baselines, or human-produced summaries. Since human-produced summaries lack agreement (Rath et al., 1961), automatic summaries can be compared against chosen sets (e.g., intersection, union) of multiple human-produced summaries (Salton et al., 1997). The lead baseline, which is very hard to beat in news summarization, is widely used (Brandow et al., 1995). Initiated by (Lin and Hovy, 2003), the content-oriented intrinsic evaluation has been fully automated and implemented as ROUGE (Lin, 2004) and BE (Hovy et al., 2005) in the DUC/TAC tasks. Coherence or overall quality-oriented intrinsic evaluation can be done by human judges according to the Pyramid Method (Passonneau et al., 2005).

✓Provides good clear topic sentence linking to previous paragraph using "these"
✓Introduces forms of evaluation
✓Explains method
✓Extends explanation using "so that"

Extrinsic methods evaluate a system by means of external tasks and makes cross-species (e.g., summary vs. source document) comparisons. A classic example is the reading

Avoid etc. An alternative is "followed by a number of other studies, including Firmin and Chranowski (1999) and Mani et al. (2002)".

comprehension test reported in (Morris et al., 1992), followed by (Firmin and Chrzanowski, 1999), (Mani et al., 2002), etc.

Looking beyond and taking the summarization context into consideration, Spärck Jones (2007) explicates three classes of context factors for summarization: **input factors**, **purpose factors**, and **output factors**. Input factors are used to characterize source material, including form (language, register, medium, structure, genre, length), subject, unit, author, and metadata; purpose factors relate to the purpose or intended use of summarization, including use, audience, and envelope (time, location, formality, trigger, destination); output factors regard the presentation and formatting of the result, including material (coverage, reduction, derivation, specialty), style, and format (language, register, medium, structure, genre). Those factors are often the hotbed for new problems and breakthroughs. For example, Moens and Dumortier (2000) distinguish opinions from reportage, a commonly used genre — an input factor — for many news-targeted systems; Farzinder and Lapalme's (2004) system is oriented for legal audiences — a purpose factor; White and Cardie (2002) produce rich hypertext as the output structure — an output factor.

💡 *Avoid spoken phrases, e.g. 'hotbed' use formal words, i.e. "origin of".*

2.1.4 An Implementational Account

In this section, I will document a list of implemented summarizing systems since 1980. All of them are denoted by acronyms and most are not commercialized (cf. Microsoft Word's AutoSummarize).

The first batch of systems, spanning the time period 1980–1990, typically incorporates text understanding and knowledge engineering techniques, which is motivated by theories about human cognition in summarization (Endres-Niggemeyer, 1998: 310–312). Internal representations as the product of deep semantic and/or pragmatic analysis are invariably used for further processing. Another noticeable feature shared by those systems is that they don't distinguish "abstract" from "extract", as human-like output (i.e., abstract) is the only goal they pursue.

✓Outlines contents of section
✓Explains method
✓Refers to key previous study for authentication

FRUMP (Dejong, 1982) is a pioneering system in this camp, using event schemata and sketchy scripts to organize its domain knowledge and adopting an expectation-driven method to activate scripts for news summaries.

✓Details individual studies

A parsing-intensive but partially implemented system is SUSY (Fum, et al. 1982). It is intended to summarize scientific texts with a comprehensive text processing model. The model generates a propositional representation from the text input, extends it with logical and rhetorical structures before submitting the extended representation to a hierarchical propositional network for importance ranking. Sophisticated syntactic, semantic, and rhetorical parsers are needed, which explains why a considerable part of its design remains on paper.

SCISOR (Jacobs and Rau, 1990) is the closest early system to the modern multi-document summarizers in that it can produce summaries from multiple input texts. Owing to its memory of a large-scale knowledge base, it can output conceptual structures to assist conceptual retrieval. Assisted by syntactic and semantic analysis of the input text, it is mainly operated on the conceptual, instead of textual, level.

TOPIC (Hahn, 1990) is an early indicative-summarizing system. The input text interacts with knowledge base concepts and ontology to identify important concepts in the relevant text keywords. Such concepts are then semantically grouped and encoded into text segments for output, which can be presented textually or graphically for retrieval.

✓Summarises system
✓Highlight its effectiveness
✓Explains limitations

A very different member in this group is PAULINE (Hovy, 1988a), possibly the only pragmatically driven summarizer known to the community. It accepts semantic representations and adapts the summaries to specific communicative intentions and goals of the user, making automatic summaries sensitive to human needs.

Do not simply list studies. Highlight why they were important, i.e. "first to", "further developed" "introduced" "improved".

The next decade (1990–2000) saw the birth of a new generation of summarizers. Though some of them inherit the knowledge processing legacy, more influential systems such as SUMMARIST and SUMMON demonstrate people's inclination to treating text summarization as a knowledge-independent NLP task, with clear distinctions of extraction / abstraction and single-document summarization / multi-document summarization built into the systems.

Like the above-mentioned TOPIC, SIMPR (Gibbs, 1993) is another indicative-summarizing system. But unlike TOPIC, it produces indexes for quick retrieval. The indexing process incorporates both morpho-syntactic constraints and knowledge-based generation rules. Some procedures, such as text compression, normalization, and stopword filtering, resemble their counterpart in the modern systems.

✓Compares using effective language, e.g. "unlike", "another", "both", "resemble"

Two systems generating abstracts of domain-specific documents—STREAK and PLANDOC—are reported in (McKeown et al., 1995). STREAK summarizes basketball game results and PLANDOC summarizes telephone network planning activity. Both make use of language generation techniques. Each of them applies to a specific domain and accepts only structured data as input.

A more recent variant that summarizes structured data is SumGen (Maybury, 1999). The system consists of three main procedures — content selection, aggregation, and presentation — which are all targeted at event data.

✓Highlights motivation for system

✓Evaluates system, e.g. "disappointingly".

Since structured data or semantic representation involve expensive parsing, deep semantic analysis, domain-specific knowledge engineering, and generation techniques, efforts are made to develop lightweight text-to-text systems. One such example is ANES (Brandow et al., 1995), a news summarizing system that selects sentences according to the tf.idf statistics at different lengths. Disappointingly, the lead baseline overwhelms the ANES output by a wide margin.

SUMMON is a well-known multi-document summarizing system (McKeown and Radev, 1995). Since it is built on language generation models, templates instead of raw text are used as input. Its major constituents are a content planner and a linguistic component. As is characteristic of multi-document summarization, the similarities and differences between news sources are found by comparing their templates. Then, SUMMON's "summary operators" (e.g., change of perspective, contradiction, addition, superset, etc.) are used to merge the related content in different documents.

A champion system in this period is SUMMARIST (Hovy and Lin, 1999), a modulated, comprehensive system to deal with both extraction and abstraction. It was a state-of-the-art prototype because it adopts the new wavefront and topic signature techniques and is constructed on the principle of combining symbolic knowledge and statistical/IR techniques. The four major modules of SUMMARIST are preprocessing, topic identification, topic interpretation/concept fusion, and summary generation.

✓Evaluates system, e.g. "state-of-the-art"

SUMMARIST would have been a truly comprehensive system with, as the authors suggested, a "Discourse Structure module" added to its topic identification (Hovy and Lin, 1999: 91). But that call was answered by the rhetorical structure tree (RST) model (Marcu, 1999, 2000). The RST model adopts a tree structure to represent a text, with textual units (sentences/clauses) and their rhetorical relations as the nodes. For sentence extraction, the salience of sentences is determined by the depth of their corresponding node, which in turn is determined by their nucleus/satellite status constrained by the RST rhetorical relations. RST is still one of the most influential models for global coherence modeling.

Avoid spoken expressions, e.g. "But that call was answered". Use more formal structures, i.e. "that was achieved by...."

✓Gives a good summary sentence to finish paragraph

✓Evaluates model, e.g. "most influential"

In the first decade of this century, increased interest in automatic summarization and public competitive events (DUC and TAC) stimulated the growth of new systems. With each participant since DUC 2001 counted as a distinct system, the total will be in the hundreds. Nonetheless, for the limit of space I will only discuss some representative, well-known, or publicly available systems. See (Spärck Jones, 2007: 1474–1476) for further details.

The first such system is SumUM (Saggion and Lapalme, 2002), which is targeted at technical documents. It addresses the need of abstracts and integrates indicative and informative summarization in “selective analysis”. It also distinguishes itself from the other systems in three aspects: 1) inclusion of a text generation module to simulate human abstractions, as suggested by (Endres-Niggemeyer, 1998); 2) summarization of technical documents instead of newswire articles; 3) support by a corpus study of manual alignments between human abstracts and source documents.

✓Gives good explanation of how system works and shows how it differs from previous systems, using a parallel list for each item starting with a noun

Avoid only listing previous systems studies in the review. Try and link them to current project or to how knowledge of a subject developed.

Lin and Hovy (2002) built a multi-document summarizing version of their single-document SUMMARIST — NeATS — which distinguishes itself at DUC 2001. The extraction component makes use of sentence position, term frequency, topic signature, and term clustering. Redundancy removing (using Maximal Marginal Relevance) and coherence enhancing (by adding lead sentence) techniques are also used.

Another sophisticated system that made its debut at DUC is GISTexter, which is capable of producing both single-document and multi-document summaries (Harabagiu and Lacatusu, 2002). A noticeable feature is that it uses information extraction-style templates, which collect sentence information for a given set and are classified for summary generation. Expensive parsing, coreference resolution, and template filling are needed. GISTexter was replaced by Lite-GISTexter at DUC 2003 (Lacatusu et al., 2003) and incrementally modified in accordance with the changing DUC tasks.

✓Gives strong topic sentence linking to previous paragraph, e.g. “Another sophisticated system”

A representative discourse-level summarizer during the period is PALSUMM (Polanyi et al., 2004). Like RST, it extracts sentences based on discourse structure; unlike RST, it relies on more thorough syntactic and semantic interpretation of discourse units according to the Linguistic Discourse Model (LDM). The implemented system can produce quality summaries that preserve the language and style of the source document.

A public domain and open source platform for multi-document summarization is MEAD (http://www.summarization.com/mead), developed by Radev et al. (2004b). It has implemented a number of summarization algorithms (centroid-based, query-based, position-based, keyword, etc.) and popular classifiers are provided for training purposes. Another publicly available and mass audience-oriented system is Columbia University’s Newsblaster (http://newsblaster.cs.columbia.edu/), developed by McKeown et al. (2002).

✓Summarises main idea of the section with a clear subtitle

2.2 Non-Coherence-Based Approaches

After a bird's-eye-view of summarization, I will now turn to more technical details in this section and the next. A rough split is to divide everything into coherence-based and non-coherence-based. Although the former is the incubator for many of the ideas and models in this project, the latter lays the foundation and provides general reference. For this reason, I review non-coherence-based approaches first.

Avoid idiomatic expressions, e.g. "bird's-eye-view", "incubator for" "rough".

In the following, I will discuss the core techniques of this large camp, including the classic shallow feature-based approaches (2.2.1), the lexical relation-based approaches (2.2.2), the corpus-based and machine learning-integrated approaches (2.2.3), the graph-based approaches (2.2.4), the discourse structure-based approaches (2.2.5), and the knowledge processing-based approaches (2.2.6). I will focus mainly on extractive summarization because of its dominance.

Avoid unclear topic sentences, e.g. "In the following". Use "The following sections outline the core techniques…Section 2.2.1 reviews…"

2.2.1 Shallow Feature-Based Approaches

The most basic sentence extraction makes use of shallow linguistic features, such as word frequency, length, position, text layout, keywords, etc. The earliest such approach is reported by Luhn (1958), who measures sentence extract-worthiness by word frequency only, assuming that the sentences selected for extract must contain the most frequent words in a text.

Edmundson (1969) extends Luhn's work by considering cue phrases, title, and location in addition to keywords based on word frequency. The findings that the combination of cue-title-location gives the best performance and that location is the best individual feature are often quoted as the most substantial achievements made by shallow feature studies.

✓Explains how Edmundson's work relates to Luhn's in order to show how theory developed

✓Highlights most important features of Edmundson's findings

Pollock and Zamora (1975) apply the shallow feature-based approach to chemical abstracts, but pure shallow feature-based studies are rarely reported after (Edmundson 1969). However, an important amendment is made by (Lin and Hovy, 1997), where the finding about location ("Position Hypothesis") is rigorously tested. The authors use the Ziff-Davis corpus composed of document with keywords and abstracts, and evaluate the position-based extract under the "Optimal Position Policy". The resultant ranked position list is Title, Paragraph 2 Sentence 1,

Do not use brackets when the writers referred to are part of the sentence's grammatical structure, i.e. "after Edmundson (1969)."

Paragraph 3 Sentence 1, Paragraph 2 Sentence 2, Paragraph 4 Sentence 1, Paragraph 5 Sentence 1, etc.

2.2.2 Lexical Relation-Based Approaches

Edmundson (1969) also finds that keyword alone is the worst individual feature. A possible explanation is that word frequency is not as valuable as word relations, such as synonymy, hyponymy, and meronymy, in sentence extraction. The availability of machine-readable dictionaries and thesauri like WordNet (Fellbaum, 1998) makes it a promising field.

Try and distinguish between using past simple tense to report studies and show what is no longer true and the present simple tense to indicate your own view on the study and what is still true, i.e."Edmundson (1969) found" and "makes it a promising field".

Based on previous studies on lexical relations (Morris and Hirst, 1991), Barzilay and Elhadad (1997) explore producing summaries by using lexical chains, a useful tool to measure the connectedness between sentences with reference to lexical relations. The authors employ a non-greedy disambiguation heuristic to select chain member senses and extract sentence according to chain scores and word frequencies in a chain. A recognized limitation of their method is the lack of control of the compression rate.

The efficiency of the lexical chain-based method is later improved by Silber and McCoy (2000), who use meta-chains, a special data structure, to achieve a linear core runtime.

The idea of lexical chains also finds its way in an early paper on query-focused summarization by Manabu and Hajime (2000). Their lexical chains are constructed by word co-occurrence information via cosine distance-based clustering. Then the passage score is determined as the sum of the scores of lexical chains, weighted by the degree of lexical chain overlap.

Like shallow feature-based approaches, pure lexical relation-based approaches are not actively pursued. Anyhow most word-level concepts and tasks (lexical cohesion, word sense disambiguation, stemming and lemmatization, POS tagging, named entity recognition, etc.) are integral parts of many advanced approaches and models.

✓Places findings of previous studies in context of current models and approaches

2.2.3 Corpus-Based Approaches

This is a large camp of non-coherence-based approaches, characterized by the use of annotated or un-annotated corpus, statistical measures, and machine learning algorithms.

The foundation is laid by (Kupiec et al., 1995), commonly known as KPC. The authors use a set of five features (sentence length, fixed-phrase/cue phrase, paragraph (location), thematic word (frequency), uppercase word) to train a Bayesian classifier. Like Edmundson (1969), they find paragraph location the most useful feature and the combination of paragraph location + fixed-phrases/cue phrases + sentence length optimal. They also report the training method’s significant recall improvement on its non-training counterpart (84% vs. 44%), which leads to sustained enthusiasm about corpus-based approaches in the following years.

Do not use both what is finished “in the following years” and what is not finished ”leads to” together, i.e “led to…in the following years”.

A direct inheritor of KPC is (Myaeng and Jang, 1999), which applies a similar corpus-based approach to summarizing Korean texts, with two noteworthy modifications: 1) using a text component identification model to filter sentences before ranking and selecting them; 2) limiting the KPC approach to individual features and then computing the final score for each sentence with the Dempster-Shafter combination rule.

The most obvious benefit of using a corpus in text summarization is that weighted term frequency, or tf.idf (Spärck Jones, 1972), can replace the simple word frequency as a better feature. It is adopted by Aone et al. (1999) in extensive experiments on different feature combinations and calculation methods. Like KPC, they also find training methods significantly better than non-training methods.

✓Places studies in context of research, e.g. “a direct inheritor”, “an alternative to” and “a more general method”

An alternative to tf.idf for content selection is the log-likelihood ratio (Dunning 1993), which is used to estimate “topic signature” in (Lin and Hovy, 2000). It is reported that the topic signature method, a core component of SUMMARIST, outperforms the tf.idf method in topic identification.

A more general method than tf.idf or log-likelihood is the centroid algorithm developed by Radev et al. (2004a). Cluster centroids are used to generate multi-document summaries. Following the line derived from word frequency, Nenkova and Vanderwende (2005) focus on word frequency and content frequency in their SumBasic system.

2.2.4 Graph-Based Approaches

Recently, sustained efforts are invested in graph-based approaches, which apply the graph model to text by mapping textual units (e.g., sentences) to nodes and their relations to links. This line of work is mostly encouraged by the success of the PageRank (Brin and Page, 1998) and HITS (Kleinberg, 1999) algorithms in modeling hypertext structure.

Place adverbs next to main verb and use “recently” with present perfect tense, e.g. “have recently invested”.

The most representative approach is TextRank (Mihalcea, 2004, 2006; Mihalcea and Tarau, 2004), which establishes sentence connection as a similarity relation computed from same-token overlaps normalized by sentence length. The authors show that HITS and PageRank, when used in backward directed graphs, lead to optimal sentence ranking.

A more complicated variant is the LexRank algorithm proposed by (Erkan and Radev, 2004). It ranks sentences for extraction by using eigenvector centrality operated on a connectivity matrix of the graph representation of sentences, instead of computing the conventional centroids.

Wei et al. (2009) observe that sentence ranking is not only determined by sentence importance relative to each document, but also influenced by document similarities. In their work, a graph model based on LexRank is established that takes into account both generic summarization and query-oriented summarization. A PageRank-like algorithm is adopted, which emphasizes the document impact on the sentence affinity matrix and the preference vector. The result compares favorably with the LexRank models.

✓Outlines main changes employed by each study

✓Comments on changes, e.g. "more complicated", "compare favorably"

A different graph-based approach is recently proposed by (Ganesan et al., 2010), which is targeted at highly redundant opinion sentences. Words are represented as nodes and annotated with sentence ID and sentence position. Relying on "valid paths", their algorithm is able to find redundant paths, collapsible nodes, and collapsible sentences to generate stitched sentences. Summarization is based on ranking of valid paths and elimination of similar paths.

💡Only the year of publication should be in brackets and not the authors' names when the study is the subject of the sentence, e.g. Ganesan et.al (2010) is more accurate.

2.2.5 Discourse Structure-Based Approaches

Most of the approaches introduced above regard the source text as a collection of sentences and operationalize their core algorithms on the sentence or word level. A different family of approaches, however, take the whole discourse in their view and extract discourse units on this level. There are two strains of this family: one that studies the coherence relations between discourse units, which will be surveyed in 2.3, and the other that is simply based on the structural characteristics of a discourse, which will be introduced now.

✓Uses transitional sentence to link Section 2.2.5 with previous section, e.g. "introduced above"

✓Explains content of the current section

✓Uses passive voice to outline contents, e.g. "will be introduced".

💡Do not use time phrase "now" when describing structure. Use "next" or "in this section".

Parallel to the classic shallow feature-based approaches, Teufel and Moens (1999) explore discourse-level summarization by studying the "argumentative structure" of science research papers. They identify 7 rhetorical roles (Background, Topic/Aboutness, Related Work, Purpose/Problem, Solution/Method, Result, Conclusion/Claim) as global rhetorical features and exploit KPC-style heuristics to classify and extract sentences.

The possibility of using paragraphs, instead of sentences, as extraction units is tested by Salton et al. (1997), who utilize text structuring and segmentation. A paragraph relationship graph is established for a text, based on which topic-bearing paragraphs can be identified and extracted with "bushy" or "depth-first" algorithms.

✓Uses range of verbs to describe previous research, e.g. "tested", "utilise", "established", "reported" "identified", "work on", "rely on" "implemented"

A more comprehensive application-grade endeavor is reported by Strzalkowski et al. (1999). The authors exploit the Discourse Macro Structure (DMS), such as the Background-Main Story structure in most news-style documents. Like Salton et al. (1997), they work on the paragraph level. They also rely on shallow (including DMS) features to score paragraphs, such as titles, cue words, location, etc. in the Edmundsonian paradigm.

The PALSUMM (Polanyi et al., 2004) introduced in 2.1.4 is an implemented system that works on the syntactic and semantic structure of the discourse.

The last batch of this group consists of models tailored for specific types of discourse, such as technical (Teufel and Moens, 1999), legal (Grover et al., 2003; Farzinder and Lapalme, 2004), and medical (McKeown et al., 1998; Elhadad and McKeown 2001) types.

✓Gives list without using etc., e.g. "such as ..."

2.2.6 Knowledge Processing-Based Approaches

Aimed at concept extraction instead of content extraction, some summarization models even go beyond the linguistic level and work on domain knowledge and logical representations to produce non-extractive summaries. Since this used to be an active research filed, several major works are cited below.

✓Gives motivation for citing studies

Lehnert (1999) is the first such model, which is often credited with tapping narrative summarization — a highly challenging summarization task — from a knowledge-based perspective. The summarization generation is based on the identification of plot units and causal links. The narrative structure is in turn represented as a graph that reflects the relations between plot units.

Hahn and Reimer (1999) treat text summarization as a transformation process on knowledge representation structures. They use a terminological logic to identify salient concepts, salient relationships, salient properties, and related salient concepts formalized as various operators. Then a topic description is decided

✓Represents time relationships, e.g. "in turn", "then", "final" from the paragraph-level salient information. The final result is a hierarchical text graph built on topic descriptions.

Taking structured data instead of unstructured text as input, Maybury (1999) addresses a specific summarizing task from an event database. The main procedures — content selection, aggregation, and presentation — are all targeted at event data. As is introduced in 2.1.4, SumGen is the implemented summary generator of this model.

2.3 Coherence-Based Approaches

Most of the researches listed so far are directed at producing summaries that are nearly as informative as the source documents. Many other researchers turn their attention to the language quality in the output. For most extractive models, good quality often distills down to coherence. In the following sections, I will review the representative works in this direction. As I mentioned, coherence can be studied both globally (2.3.1) and locally (2.3.2). Hybrid models are also being developed (2.3.3). Works in sentence ordering, a coherence-motivated topic, will be addressed in 2.3.4. Finally, in 2.3.5, I will introduce work on coherence evaluation.

Do not use "researches" and "works" as a countable noun, i.e. "Most of the research listed..."

Use reporting verbs correctly, e.g. "mentioned" means not the main point. Better choices are: "discussed" or "described".

2.3.1 Global Coherence Models

Many coherence-based approaches model coherence on a discourse level by focusing on the coherence patterns between units (e.g., sentences or clauses) of a source document. Then a global architecture can be built to represent the whole text as a tree or graph, providing clues for sentence extraction.

Use "a large number of" rather than "many".

There are various accounts for coherence relations and (Hobbs, 1985) is one of the earliest in the AI community. Hobbs's major contribution is a group of 10 coherence relations discussed along two dimensions: pragmatic function and discourse structure. Compared with some other taxonomies that result in hundreds of coherence relations (Hovy and Maier, 1995), this is a modest but representative collection: {occasion, evaluation, parallel, elaboration, background, explanation, contrast, violated expectation, generalization, exemplification}. A similar taxonomy is made by Kehler (2002), which is philosophically justifiable.

Give support to explain strong statements, i.e. "which is philosophically justifiable because..."

Coherence relations are recast as "rhetorical relations" in the seminal paper by Mann and Thompson (1988) and lay the foundation of the Rhetorical Structure Theory (RST), an extensively used model in coherence-based NLP.

Hovy (1988b), for example, applies RST to text planning, a subtask of text generation. Four coherence relations of RST are formalized to act as constraints for directing the ordering of sentences. It is one of the pioneering works in using coherence for automatic language tasks.

Do not start a sentence with "But"; use "However".

But the extensive use of RST to text summarization is usually credited to Marcu (1997, 1999, 2000). He shows that guided by rhetorical relations between clauses, it is possible to parse a discourse. In (Marcu, 1997), he implements a robust rhetorical parser by a manually built corpus and a rhetorical parsing algorithm. The output is a desirable binary tree that represents the rhetorical structure of a text. In the next step, RST discourse trees are used for sentence extraction. According to (Marcu, 1999), the salience of textual units (sentences/clauses) is determined by the depth of their tree nodes, which in turn is determined by their nucleus/satellite status constrained by the RST rhetorical relations. Then sentences/clauses can be ranked and selected according to their discourse salience. Further theoretical proofs and parsing details are provided in (Marcu, 2000).

Despite the initial success, Marcu's RST tree model is also criticized. For example, Wolf and Gibson (2004, 2006) find fault with the binary tree in RST, which they contend to be inadequate due to its structural constraints. Instead, they advocate a "chain graph structure" that can represent crossed dependencies and multiple-parent nodes and is thus descriptively more adequate than RST trees. They also study the contribution of coherence-based approaches to sentence ranking by comparing non-coherence-based models and coherence-based model, including the RST tree and their chain graph model. The best performance is given by a version of the chain graph algorithm.

✓Describes how Marcu fits into body of work
✓Highlights most important feature of work
✓Explains methodology
✓Outlines two important criticisms
✓Evaluates criticisms

Another famous criticism is made by Knott et al. (2001), who argue against a problematic rhetorical relation in RST — (object-attribute) elaboration. The authors prove that it is different from all the other rhetorical relations with empirical evidences. They suggest that elaboration is a relation based on the entity, instead of proposition, level. Therefore, they propose supplementing RST with entity-based coherence, a contribution that local coherence models can make.

✓Uses strong reporting verb to show agreement with researchers view, e.g. "proves"

2.3.2 Local Coherence Models

Unlike global coherence, local coherence is concerned with how information flows smoothly from one sentence to the next. Therefore, most researchers in this camp focus on adjacent sentence pairs and their coherence patterns manifested on the word or entity level.

The Centering Theory (CT) proposed in Grosz et al.'s (1995) seminal paper is a theoretical prototype in the local coherence literature. Though it was originally intended to deal with the linguistic problem of anaphora resolution (Beaver, 2004), it finds extensive applications in text generation and summarization.

✓Explains the importance of “Grosz et al.”s” paper using the term “seminal”

One important application is to generate metrics from CT's constraints and rules (Brennan et al., 1987) for local coherence. Karamanis and his colleagues (Karamanis, 2001; Karamanis et al., 2004a, 2004b; Karamanis and Mellish, 2005) experimented extensively with various CT-derived metrics for sentence ordering, a subtask of summarization (see more details in 2.3.4).

💡Place citations at the end of the sentence when listing multiple authors that are not part of grammatical structure.

Hasler (2004) directly applies CT's transitions to text summarization. The author undertakes two tasks about text extracts. In the first, she measures the coherence of human extracts and machine extracts of the same texts by counting different types of transition and finds similar patterns in both versions. In the second, she tests the usefulness of CT for extract evaluation. Her experiments also show a major defect inherent in CT — much useful coherence-related information may be hidden in adjacent sentences with no entity transitions.

✓Links ideas by using pronouns rather than repeating the study name, e.g. “she” and “her”

CT's limitation is also revealed by Poesio et al.'s (2004) parametric research, which discovers that many real documents do not follow the CT constraints and rules. The authors observe that CT provides at most an account of entity coherence as part of local coherence. A more comprehensive coherence account must also consider rhetorical coherence and temporal coherence.

The idea of “entity coherence”, a simplified variant of local coherence, gives rise to a wave of new research interests. Barzilay and Lapata (2005, 2008) pioneer the use of entity coherence in text summarization. They propose a novel CT-inspired, entity-based representation of text coherence. Using entity grids, the authors are able to compute the entity transitions in adjacent sentences with transition-based vectors. Coherence assessment is thus recast as a ranking task and the model is evaluated for sentence ordering and summary coherence. The results show that that a linguistically rich version (including coreference, syntax, and salience) of the model gives the best performance.

💡Clearly express time relations by using a range of tenses, i.e. “has given rise” or “gave rise”

Filippova and Strobe (2007) build on an entity-grid coherence model and extend it from coreference to semantic relatedness. They experiment on German newspaper texts and find lowered performance as compared with Barzilay and Lapata's (2005) experiment on English news texts. The important findings are: 1) coreference (word-identity) information is important; 2) entities are distributed unevenly throughout a text; 3) syntactic information helps little, if not at all. The extension from coreference to semantic relatedness, however, is not very profitable in their research.

Another effort to extend the entity-grid model is made by Nahnsen (2009), who resorts to a number of shallow features: group similarity, WordNet relations, temporal orderings, and longer range relations. The experiments basically follow the design of Barzilay and Lapata (2008). An important finding is that "group similarity + WordNet relations + Longer range relations" gives the best performance, though not as good as the "coreference + syntax + salience" in Barzilay and Lapata (2008). Another one is that the author's shallow method is not sensitive to the topic sequences captured by the entity grid method by Barzilay and Lapata (2008). But topic sequences hold important information about a specific genre or domain, which can be crucial in a coherence model.

✓Links paragraph to section using "another"
✓Clear topic sentence
✓Uses relative clause "who..." to define topic sentence
✓Highlights key findings

💡Do not use full stop before "But", comma is better.

2.3.3 Hybrid Models

Both the global coherence models and local coherence models may only reveal some coherence patterns and address some issues involved in text summarization. As it is assumed that hybrid models can combine the strengths of individual models, many researchers have explored the potentials of such combinations, including coherence-based models with lexical cohesion, coherence-based models joined with machine learning algorithms, and "global coherence + local coherence" models.

💡Use more formal expressions as quantifiers. Rather than "some" or "many", use "a limited number of", "few", "a large number of" or "numerous".

The lexical cohesion methods (see 2.2.2) are capable of capturing sentence relatedness on a word level, and it is possible to integrate lexical cohesion into a global coherence model. Such an attempt is reported by Alonso i Alemany and Fuentes Fort (2003), who build a hybrid model of text summarization that combines rhetorical relations to account for coherence and lexical chains to account for cohesion.

Other hybrid models seek to build computationally robust algorithms into coherence approaches. One example is Barzilay and Lee's (2004) HMM-derived content model in which each HMM state corresponds to a topic around which sentences are generated. In effect, the content model captures the coherence pattern as a shift between topic states. The technique is applied to extractive summarization and is compared with a KPC model (using words and locations) and a lead baseline and outperforms both.

The HMM model finds another application in the "utility-trained coherence models" developed by Soricut and Marcu (2006). Different from most other hybrid models, their model integrates a number of heterogeneous coherence models, both local ones (word-co-occurrence coherence models and entity-based coherence models) and global ones (HMM-based content models), in a log-linear fashion. The results show that the utility-trained hybrid model is a more powerful model that combines the strengths of individual models and outperforms any one of them.

Besides the various descendants of the entity-coherence model, the more general CT-based local coherence model also receives attention. (Orasan, 2003) is a case in point. The author's purpose is to develop a local coherence-based algorithm for sentence extraction by incorporating a well-studied AI algorithm, such as the evolutionary algorithm. In practice, sentences are ranked on the basis of a content-based method (e.g., the usual keywords, cue phrases, location, title, etc.) and a context-based method, which is based on the Continuity Principle derived from CT. Two algorithms are used, a greedy algorithm and an evolutionary algorithm. The results show that the evolutionary algorithm performs better than the greedy one. However, the author also shows that the Continuity Principle has played a limited role in producing quality summaries.

💡 Do not use "e.g." and "etc." in the same structure.

💡 Explain terms when introduced for first time i.e. "a greedy algorithm which is..."

Perhaps the ideal hybrid coherence-based models should come from two existing coherence-based models — a global one (like RST) and a local one (like CT) — so that both the discourse-level rhetorical structure information and the sentence/entity-level continuity information can be exploited. Indeed, this is the idea suggested by researchers working in either global coherence (e.g., Knott et al., 2001) or local coherence (Poesio et al., 2004).

Cristea et al. (1998) reports an early successful attempt in this direction. They establish the Veins Theory (VT), which extends the arguments of CT to text spans beyond adjacent units, thus addressing global coherence. It starts from the RST tree that identifies the global discourse structure with nuclear/satellite nodes and then calculates the veins of each leaf node representing a discourse unit. Next, it calculates the accessibility domains of each leaf node according to the veins and referential entities. Finally global coherence is computed by calculating the smoothness scores for CT-based transitions. The authors claim that the theory can apply to text summarization by following Marcu's (1997) basic idea. Moreover, VT can be used to summarize a given unit or sub-tree of the text.

✓Uses range of grammatical words to link sentences and ideas, e.g. "this", "they", "which", "thus"

✓Describes process linking steps of process, e.g. "starts", "then", "next", "finally" in present simple tense.

If VT is essentially an RST-based CT model in which the local coherence model dominates, Kibble and Power (2004) present a CT-guided RST model in which the global coherence model dominates. Building on the propositional representations and the established RST rhetorical structure of the text, it explores a CT-guided text generation scheme that integrates text planning, sentence planning, and pronominalization. First all text structures that can be derived from the RST structure are enumerated, then all the possible choices for the backward-looking center (CB) and preferred forward-looking center (CP) of each proposition are considered for each text structure. They are evaluated according to violations of salience, cohesion, cheapness, and continuity. The optimal solution is the choice with the least cost. In this unified coherence framework, both the text structure and the sentence-to-sentence transitions are taken into account.

✓Uses summary sentence at end of paragraph highlighting advantages of

2.3.4 Information Ordering

The ordering of information (usually extracted sentences) is a necessary phase in single-document and multi-document summarization. Although text order is often considered sufficient for SDS, the same is not true for MDS (Jurafsky and Martin, 2009: 830–831).

An intuitive ordering criterion is the chronological order, i.e., ordering sentences according to the time the events represented in the sentences take place. But as Barzilay et al. (2002) show, this time coherence has too weak constraint on the content coherence between sentences.

✓Explains key terms introduced for first time, e.g. "chronological order, i.e., order..."
✓Explains weaknesses in model

In practice, sentence ordering is often considered as a local coherence optimization problem and is generally considered NP-complete. But there are good approximation methods for solving this problem (Brew 1992; Cohen et al., 1999; Knight, 1999; Althaus et al., 2004). Althaus et al. (2004), for example, present a branch-and-cut algorithm that deals effectively with the 2-place or 3-place ordering problems.

💡Give the full form of acronym when introduced for first time, i.e. "Noun Phrase (NP) complete".

One way to capture local coherence in sentence ordering is lexical overlap. The sentences can be ordered in such a way that adjacent sentences have the greatest lexical overlap on average. This idea is implemented by Conroy et al. (2006) in their CLASSY system.

💡Only review studies linked to current study.

Lexical cohesion can also be integrated into more complicated statistical or machine learning models. Barzilay et al. (2002) explore augmenting the chronological ordering with lexical cohesion information. They first identify topically related blocks with lexical methods and then apply chronological ordering on the sub-block level. Lapata (2003) considers both lexical and syntactic features in calculating local coherence between neighboring sentences using a greedy algorithm. The experimental results are encouraging and the general method is readily applicable to text summarization. But it is outperformed by Barzilay and Lee’s (2004) content model built on HMM. Ji and Pulman (2006) extend this line of research to “historical ordering”, as opposed to majority ordering (Barzilay et al., 2002) or probabilistic ordering (Lapata, 2003). The algorithm is composed of three major steps: 1) sentence network construction, or building a sentence-probability matrix according to lexically based sentence distances; 2) sentence classification, assuming that each summary sentence represents a distinct “topic” in the source documents and using the EM algorithm; 3) sentence ordering, based on a graph constructed in a similar way to that in the majority ordering. The semi-supervised classification and historical ordering adopted is shown to overcome the deficiencies of the majority ordering, which is dependent only on relative sentence positions, and the probabilistic ordering, which takes no account of previous sentence selections.

✓Groups previous research showing relationship between studies, e.g. “built on”, “extend”, “as opposed to”

All those models manipulate topics and topic transitions in some manner. But identifying topics in general is not sufficient for sentence ordering, as is experimented on in (Bromberg, 2006). In her enriched LSA-coherence model, sentence vectors are established on word vectors and are then summed or averaged to arrive at a centroid — hypothetically the topic of the text. Experimental results show that the best result comes from this topicality-added LSA model.

✓Develops topic by linking each paragraph topic sentence to previous paragraph, e.g. “All those”, “The other way”

The other way to capture local coherence in sentence ordering is the CT-inspired entity-coherence approach, which is advocated by Barzilay and Lapata (2005, 2008). In their entity grid model, entity identification is based on coreference recognition. Syntactic roles played by entities and transitions between these syntactic roles underlie the coherence patterns between sentences and in a whole passage. An entity-parsed corpus can be used to train a model that prefers the sentence orderings that comply with the optimal entity transition patterns.

The entity-coherence approach only makes use of the rule of continuity within CT, one of the many possible contributions CT can make. But more extensive explorations of the CT resources (Karamanis, 2006, 2007; Karamanis et al., 2008) indicate that (entity) continuity may be the best CT can offer to sentence ordering.

In (Karamanis, 2006, 2007), the assumption that extending the CT model with local rhetorical coherence (Knott et al., 2001) is helpful is empirically falsified. The author uses the CT-based metrics of coherence and classification rate in the task of information ordering and finds that the baseline metric, which favors the ordering with the fewest violations of entity continuity, is the best metric, with or without an additional coherence constraint. Moreover, the introduction of local rhetorical coherence actually decreases the

✓Defines scope of findings, e.g. “with or without” performance. This surprisingly simple result is confirmed in (Karamanis et al., 2008) with the evidence from four heterogeneous corpora.

2.3.5 Coherence Evaluation

Most summarization evaluation methods, intrinsic and extrinsic alike, are content-targeted, which makes automatic evaluation possible. Coherence evaluation, on the other hand, is a very different challenge. Since coherence is ultimately a subjective criterion, most coherence evaluation methods are manually or semi-automatically done.

Given a gold standard ordering, Kendall's τ (Lapata, 2003, 2006) is proven to be the best metric for evaluating alternative orderings. The best known automatic evaluation of coherence is based on the entity grid model (Barzilay and Lapata, 2005, 2008), but it is not tailored for summary coherence, where a gold standard ordering is not available.

Further explain concepts central to current study by detailing methods used in these studies.

Lapata and Barzilay (2005) experiment with various mainstream theories and models in coherence evaluation, broadly categorized as syntactic models and semantic models. The syntactic model is based on (Barzilay and Lapata, 2005) that captures the local coherence through entity transitions. The semantic models do not concern syntactic structure or even word order. They compare three types of models: word-based (word overlap) models, LSA (distributional similarity) models, and WordNet-based (taxonomical similarity) models. The experimental results show that individually, the models that are most highly correlated with human assessment are the entity grid, the LSA (Foltz et al., 1998), and two WordNet-based models (Hirst and St-Onge, 1998; Jiang and Conrath, 1997). Collectively, the combination of the entity grid, word-overlap, LSA, Hirst and St-Onge, and Lesk (1986) models is the optimal solution.

Further develop and explain why these are optimal solutions as they are important for methodology used.

The Literature Review is an evaluation of how previous studies lead towards the current study. The Literature Review should end by clearly highlighting how previous research leads into the current study. When there are reviews of different methods employed, the end of the Literature Review should comment on how they can be combined in a way that is relevant to the current study. Longer Literature Reviews also contain a summary section.

3. Methodology

A strong assumption held by this project is that coherence pervades text summarization when readability and expressiveness are accounted for. In this section, I will describe how coherence can be integrated into text summarization at different phases and on different levels.

3.1 Post-Extractive Ordering

For popular sentence extraction-based systems, coherence comes into play when many acknowledge the lack of coherence in the output if selected sentences are arranged by some default or simple order. If ordered by coherence-motivated principles, the readability of the output summary will be significantly improved. The use of coherence at this level is very shallow in that coherence concerns are only ancillary to the core process of summarization and ordering does not change the selected content.

3.1.1 Ordering for Single-Document Summarization

Given a default ordering of extracted sentences from a source document, my purpose is to tune it locally by reordering the sentences so as to improve overall coherence. Local tuning means a compromise between: (1) enhancing local coherence between adjacent sentences, and (2) partially preserving global coherence manifested in the relative order among sentences of the same topic. The first subgoal will be achieved by extract-level sentence grouping and the second subgoal will be met by group-level sentence reordering.

✓Defines key term as it is introduced

To achieve them, many popular MDS ordering strategies such as chronological ordering, majority ordering, sentence precedence etc. are not applicable because they are tailored for sentences from different source documents. While the most relevant MDS ordering strategy is the entity coherence approach such as (Barzilay and Lapata, 2005, 2008) because entity coherence captures the concept about local coherence and is not sensitive to sentence origins. Such an approach will build on an entity-based vector space model to facilitate sentence similarity computation. The following shows the main steps of the method.

💡 Discuss theoretical background in Literature Review chapter.

1. Compute pairwise sentence similarities based on an entity-based vector space model that utilizes WordNet relations.
2. Group sentences according to their similarity.
3. Do intra-group and inter-group reordering.

✓Lists key steps

✓Use same grammatical pattern at start of each point

In the first step, I obtain a set of distinct common entities and named entities e_1, e_2, …, e_m with each entity instantiated by a group of closely related common nouns or proper nouns. Next I represent each extracted sentence as a vector of the weighted relative frequencies (*wf*) of entities in a given extracted set S_i.

$$S_i = (wf(e_{i1}), wf(e_{i2}), …, wf(e_{im}))$$

where $wf(e_{ik}) = wk \times f(e_{ik})$ and $f(e_{ik})$ is the relative frequency of e_{ik}. We define $wk = 1$ if e_{ik} is a common entity and $wk = 2$ if e_{ik} is a named entity. Based on all the sentence vectors, we compute pairwise sentence similarity as their cosine similarity. For frequency calculation, I also identify words of other classes (verbs, adjectives, adverbs) that are derivationally related to the entity-indicating nouns.

✓States model to be used

✓Explains scope

In the second step, I use sentence similarities to group sentences. Two algorithms are applicable: one is to treat sentences as vertices in a text graph and then find connected components. The text graph may not be fully connected because two sentences are connected only if their similarity is above a threshold. The second algorithm is based on sentence clustering. Following Wilpon and Rabiner (1985), I develop a modified K-means (MKM) algorithm, which leaves the number of clusters to be decided automatically. Let's denote a cluster by CL_i. Further, $Sim(CL_i)$ is the minimum similarity of vector pairs in CL_i. $MIN(Sim(CL_i))$ is the minimum of all cluster similarities and t is a threshold. The following illustrates the main steps.

✓Refers to origin of algorithm used, e.g. "Following Wilpon…"

💡Discuss key references in the Literature Review where reasons for selecting algorithm can be discussed.

```
1. Compute the centroid $CL_1$ of all the sentence vectors making up the extract;

2. Do the 1-centroid K-means clustering by simply assigning all the vectors to $CL_1$;

3. While at least 1 cluster has at least 2 sentences and $MIN(Sim(CL_i)) \leq t$, do:
3.1 If $Sim(S_m, S_n) = MIN(Sim(CL_i))$, create two new centroids as $S_m$ and $S_n$;
3.2 Do the conventional K-means clustering until clusters stabilise.
```

It is obvious that the above algorithm stops iterating when each cluster contains all above-threshold-similarity sentence pairs or only one sentence.

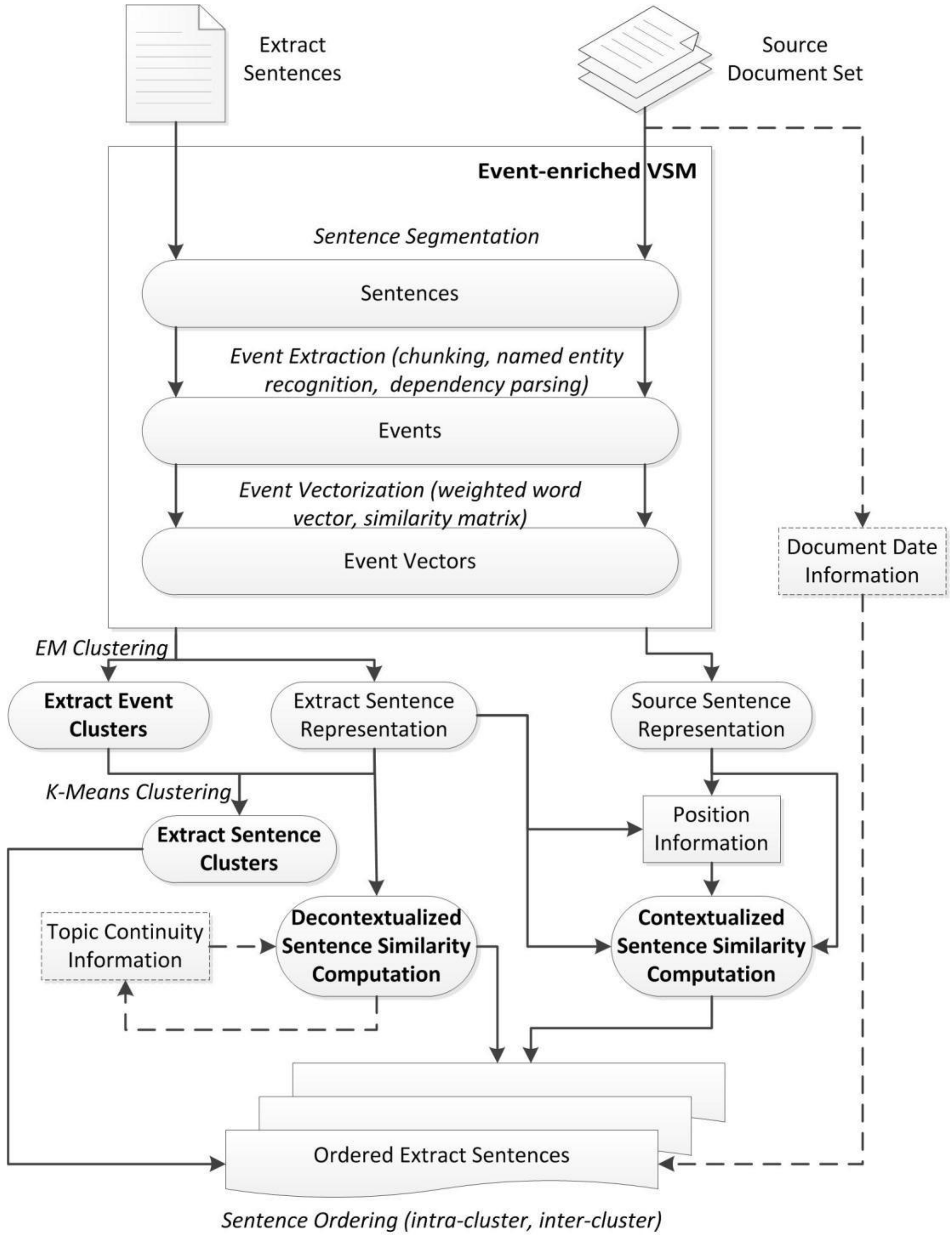
In the third step, after sentence groups (connected components or clusters) are established, inter-group ordering and intra-group ordering can be done. Inter-group ordering is oriented to global coherence. Accordingly, I first identify a “group leader” to be the textually earliest sentence in each group and then order groups according to their leader sequence (text order) in the source document. Intra-group reordering can take a global coherence-biased strategy or a local coherence-biased strategy. The former means ordering the same-group sentences according to the text order. The latter means I have to weigh textually-ordered sentences against locally more coherent adjacent sentences and prioritize the latter if conflict occurs. Note that this strategy is still anchored on text order, since in each iteration I consider the possible ways of outputting the two textually earliest sentences, either with no other sentence between them or with inserted sentences that result in better local coherence. By treating sentence groups as graphs weighted with sentence distances and using graph terminology, this task is equivalent to finding the shortest path between two vertices, which applies to both connected components and clusters treated as graphs.

Be less subjective when discussing methodology. Rather than using “I”, passive voice can be used to sound more objective, i.e. “”Group Leader was first identified…”

Use subject before “Note”, i.e. “It should be noted that…”

3.1.2 Ordering for Multi-Document Summarization

Sentence ordering is more important for MDS because the extracted sentence may come from different source documents and no “text order” applies in this situation. The basic ordering approach is clustering-based, but more dimensions are considered. What I describe below is the methodology implemented in a published work (Zhang et al., 2010). The following is the system framework.


Extract Sentences
Source Document Set
Event-enriched VSM
Sentence Segmentation
Sentences
Event Extraction (chunking, named entity recognition, dependency parsing)
Events
Event Vectorization (weighted word vector, similarity matrix)
Event Vectors
Document Date Information
EM Clustering
Extract Event Clusters
Extract Sentence Representation
Source Sentence Representation
K-Means Clustering
Extract Sentence Clusters
Position Information
Topic Continuity Information
Decontextualized Sentence Similarity Computation
Contextualized Sentence Similarity Computation
Ordered Extract Sentences
Sentence Ordering (intra-cluster, inter-cluster)

First, an event-enriched VSM is used to replace the traditional term-based VSM for sentence representation. Following (Li et al., 2006), I define events as structured semantic units that consist of *event terms* and *event entities*, which are all event *elements*. An event *E* has one event term *Term*(*E*) and a set of event entities *Entity*(*E*). Event terms are typically action verbs that denote actions or activities or deverbal nouns, which are

Label charts, tables and diagrams with a clear title and number, i.e. "Figure 1".

Refer to figures and charts, i.e. "Table 1 shows that firstly…"

grammatically nouns but function like verbs. Event entities are all text-level entities, including named entities and common entities. Unlike the triplets (two named entities and one connector) in (Filatova and Hatzivassiloglou, 2003), an event in our model can have an unlimited number of event entities. Such event-enriched representations help to alleviate the semantic deficiency problem in the traditional VSM. The conversion of events to vectors is similar to sentence vectorization in the traditional VSM. The only subtlety is that unlike bag-of-word sentences, events are structured, with event terms and entities on different conceptual levels. Our strategy is to "flatten" such a structure by organizing all corpus-wide event elements into a concatenation of all event terms and all event entities in that order. Given *m* distinct event terms and *n* distinct event entities, each event can be converted to an *m+n*-dimension vector with ternary values {0, 1, 2}. For event terms and common event entities, 1 and 0 denote the existence or non-existence of an element. For named entities, non-existence is denoted by 0 but existence by 2. The term-entity flattening is important for constructing a similarity matrix to compute event similarity.

Discuss previous studies and methods in Literature Review.

Do not bracket references when they are part of the grammatical structure, i.e. in Filatova and Hatzivassiloglou (2003).

The basic idea of computing event similarity is to multiply the event vector $\vec{E}$ with a similarity matrix *W* to get a new vector $\vec{E}$ ', after a similar technique taken by Stevenson and Greenwood (2005) to compute pattern similarity. With *m+n* event elements e_1, … , e_m, e_{m+1}, …, e_{m+n} (*m* terms + *n* entities), *W* is an (*m n m n* u) ( ) matrix with each w_{ij} denoting the similarity between e_i and e_j. As event terms and entities are situated at different conceptual levels, their similarity is assigned 0. Specifically,

$$\begin{cases} Sim_{ET}(e_i, e_j) & 1 \le i, j \le m \\ Sim_{EE}(e_i, e_j) & m+1 \le i, j \le m+n \\ 0 & \text{otherwise} \end{cases}$$

WordNet is used to populate the symmetric matrix *W*. The similarity between two event terms, $Sim_{ET}(e_i, e_j)$, is defined as the maximum Jiang-Conrath similarity (Jiang and Conrath, 1997), $Sim_{JCN}(e_i, e_j)$, between their WordNet senses.

Centre equations and formula.

Refer to formulae in-text, i.e. "This can be expressed as".

Number formulae as they appear.

$$Sim_{ET}(e_i, e_j) = Sim_{JCN}(e_i, e_j) = \max_{\substack{s\ senses\ e□(_i),\\ s'□senses\ e(_j)}} 1/\ (IC(s)\ IC(s')\ \text{-2x}\ IC\ (lcs(s\ ,s')))$$

IC is the Information Content from corpus statistics and *lcs*(*s*, *s'*) is the least common subsumer or most specific ancestor node of senses *s* and *s'*.

Computing the similarity between two event entities, $Sim_{EE}(e_i, e_j)$, is slightly more complicated. Because of WordNet's limited coverage of proper nouns, the Jiang-Conrath similarity may not apply to two named entities (and returns 0). Therefore we also compute a word overlap score $Sim_{WO}(e_i, e_j)$ between two named entities as follows.

$$Sim_{WO}(e_i, e_j) = \frac{|Word(e_i) \cap Word(e_j)|}{|Word(e_i) \cup Word(e_j)|}$$

Word(*e*) is the set of all words in *e*. This score captures the surface similarity between two named entities. Then for two named entities e_i and e_j, we take the maximum of $Sim_{WO}(e_i, e_j)$ and $Sim_{JCN}(e_i, e_j)$. If one of them is a common entity, we still use $Sim_{JCN}(e_i, e_j)$.

✓ Uses "we" rather than "I" when referring to methods used

After *W* is established, computing the similarity between two event vectors E_i and E_j, $Sim_E(E_i, E_j)$, is straightforward.

$$Sim_E(\overrightarrow{E}_i, \overrightarrow{E}_j) = Sim_{COS}(\overrightarrow{E}_i W, \overrightarrow{E}_j W) = \frac{\overrightarrow{E}_i W \cdot \overrightarrow{E}_j W}{\|\overrightarrow{E}_i W\| \|\overrightarrow{E}_j W\|}$$

Sim_{COS} is the cosine similarity, a standard distance measure for high-dimensional vectors.

Second, we derive sentence similarity from event similarity and adopt a novel two-layered clustering based on the event similarities. The first layer clustering is on events. Because hard clustering of events, such as K-means, will result in binary values in sentence vectors and data sparseness, we use a soft clustering technique, the Expectation-Maximization (EM) algorithm, and assume a Gaussian mixture model for event vectors. An outcome of the EM clustering of events is that each sentence event is assigned a probability distribution over all event clusters. Next, we vectorize a sentence by summing up the probabilities of its constituent event vectors over all event clusters (*EC*s) and obtaining an *EC*-by-sentence (S_n) matrix $\boldsymbol{S} = [s_{ij}]$.

Do not start a sentence with "because".

$$s_{ij} = \sum_{E_r \in S_j} P(\overrightarrow{E}_r | EC_i)$$

where E_r is the corresponding vector of event E_r. Compared with the traditional term- or word- based sentence vectorization, the event-enriched VSM enables event information to be coded in sentence representation via soft clustering, thus endowing sentence vectors with coarse-grained semantics. The second-layer clustering is then on sentences, where hard clustering is sufficient because we need definitive, not probabilistic, membership information for the next step – sentence ordering. The popular K-means is used for the purpose.

To alleviate data sparseness and leverage the latent "event topics" among the event elements, I used the Latent Semantic Analysis (LSA, Landauer and Dumais, 1997) approach by doing Singular Value Decomposition (SVD). We apply LSA-style dimensionality reduction to the event element-by-event matrix and the event-by-sentence matrix by doing SVD. A problem is the selection of the reduced dimensionality, which affects the performance of dimensionality reduction. I adopt a utility-based metric to find the best *h** for the clustering purpose by maximizing intra-cluster similarity and minimizing inter-cluster similarity. The attested Davies-Bouldin index (*DB*, Davies and Bouldin, 1979) is used for that purpose.

Do not use both "I" and "we" when describing methodology.

Third, I need to group and order sentences based on their similarity. Two types of sentence similarity are computed: decontextualized sentence similarity and contextualized sentence similarity. The former is to only look at the sentences by themselves, or to treat them as isolated and decontextualized. The decontextualized sentence similarity $Sim_{-C}(S_i, S_j)$ is defined as the maximum event similarity between their events.

$$Sim_{-C}(S_i, S_j) = \max_{\substack{e\in Event(S_i)\\ e'\in Event(S_j)}} Sim_E(\bar{e}, \bar{e}')$$

Event(*S*) is the set of events contained in *S*. This measure suffices for truly decontextualized sentences, but the fact is that the two extract sentences do not come from nowhere. Suppose we are to decide how well S_2 succeeds S_1 in the new extract context, we should also seek clues from their source context, which is inspired by the "sentence precedence" by Okazaki et al. (2004). Therefore, contextualized sentence similarity $Sim_{+C}(S_i, S_j)$ measures to what degree S_i and S_j resemble each other's relevant source context. More formally, let $LC(S_i)$ and $RC(S_i)$ be the left source context and right source context of S_i respectively and suppose S_i and S_j are to be arranged in that order in the new extract, $Sim_{+C}(S_i, S_j)$ is defined as follows.

$$Sim_{+C}(S_i, S_j) = \frac{Sim_{-C}(S_i, LC(S_j)) + Sim_{-C}(S_j, RC(S_i))}{|LC(S_j)| + |RC(S_i)|}$$

I can simply take $LC(S_i)$ and $RC(S_i)$ to be the left adjacent sentence and right adjacent sentence of S_i in the source document, but expanding the context range to more than one sentence is also feasible. The final score for the similarity of S_i and S_j, $Sim_S(S_i, S_j)$, is the product of $Sim_{-C}(S_i, S_j)$ and $Sim_{+C}(S_i, S_j)$.

With sentence clusters and sentence similarity measures, we are ready to order sentences to maximize sentence coherence within a cluster and between neighboring clusters. Therefore, the ordering algorithm is composed of intra-clustering ordering and inter-cluster ordering, motivated by local coherence and global coherence in block-style writing. Using the heuristic of time and textual precedence, I first generate a set of possible leading sentences $L = \{L_i\}$ as the intersection of the document-leading extract sentence set L_{Doc} and the time-leading sentence set L_{Time}. If L is a singleton, finding the leading sentence S_L is trivial. If not (when more than one document are published on the same earliest date), S_L is decided to be the sentence in L most similar to all the other sentences in the extract so that it qualifies as a good topic sentence.

✓Uses a range of structures to express reason, e.g. "to maximize", "therefore", "motivated by"

💡Refer to formulae in text, i.e. "This is shown below:"

$$S_L = \arg\max_{L_i \in L} \sum_{L' \in P \setminus \{L_i\}} Sim_S(L_i, L')$$

After the leading sentence is determined, we identify the leading cluster it belongs to. Intra-clustering ordering now starts with this cluster. We adopt a greedy algorithm, which selects each time from the unordered sentence set a sentence that best coheres with the sentence just ordered. The selection of the next best sentence is according to both decontextualized similarity and contextualized similarity. After all the sentences in the current sentence cluster are ordered, we select the next sentence cluster and do the intra-cluster ordering again. We iterate this process until all the sentences in the extract are ordered. The remaining question is determining the next best sentence cluster. Given a processed sentence cluster SC_i, the next best sentence cluster SC_{i+1} among candidate SC_j's is the one that maximizes the cluster similarity $Sim_{CLU}(SC_i, SC_j)$. Since clusters are collections of sentences, their similarity should be measured in terms of all cross-cluster sentence similarities.

$$Sim_{CLU}(SC_i, SC_j) = \frac{\sum_{S \in SC_i, S' \in SC_j} Sim_S(S, S')}{\left| SC_i \times SC_j \right|}$$

In my implementation of the above algorithms (with results reported in Section 4.2), I have also introduced some factors to enhance coherence, such as the CT-inspired topic continuity and the chronological order based on the document date.

3.2 Coherence-Based Extraction

A deeper integration of coherence concerns with extraction-based systems is to select sentences on coherence grounds. If extracted sentences are well connected in their content, the output tends to demonstrate better global coherence than otherwise. There are two key aspects about the proposed method: recognizing sentences with globally coherent details and selecting candidate sentences for extraction.

💡 Avoid weak "There are…" structures if the object can be the subject, i.e. "Recognizing sentences…and selecting…are two key aspects of the proposed method".

3.2.1 Recognizing Sentences with Globally Coherent Details

Compared with post-extractive ordering, coherence-based extraction is more concerned with global coherence because sentence extraction is determined primarily by how they are connected in terms of content, and secondly by how much salient information they convey. Psychological experiments show that human readers are sensitive to the global coherence manifested as the connectedness between sentences over long distances in text (Tapiero, 2007). Sentences are typically connected via some discourse-level rhetorical relations (2.2.5) such as topic-elaboration, cause-effect, or continuance. If the extracted sentences contain details that relate in one of those ways, the extract is expected to be globally more coherent than an extract constructed otherwise.

✓Gives a clear explanation of key concepts:

a. Compares

b. Elaborates definition by citing key points

c. Gives examples

d. Explains examples

The first step to attain this goal is to identify a set of textual details that are globally coherent. The details may be predefined or induced. In case they need to be induced, for query-focused summarization, such details are selected around the query (e.g., what happened, when, where, who were involved, etc.). For generic summarization, such details can be induced by salient information in the document(s) and detail lists manually composed or automatically learned.

After the coherent details are obtained, the next task is to recognize sentences bearing them, which is defined as a classification problem. Moreover, as a sentence can have more than one textual detail, it is a multi-label classification problem.

The objects to be classified are sentences, with features incorporating lexical, syntactic, and semantic information since textual details are free-form content units that can reach beyond the text surface. I have used two kinds of features for the purpose. The first one is the usual word unigram features. The second one is meta-phrase features. A meta-phrase is a 2-tuple (m_1, m_2) where m_i is a word/phrase or word/phrase category, which is a syntactic tag, a named entity (NE) type, or the special /NULL/ tag.

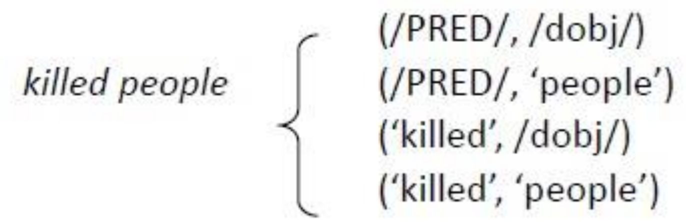
Syntactic tags represent the logical and syntactic attributes of words in a sentence, including 2 logical constituents: predicate and argument, and 11 grammatical roles: nominal subject, controlling subject, passive nominal subject, direct object, indirect object, agent (passive verb complement), prepositional modifier, adjectival modifier, appositional modifier, noun modifier, and abbreviation modifier. A predicate can be a verb, noun, or adjective and an argument is a noun. The combination of syntactic tag and/or word gives rise to meta-phrases of the syntactico-semantic pattern, including the predicate-argument pattern and the argument-modifier pattern.

Spell numbers from 1-10, i.e. "Two".

Give examples when introducing key constituents.

NE types represent the semantic attributes of special NPs in a sentence, which are indicative of particular types of textual details like time and place. I use 6 NE types: person, organization, location, date, money, and percentage. The combination of NE type and/or NE word/phrase gives rise to meta-phrases of the name-neighbor pattern, including the left neighbor-name pattern and the name-right neighbor pattern.

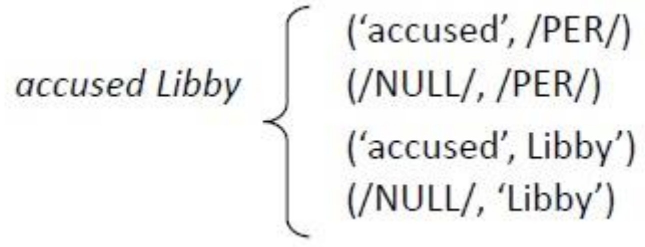
For syntactico-semantic patterns, two related words and their syntactic tags give a total of 4 combinations as shown in the following.



killed people

(/PRED/, /dobj/)
(/PRED/, 'people')
('killed', /dobj/)
('killed', 'people')

For name-neighbor patterns, an NE or its type alone (with the /NULL/ tag) or with its left/right neighbor give 4 combinations as shown below.



accused Libby

('accused', /PER/)
(/NULL/, /PER/)
('accused', Libby')
(/NULL/, 'Libby')

Such syntactico-semantic and name-neighbor meta-phrases are designed to capture concept relations and NE contexts at different levels of abstraction.

Name-neighbor meta-phrase extraction is a simple extension of NE recognition; syntactico-semantic meta-phrases are extracted in three scans as predicate-argument or argument-modifier relations are extracted via dependency parsing.

1. Find all predicate-argument pairs in the sentence from dependency relations: nominal subject, direct object, agent, etc.;
2. Find all nominal argument modifiers from dependency relations: noun modifier, appositional modifier, etc.;
3. Find all adjectival argument modifiers from the dependency relation of adjectival modifier.

To classify sentences as multi-label objects, I adopt the popular binary decomposition methods (Boutell et al., 2004; Tsoumakas and Katakis, 2007), instead of label combination, because the latter does not guarantee that sufficient training data are available for each transformed single-label class and cannot entertain the possibility that different types of textual details (e.g., simple vs. inferred) are amenable to different features. Only by doing binary decomposition can I treat each textual detail (class) differently and explore the optimal feature sets for each detail.

✓Lists each step using parallel structures

As the basic binary decomposition approach does not consider label dependencies or object dependencies, I will extend it with three methods: stacking, chain, and context. The stacking approach (Wolpert, 1992) is based on Godbole and Sarawagi's (2004) first improvement on their Support Vector Machine (SVM) classifier. Initially, a classifier is trained and all labels are predicted using binary decomposition. In the second round, the predicted labels are treated as new features and used to augment the feature space of the training data. Then the augmented training data are merged with the original training data for re-training and re-predicting. The algorithm is iterated until classification results converge.

Give examples when introducing key constituents.

The chain extension is proposed by Read et al. (2009), which leverages label dependencies in an incremental way. Its major difference from stacking is that in each iteration, only one label is predicted and then the newly predicted label is used to augment the feature space of the training data. Therefore, the algorithm is iterated *k* times, each time doing one binary classification.

✓Compares and contrasts using a wide range of grammatical structures, i.e. "Both", "but", "besides", "neither", "nor".

Both stacking and chain are designed to leverage label dependencies unaddressed by binary decomposition. But besides textual detail dependencies, sentences are also interrelated. For example, if an "attack issue" is found in a sentence, the following sentence probably describes "casualties". Neither stacking nor chain can capture this kind of dependency. Our solution to the object dependencies is the context extension, i. e., in each iteration, each sentence's feature vector is augmented with the newly predicted labels of its adjacent sentences in the source document, which are the context of the current sentence. In case the current sentence is the first or last sentence in its source document, default labels (0’s) are used for an absent adjacent sentence. Similar to stacking, the context algorithm is iterated until classification results converge.

Each of the extension methods can be considered as a local strategy that addresses one aspect of the problem. The ensemble method represents a global strategy that combines local concerns and often improves overall performance (Tsoumakas and Vlahavas, 2007; Read et al., 2008). I apply two ensemble extensions. The first one follows (Read et al., 2009) by using the ensemble method on multiple chains to neutralize order sensitivity. The second combines stacking, context, and chain extensions. The results are combined by majority vote or one vote (positive set is the union of positive objects predicted by each classifier). The latter is chosen because it outperforms majority vote in my experiments.

3.2.2 Selecting Sentences for Coherence-Based Extraction

After sentences with globally coherent details are recognized, the next step is to select extract-worthy sentences from the set of such sentences, a subset of all document (set) sentences. I can adapt the mainstream shallow feature-based models (2.2.1) or graph-based models (2.2.4) to accommodate content salience and redundancy control, two major concerns for non-coherence-based extraction.

Explain key concepts when first introduced, i.e. content salience <u>which is</u>…”

Two important changes to the current models are that content salience is ultimately measured in terms of details (instances) instead of words, and that redundancy control is also ultimately applied to details (instances) instead of words. The goal is to select sentences with globally coherent and salient details and with minimal detail overlap.

A more challenging route is to advance to the phrase level and identify the exact detail instances in the recognized sentences. Then such identified detail instances can act in lieu of the “words” in non-coherence-based extraction. An implemented method on the TAC 2010 summarization track is to rank sentences according to their detail instance number and diversity. For detail a_i and its jth instance a_{ij}, we score a_{ij} according to the frequency of a_i ($freq(a_i)$) and the percentage of patterns that recognize a_{ij} ($support(a_{ij})$), thus preferring high-fidelity and rare detail instances.

✓Comments on difficulty, i.e. “A more challenging route”

$$Score(a_{ij}) = support(a_{ij}) \,/\, freq(a_i)$$

The sentence score is the sum of all its aspect instance scores normalized by sentence length.

$$Score(S) = \sum_{a_i \in S} \max(Score(a_i)) / |S|$$

where *Score*(a$_i$) is the sum of all the *i*th aspect instance scores in *S*. We use max(*Score*(a$_i$)) because it is possible for a sentence fragment to be recognized as different detail instances.

In the spirit of MMR (Carbonell and Goldstein, 1998), after the highest-ranking sentences is selected to generate the summary, all the *Score*(a_{ij}) are discounted with reference to the similarity between a$_{ij}$ and any same-aspect instances contained in the selected sentence. The process is iterated until the summary word length is reached.

3.3 Coherence-Based Revision

Aside from disorder and global disconnectedness, the lack of coherence in most extracts is also attributed to the limitation of the extractive method itself. The next challenge for this project is to revise extracted sentences out of coherence concerns. Despite many sentence revision/reduction/compression efforts, few are directly motivated by coherence. My description of this part is rather sketchy because most of it is still on the conceptual level and will be solidified in the future.

✓Introduces next section and comments the contents, i.e. "rather sketchy'

3.3.1 Revising Sentences for Global Coherence

A marked difference between automatic summarization and human summarization is that in human summaries, many sentences are reduced or merged to strengthen their bond with other sentences. In other words, such reduction or merging results in better global coherence.

I wish to simulate the human strategy automatically. After sentences are selected, by non-coherence-based or by coherence-based methods (3.2), they should be semantically checked for anything irrelevant to the other selected sentences as a whole. For news articles, citation contexts and entity introductions are typical examples. After the globally irrelevant pieces of information are identified, they should be removed so that the reduced sentences fit better in the global textual fabric. As removing sentential parts will potentially lead to ungrammaticality, this operation should be based on parsing because only syntactically detachable units can be safely removed.

💡State main point first, i.e. "Typical examples are…"

On the other hand, semantically close or complementary sentences can be merged into a new sentence so that closely related pieces of information (e.g., time and place of some happening) are not scattered around. If sentence merging is infeasible or potentially leads to long and unreadable sentences, connectives can be added according to the rhetorical relations between sentences – a strategy often adopted by human summarizers and writers.

This part of work should be done in a machine learning framework, like that developed by Barzilay and Lee (2003) for paraphrase learning. Some annotated data for sentence revision are needed and some useful revision templates are expected to be learned.

✓Refers to future work with a range of structures to show possibility, i.e. "should be", "are needed", "are expected to"

3.3.2 Revising Sentences for Local Coherence

It is acknowledged that the disharmony between adjacent summary sentences is often due to two reasons: disorder and referential vagueness. Post-extractive ordering (3.1) is proposed to overcome the former, but to address the latter, content has to be changed. Therefore, the first drive to revise sentences is to enhance referential clarity and local coherence. Ideally, anaphors should be resolved and co-referring NPs be unified (e.g. "Obama" and "the US president").

More generally, contextualized information conveyed by selected sentence should be either specified ("the spot" ⇨ "the spot where the crime took place") or suppressed, i.e., deleted if the contextualized information is unimportant and the deletion is grammatically possible.

On the sentence level, local coherence is often correlated with entity and event continuity in adjacent sentences. Entity or topic continuity can be modeled according to the Centering Theory, which I have implemented in the MDS ordering task (3.1.2). But as sentence revision is concerned, I will work at the deletion of superfluous entities and the explicit presentation of implicit entities. My previous work on the event-enriched VSM (3.1.2) can also lay the foundation for enhancing event continuity. In this task, not only event entities but also event terms are to be inspected for deletion or explicit presentation.

4. Preliminary Results and Discussions

💡Refer to discussion as an uncountable noun in reports, i.e. "Discussion".

Currently, some subtasks of the proposed methods in Section 3 have been implemented. In this section, I will report the preliminary results of three subtasks: 1) ordering for single-document summarization (4.1); 2) ordering for multi-document summarization (4.2); and 3) recognizing sentences with textual details (4.3). Discussions of their significance are also provided.

✓Refers to contents of section

💡Do not mix "I" and passive voice when outlining the next section.

4.1 Ordering for Single-Document Summarization

In order to fully evaluate the single-document reordering schemes against the goal of coherence improvement, I experimented with three news datasets, each characterized by some unique linguistic and stylistic features. The first dataset D400 consists of short (about 400w) documents from the Document Understanding Conference (DUC) 01/02 test set, which were manually confirmed to be IP-structured hard news articles. The second dataset J1k consists of medium-length articles (about 1000w) selected from popular English journals such as *The Wall Street Journal*, *The Economist*, *The Washington Post*, *Time*, etc. I manually check them to ensure they are all soft news. The third dataset D2k consists of very long articles (about 2000w) randomly selected from the DUC 01/02 test set, all of which are manually verified to be soft news and non-IP-structured. Each set contains 60 documents, resulting in a total of 180 documents.

✓Outlines key data used in the experiment

💡 Consider including experimental methodology in methodology chapter,

To prepare the test set, I produced 25% extracts for D400 to meet the 100w DUC requirement. Assuming that reordering works better with longer extracts (> 5 sentences), I produced 20% extracts for J1k and 10% extracts for D2k so that the extracts in these two sets are of comparable lengths. Since sentence extraction is not our focus, the 180 extracts are produced with a simple but robust summarizer built on tf.idf and sentence position (Aone et al., 1999).

💡 Use only past simple when describing experiments, i.e. "determined" and avoid using present simple, e.g. "are produced".

Using the textually ordered extracts as baselines, I empirically determined coefficient *c* and produced all the CC-grouped and MKM-grouped versions. For better comparison, we also produced a ranking-based ordering (higher-ranking sentences preceding lower-ranking sentences) obtained from the summarizer and a randomly shuffled version for each baseline ordering.

The automatic evaluation consists of local coherence evaluation and reordering evaluation. In order to provide reference orderings for both tasks, I followed Madnani et al.'s (2007) recommendation and employed 3 human annotators, all native speakers of English, to provide reference orderings. Each of them was asked to reorder all the 180 shuffled extracts to optimize coherence and mark paragraph (of at least 2 sentences) boundaries, which will be used by one of the evaluation metrics.

4.1.1 Local Coherence Evaluation

For each ordering, I define the Local Coherence (*LC*) score as the average sentence similarity of all adjacent sentence pairs. The sentence similarity is calculated as their cosine. The following table shows the result, where I report the average *LC* scores for each category.

✓Explains key features of the table

“Baseline” is the textually ordered extract; “Random” and “Ranking” are randomly shuffled and ranking-based extracts, respectively; “References” are human orderings. “CC-G”, “CC-L”, “MKM-G”, and “MKM-L” correspond to the four different reordering schemes based on connected components or modified K-means clustering, with globally or locally biased grouping (3.1.1). For each news category, I conducted two-tailed t-tests between the baseline and all the other versions and mark statistical significance with * ($p < 0.05$).

	D400	**J1k**	**D2k**
Baseline	0.1428	0.0894	0.0924
Random	0.1465	0.0832	0.0732
Ranking	0.1489	0.0750	0.0808
CC-G	0.1663*	0.1120*	0.1186
CC-L	0.1691*	0.1176*	0.1296*
MKM-G	0.1587*	0.0983	0.1044
MKM-L	0.1587*	0.0983	0.1044
Reference 1	0.1624*	0.0906	0.1023
Reference 2	0.1542	0.0881	0.0948
Reference 3	0.1560	0.0922	0.0995
Avg(Reference)	0.1575	0.0903	0.0989

Give the table a title.

Number tables.

As expected, the CC and MKM reordered extracts demonstrate considerable local coherence improvement compared with the baselines, with percentage gain up to 40.26% (CC-L for D2k). In most categories, the improvement is statistically significant, especially for CC-G and CC-L. This clearly shows the textually ordered baselines are not locally coherent in all news categories.

The CC versions consistently score the highest in all categories, showing that the graph algorithm is effective for enhancing local coherence. Moreover, the local coherence-biased CC-L is the top scorer with significant improvement over the baseline in all categories. By comparison, the MKM versions are not as effective as CC versions in boosting local coherence and there is no difference between the MKM-G and MKM-L scores in all categories. My explanation is that clustering has produced groups of sufficiently coherent sentences, for which a second-level local coherence tuning is wasteful.

✓Comments on key results using key numbers in the text, e.g. “40.26%

✓Compares key findings, i.e. “not as effective as”

✓Explains noticeable results, i.e. “My explanation is…”

The baseline score for D400 consisting of “IP + hard news” article extracts is lower (though not significantly) than a randomly ordered version and a ranking-based version, which is not observed in J1k or D2k baselines. It seems textually ordering such short

✓Explains results using tentative language, i.e. “seems” and “may” extracts degrades local coherence because many links among the sentences in the source documents are broken. Sentence links are also broken in longer extracts, but to a lesser degree as some of the links may be recovered in more selected sentences.

It is interesting to find that the reference orderings are not optimized for local coherence, scoring midway in the range for each set. This fact confirms the assumption that overall coherence is more than local coherence and factors other than local coherence are considered in human summarization.

4.1.2 Reordering Evaluation

In order to evaluate the overall efficacy of our reordering algorithms, I evaluated the reordered versions against the 3 reference orderings because of the variability among human orderings (Madnani et al., 2007). I computed the average score of each test ordering against the 3 reference orderings by using 3 different metrics.

The first metric is Kendall's τ (Lapata 2003, 2006), which has been reliably used in ordering evaluations (Bollegala et al., 2006; Madnani et al., 2007). It measures ordering differences in terms of the number of adjacent sentence inversions necessary to convert a test ordering to the reference ordering.

✓Includes clear topic paragraph giving purpose, i.e. "In order to"
✓Outlines content of section, i.e. "3 different metrics"
✓Gives clear topic sentences referring to introduction paragraph, i.e. "The first metric is…"

$$\tau = 1 - \frac{4m}{(N-1)}$$

In this formula, m represents the number of inversions described above and N is the total number of sentences.

The second metric is the Average Continuity (AC) proposed by Bollegala et al. (2006), which captures the intuition that the quality of sentence orderings can be estimated by the number of correctly arranged continuous sentences.

$$AC = \exp\left(\frac{1}{k-1}\sum_{n=2}^{k}\log(P_{\mathrm{n}} + \alpha)\right)$$

In this formula, k is the maximum number of continuous sentences, α is a small value in case P_n = 1. P_n, the proportion of continuous sentences of length n in an ordering, is defined as $m/(N - n + 1)$ where m is the number of continuous sentences of length n in both the test and reference orderings and N is the total number of sentences. Following (Bollegala et al., 2006), I set $k = MIN(4, N)$ and $\alpha = 0.01$.

I also go a step further by considering only the continuous sentences in a paragraph marked by human annotators, because paragraphs are local meaning units perceived by human readers and the order of continuous sentences in a paragraph is more strongly grounded than the order of continuous

✓Gives good clear summary sentence leading to the formula, i.e." This is the third…"

sentences across paragraph boundaries. So in-paragraph sentence continuity is a better estimation for the quality of sentence orderings. This is the third metric: Paragraph-level Average Continuity (*P-AC*).

$$P\text{-}AC = \exp\left(\frac{1}{k-1}\sum_{n=2}^{k}\log(PP_{n} + \alpha)\right)$$

Here $PP_n = m'/(N - n + 1)$, where m' is the number of continuous sentences of length n in both the test ordering and a paragraph of the reference ordering. All the other parameters are as defined in *AC* and P_n.

The full set of results for the three datasets is shown in the following table. Note that when the grouping threshold is very high or low (c = 0), the reordering algorithms find either one-sentence groups or one group of all sentences and the CC or MKM reordering is reduced to text ordering. A reordering score equal to the baseline is the consequence of either of those two cases, which suggests the failure of a reordering scheme. I mark such scores with a cross line. For each category, I conducted the two-tailed t-test between the top scorer and all the other versions and mark statistical significance with * (p < 0.05).

✓Introduces table, .i.e." the following table..."
✓Explains content

	τ	*AC*	*P-AC*
D400			
Baseline	0.6573	0.4452*	0.0630
Random	0.0966*	0.2120*	0.0528*
Ranking	0.6563	0.4419*	0.0623
CC-G	**0.7286**	**0.5688**	**0.0749**
CC-L	0.7094	**0.5688**	0.0714
MKM-G	0.6735	0.4670	0.0685
MKM-L	0.6722	~~0.4452~~*	0.0679
J1k			
Baseline	0.3276	0.0867*	0.0428*
Random	0.0032*	0.0343*	0.0085*
Ranking	0.2504*	0.0432*	0.0149*
CC-G	0.3324	0.0979	0.0463*
CC-L	~~0.3276~~	0.0923	~~0.0428~~*
MKM-G	**0.3390**	**0.1152**	**0.0602**
MKM-L	0.3381	0.1128	0.0588
D2k			

Baseline	0.3125	0.1622	0.0213
Random	0.0833*	0.0137*	0.0058*
Ranking	0.2254*	0.0199*	0.0102*
CC-G	**0.3389**	**0.1683**	**0.0235**
CC-L	0.3278	**0.1683**	0.0229
MKM-G	~~0.3125~~	0.1634	0.0216
MKM-L	~~0.3125~~	0.1630	0.0216

Expectedly, my algorithms work for both J1k and D2k, datasets of news articles that deviate from the "IP + hard news" paradigm and in the best situation, the improvement over the baseline is more than 40% (J1k, MKM-G measured by *P-AC*). They also work for D400, where the largest and statistically significant improvement is over 25% (CC-G/CC-L, measured by *AC*). This is hard evidence that in short extracts (≤ 5 sentences) of "IP + hard news" articles, global coherence alone cannot guarantee overall coherence. This further shows that the lack of local coherence cannot be compensated for by the global coherence-maximized textual ordering.

✓Reports key findings using data from table
✓Discusses importance of findings, i.e. "This further shows", "more effective", "most significant"

Empirically the CC algorithms are more effective for either very short or very long source documents (D400 and D2k), whereas the MKM algorithms are more effective for medium-length documents (J1k), where the most significant improvement is found. Therefore, local tuning works best for "middle-type" news articles, which are midway in the length range and the "IP + hard news" typicality, and the choice of an optimal reordering algorithm is sensitive to the stylistic features of source documents.

The improvement for D2k – a group of documents deviating the most from the "IP + hard news" paradigm – is the slightest among the three sets. I manually examined our dataset and found that the documents of this set possess global features such as chronological, biographical, or narrative sequence that play a dominant role. So the text order renders good orderings that leave very limited space for improvement by local tuning.

✓Explains and discusses reasons for findings

The statistics also show that the second-level local tuning (for intra-group reordering) is unprofitable as it either improves nothing or slightly downgrades the reordering performance. Note that although the CC-L orderings achieve higher *LC* scores than the corresponding CC-G orderings, they do not result in better reordering quality. The ranking-based orderings score consistently lower than the baselines, showing that the ranking order is an untenable ordering heuristic.

💡Do not introduce sentence with "note". Instead write: "It should be noted that…"

4.2 Ordering for Multi-Document Summarization

In this section, I report the experimental results of MDS ordering by applying the event-enriched VSM and ordering algorithm. The first experiment is to automatically evaluate the quality of system orderings with reference to human orderings. The second experiment is to recruit human judges to rate different orderings for a different set of extract sentences but from the same source documents.

✓Explains content of section

4.2.1 Automatic Evaluation

I use the dataset of the DUC 02 summarization track for MDS because it includes an extraction task for which model (human) extracts are provided. For each document set, 2 model extracts are provided each for the 200-word and 400-word length categories, which enables us to experiment with those extracts (using 1 randomly chosen model extract per document set per length category) by applying our ordering algorithm and evaluate the output against the model extracts which represent the gold standard orderings. 42 200-word extracts and 39 400-word extracts are collected to make the experimental dataset.

Use past simple tense to explain methods and findings

I want to evaluate the validity of event coherence-based ordering as against entity-based ordering and the role played by performance boosters, including topic continuity, time penalty, and LSA-style dimensionality reduction. Therefore I produce two sets of 4 peer orderings based on event coherence and entity coherence respectively. Each set consists of a version with all the three performance boosters (EventAll and EntityAll) and three versions corresponding to the absence of one of the performance boosters (EventNoTC, …, EntityNoTC, …). For the entity coherence-based orderings, sentences are converted to entity vectors before being multiplied by an entity-only similarity matrix. Sentence clustering is done by one-layered K-means based on the cosine between such vectors. The ordering details are the same as event coherence-based orderings. In addition, I produce a random ordering and a baseline ordering. The baseline only uses chronological and textual order. Source document with extracted sentences are ordered by their publication date from least to most recent. Sentences in the same documents are then textually ordered. Source documents published on the same date are randomly ordered. The following table lists the 10 peer orderings to be evaluated.

Be consistent with style of writing and keep it formal, e.g. "I want to" is informal and "evaluate the validity" is formal. "I aim to" is more appropriate.

1	Random
2	Baseline (time order + textual order)

3	EventAll (event coherence-based, using all three performance boosters)
4	EventNoTC (event coherence-based, using all but topic continuity)
5	EventNoTP (event coherence-based, using all but time penalty)
6	EventNoLSA (event coherence-based, using all but dimensionality reduction)
7	EntityAll (entity coherence-based, using all three performance boosters)
8	EntityNoTC (entity coherence-based, using all but topic continuity)
9	EntityNoTP (entity coherence-based, using all but time penalty)
10	EntityNoLSA (entity coherence-based, using all but dimensionality reduction)

For each of the peer orderings, we calculate its average τ and *AC* scores (4.1.2) for a length category. We also test the statistical significance between the top scorer in each length/metric category and all the other versions in the same category, marked by * (p < .05) and ** (p < .01) on a two-tailed t-test.

	200w		400w	
	Kendall's τ	*AC*	Kendall's τ	*AC*
Random	0.014**	0.009**	-0.019**	0.004**
Baseline	0.387*	0.151*	0.259**	0.151*
EventAll	**0.429**	0.227	**0.416**	**0.235**
EventNoTC	0.391*	0.171*	0.347*	0.189*
EventNoTP	0.425	**0.230**	0.383*	0.227
EventNoLSA	0.388*	0.175*	0.363*	0.170*
EntityAll	0.405*	0.221	0.399*	0.206*
EntityNoTC	0.389*	0.160*	0.341*	0.182*
EntityNoTP	0.410	0.197*	0.377*	0.207*
EntityNoLSA	0.385*	0.170*	0.359*	0.169*

Nearly all versions of coherence-based orderings, whether entity or event, outperform the baseline that only considers time and textual order, showing that content coherence is an important guidance for human extract generation. In addition, all event versions significantly outperform their entity counterparts (e.g., EventNoTC vs. EntityNoTC), which is expected because events are high-level constructs that incorporate all of the document-level entities. Ordering on event information thus subsumes ordering on entity information and extra information introduced by event term and sentence event structure leads to better result.

💡 Label table by numbering it and giving a title.

💡 Explain where table is located if it is on a different page, i.e. "Table 2.1 on the previous page shows..."

Among the three performance boosters I use, the LSA-style dimensionality reduction and topic continuity are more useful than time penalty. For dimensionality reduction applied to event coherence ordering, its absence lowers the performance up to 27.7% in the case of 400W/AC. The success of dimensionality reduction confirms its advantage in discovering and utilizing hidden but useful information about content-bearing units (events or entities).

💡 Use clear subject, i.e. "The absence of dimensionality...lowers" is clearer than "For...., its absence".

The use of topic continuity is also profitable because the centering transition effectively captures the coherence pattern between adjacent sentences. Without it, the performance degrades by as much as 27.6% in the case of 200W/AC of EntityAll vs. EntityNoCT. My explanation is that the quality of entity coherence orderings is more sensitive to the entity-based topic continuity and this result is generally consistent with many other CT-inspired researches (e.g., Barzilay and Lapata, 2008).

What is at issue is the effect of time information. Introducing this factor does not always enhance performance and sometimes lowers it, so that the top scorer in the 200W/AC category is EventNoTP instead of EventAll. This phenomenon is also observed in earlier experiments using a slightly different set of peer orderings (Zhang et al., 2010). There are two possible accounts. First, document time often deviates from sentence time as a sentence in an early document is not necessarily about early events. Performance will be harmed if such deviation introduces much noise. Second, the time effect is proportional to the size of extract as removing it hurts longer extracts more than short extracts. Therefore chronological clues are more valuable for arranging more sentences.

✓Gives clear topic sentence

✓Highlights important results

✓Refers to key data indirectly, i.e. "top scorer"

✓Explains possible reasons for result

It is also noteworthy that the event-enriched VSM-based multi-document ordering algorithm achieves better result with long extracts than short extracts as the largest gain of the best over worst (excluding the random orderings) across all categories is over 60% (400W/Kendall's τ). Understandably, the importance of order and coherence grows with text length.

4.2.2 Human Rating

For this test, I use the same DUC 02 source document set, but extract sentences by myself. As sentence selection is not the focus of this study, we construct the extraction module on the simple but robust SumBasic (Nenkova and Vanderwende, 2005). To SumBasic we only add word position information as such information is very useful for news documents (Ouyang et al., 2010). For each of the document sets, a 400-word extract is produced. I use the same document sets because I want to explore whether the ordering algorithm is sensitive to extraction method. After the extracts are generated, a human annotator was asked to order a randomly shuffled collection of extracted sentences for each document set.

Use clear subjects, i.e. "This section uses…"

Do not mix present and past tenses when describing methods and results.

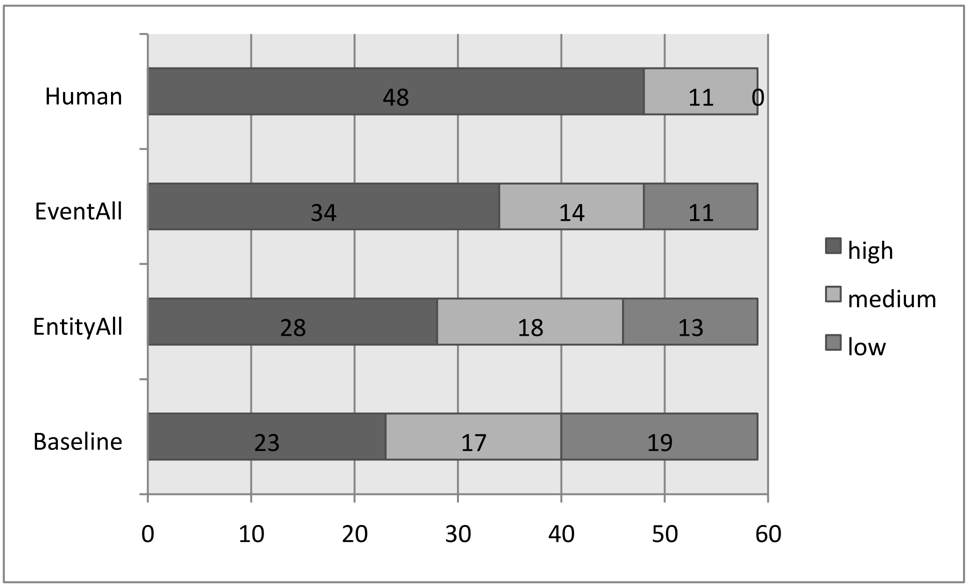
Because human rating is highly labor-intensive, we controlled the size of test sets by using 4 ordering versions for each document set: one baseline (based on time and textual order), one human ordering, one event coherence-based ordering, and one entity coherence-based ordering. The event and entity coherence-based orderings use all the three performance boosters.

Do not mix styles, i.e. "Three" and "4". Spelling numbers 1-10 is formal. Using digits is less formal.

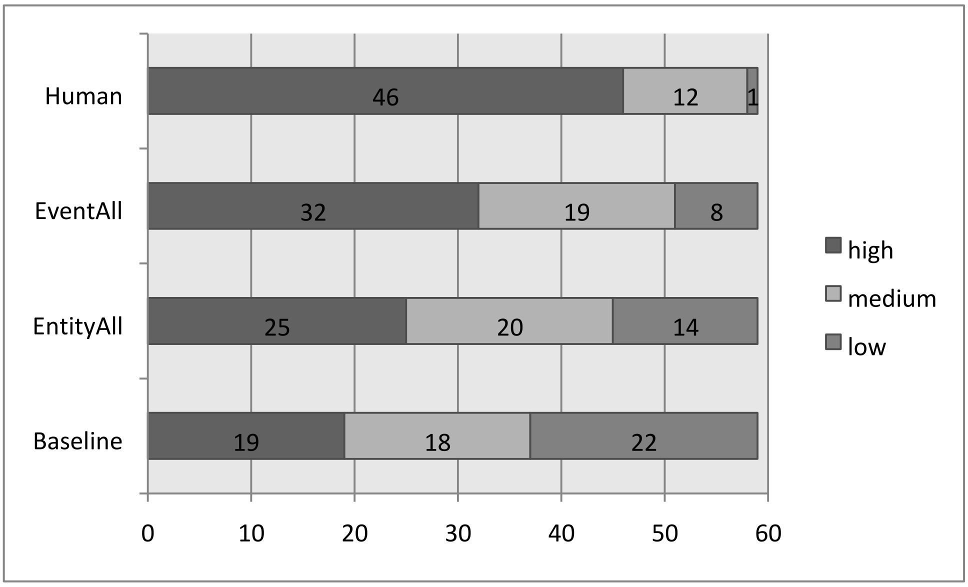
Three human judges were employed to rate the different orderings according to their degree of coherence. Each of them rated the 4 orderings for each of the 59 document sets. None of the judges is the annotator and all of them are native English speakers with teaching experience in English writing. Following (Barzilay et al., 2002) and (Bollegala et al., 2006), I instructed the judges to rank the orderings for each set as having *low*, *medium*, or *high* coherence, along a scale from being least coherent to most coherent. The orderings were randomly organized in each of the 59 groups so that the judges could not detect any pattern. The judges were also instructed to pay attention to only textual coherence and ignore any problem with spelling, punctuation, grammar, style, etc. Some coherence rating samples were provided as warm-up.

✓Gives clear topic sentence
✓Outlines methodology used giving reference to previous studies
✓Gives key figures, i.e. "59"

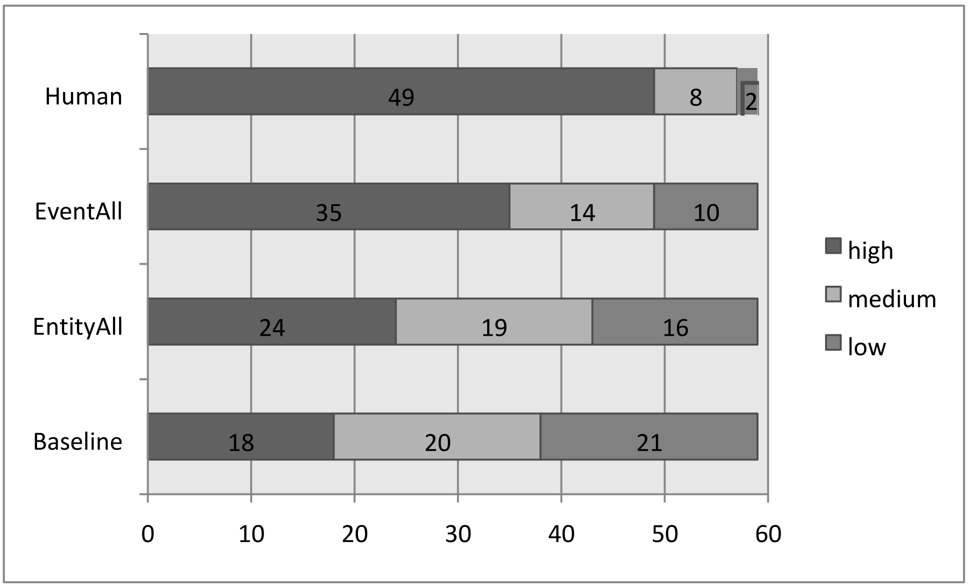
The following figures show the result of human rating by each of the judges (A, B, C) for the same set of all the orderings.



Judge A's Rating



Judge B's Rating



Judge C’s Rating

I first assess inter-judge agreement by calculating Kendall’s *W*, which ranges from 0 (indicating no agreement among the judges) to 1 (indicating total agreement among them). In our case, Kendall’s *W* = 0.893, indicating high agreement. The following table shows the aggregate rating percentages of all types of ordering.

	High	**Medium**	**Low**
Human	80.8%	17.5%	1.7%
EventAll	57.1%	26.6%	16.4%
EntityAll	43.5%	32.2%	24.3%
Baseline	33.9%	31.1%	35.0%

Overall, an obvious gap still exists between human orderings and automatic orderings, but nearly 60% of the event coherence-based orderings achieve high coherence, which is quite encouraging. By comparison, entity-based orderings produce 13% less high-coherence orderings, but 5% and 8% more medium-coherence and low-coherence orderings. The baseline achieves the lowest performance and produces more low-coherence orderings than high-coherence ones. This is clear evidence that linguistic knowledge and event semantics is useful in text ordering. The superiority of EventAll over EntityAll is also consistent with the result of the automatic evaluation. Since the extracted sentences and human-ordered extracts are different from those used in the first experiment, I claim that MDS sentence ordering is not sensitive to extraction method.

✓Reports key numbers giving approximations, e.g. “nearly 60%” or new figures, e.g. “5% more”

4.3 Recognizing Sentences with Textual Details

This task is actually a prelude of coherence-based extraction and I will report the results of two sets of experiments. In the first set, I apply different feature sets to a number of textual details to evaluate the effect of meta-phrase features and feature selection. In the second set, I implement different classification algorithms with or without extensions to evaluate their effectiveness.

Do not mix styles. Previous sections have used present and past tenses, this section uses future to outline content as well, e.g. "will".

Two datasets from the news domain are used, with all documents collected from the AQUAINT and AQUAINT-2 corpora, which have been used in Document Understanding Conference (DUC) and Text Analysis Conference (TAC) summarization tracks from 2005 to 2009. Dataset1 is about legal investigations and trials and Dataset2 is about health and safety issues. The following shows their sizes.

	#documents	#sentences	#words
Dataset1	292	7366	~140k
Dataset2	160	3879	~80k

I use 6 and 5 textual details for each dataset respectively and all the documents are manually annotated by one native speaker of English. See below for the details.

D1.1 WHO	who is a defendant or under investigation	Simple
D1.2 WHO_INV	who is investigating, prosecuting, or judging	Simple
D1.3 WHY	general reasons for the ivestigation/trial	Inferred
D1.4 CHARGES	specific charges to the defendant	Complicated
D1.5 PLEAD	defendant's reaction to charges	Complicated
D1.6 SENTENCE	sentence or other consequences to defendant	Complicated

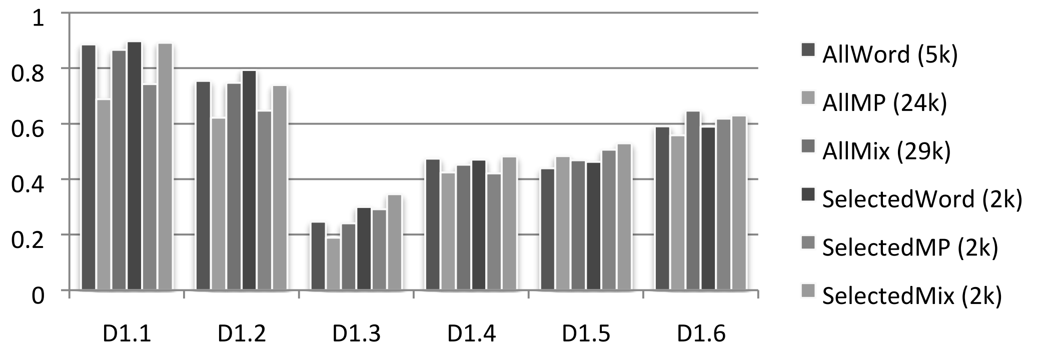
D2.1 WHAT	what is the issue	Simple
D2.2 WHO_AFFECTED	who is affected by the health/safety issue	Simple
D2.3 HOW	how they are affected	Inferred
D2.4 WHY	why the health/safety issue occurs	Inferred

D2.5 COUNTER-MEASURES	countermeasures, prevention efforts	Complicated

4.3.1 Feature Set Evaluation

For each sentence, I use the state-of-the-art Stanford Parser (Klein and Manning, 2003) to do dependency parsing and extract all meta-phrase features (3.2.1). The OpenNLP tools are used to find named entities for name-neighbor features. We do ten-fold cross validation with the SVM classifier with the linear kernel using the LIBSVM tool.

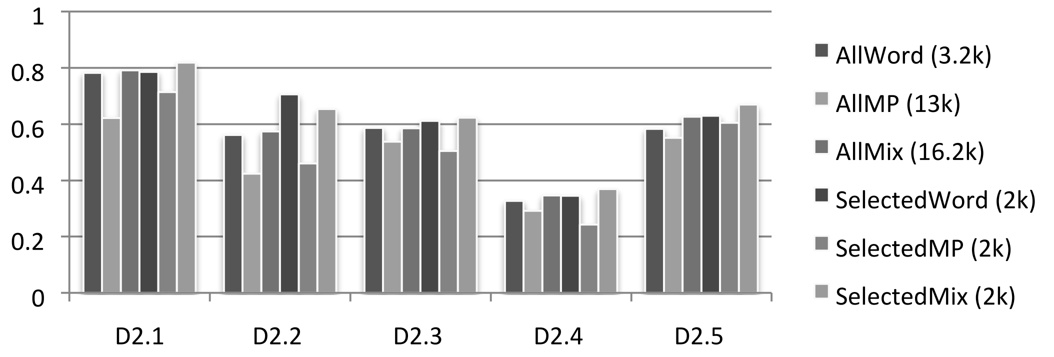
To testify the assumption that different feature sets are needed for different textual details, I evaluate on each textual detail the classification performance and time cost of 6 default feature sets – 3 full-sized feature sets: AllWord, AllMP (meta-phrase), AllMix (word + meta-phrase), and 3 selected feature sets with the top 2000 features: SelectedWord, SelectedMP, SelectedMix. The following figures show the F-measures of each of the 6 feature sets on each text detail of the two datasets.



F-measures on Dataset1

Clearly label axes.

✓Gives title to figure



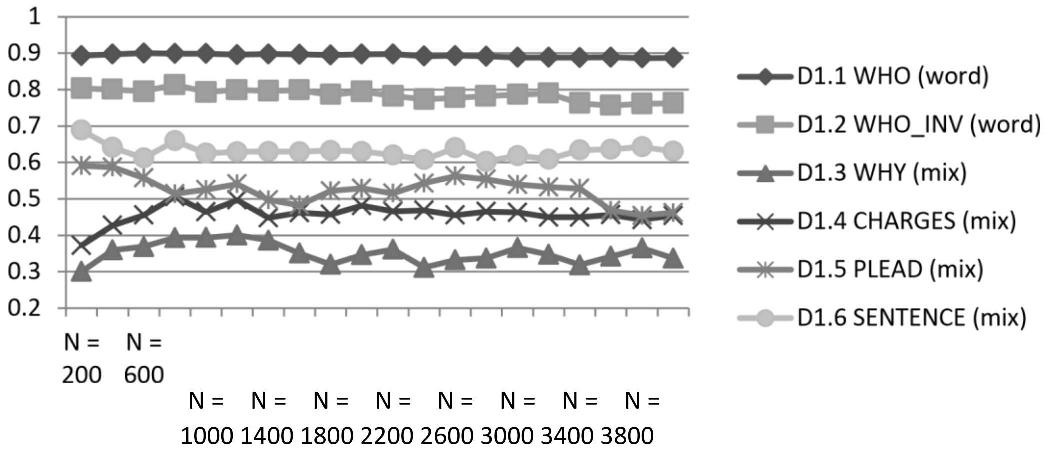
F-measures on Dataset2

On both datasets, different features perform differently across textual details. Word unigram features are generally better for simple textual details (D1.1, D1.2, D2.2), which is predictable as literal lexical information is often sufficient for them. For complicated and inferred textual details, meta-phrase features play a more important role as they either defeat word unigram features (D1.5, D1.6) by themselves or mix with words to form more predicable feature sets (D1.3–1.4, D2.1, D2.3–2.5).

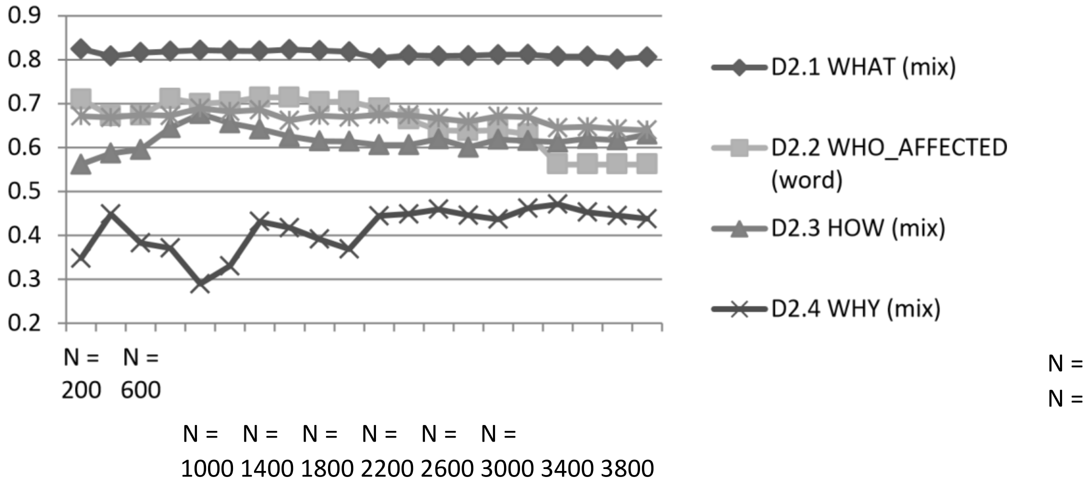
✓Explains and compares information in both figures

To explore the optimal performance of each winning feature set, I further vary feature sizes on those winning feature sets and sample 20 sizes (*N* = 200, 400, …, 4000). The F-measure results are shown in the following figures, which also indicate the best feature sets (mix/word) for each textual detail based on the previous results.

Better to explain which figure is being discussed.



F-measures with different feature set sizes on Dataset1



F-measures with different feature set sizes on Dataset2

With the only exception of D2.4, performances peak with less than 2000 features on the best feature sets. It is also obvious that simple textual details are better recognized than complicated details, which are better recognized than inferred details. Moreover, the classification of simple details is less sensitive to the variation of feature numbers than complicated or inferred details.

✓Outlines key findings

Although Dataset1 is much larger than Dataset2 more training data does not necessarily mean better performance. For the simple details D1.2 and D2.2 (both are “WHO” details and are comparable), performance seems to benefit from more training data. But for the more challenging inferred details D1.3 and D2.4 (both are “WHY” details and are comparable), better result is observed on the smaller Dataset2. I inspected the two datasets and found that apart from size, Dataset2 is composed of more diversified documents (ranging from FDA-sanctioned drugs to Chinese coal mine safety) than Dataset1 and so the “causes” are more diversified. This interesting finding suggests that for challenging textual details that go beyond the literal content, training data diversity may count more than training data size.

✓Shows importance of findings, e.g. “interesting finding”
✓Discusses key findings using tentative language, e.g. “suggests” and “may”

4.3.2 Classification Algorithm Evaluation

Since sentence recognition for details is defined as a multi-label classification problem, I first apply the binary decomposition without extension (BD) to the 6 default feature sets. For comparison, I also apply the label combination approach (LC) to the three full-sized feature sets (applying it to the selected feature sets is infeasible due to reasons explicated in 3.2.1). In addition to those tests that use the same feature sets for all textual details, I apply BD to the top-performing feature sets for individual textual details (BestSets) in order to discover to what extent feature set differentiation can improve on non-differentiation.

✓Outlines contents of section, giving the motivation

For the LC test, we use the off-the-shelf LIBSVM label combination tool. The evaluation metric is macro-average F, i.e., the average of F-measures on individual classes (textual details) and the results are shown in the following.

	LC	BD
AllWord	0.4859	0.5648
AllMP	0.4823	0.4938
AllMix	0.5230	0.5700
SelectedWord	N/A	0.5852
SelectedMP	N/A	0.5370
SelectedMix	N/A	0.6016

BestSets	N/A	**0.6500**

Macro-average F on Dataset1

	LC	BD
AllWord	0.5204	0.5677
AllMP	0.4784	0.4846
AllMix	0.5471	0.5840
SelectedWord	N/A	0.6148
SelectedMP	N/A	0.4970
SelectedMix	N/A	0.6261
BestSets	N/A	**0.6739**

Macro-average F on Dataset2

For both datasets, LC underperforms BD in all categories, which justifies the preference of BD as the basic approach. Not surprisingly, the selected mix features perform best among all 6 default features sets, proving the effectiveness of feature selection and meta-phrase features.

It might be argued that word unigram features contribute more than meta-phrase features in their mix as Selected(All)Word is close to Selected(All)Mix but Selected(All)MP is much poorer. Admittedly, word unigram features alone usually outperform meta-phrase features alone, which tend to have more noise. But in order to gauge their individual contributions in the mix set, it is fairer to calculate their percentages in the best mix feature sets. The following table shows the feature makeup in the 4 winning mix feature sets.

✓Discusses reasons for results using tentative language, e.g. "might be argued", "tend to"

Textual details	Word unigrams (%)	Meta-phrases (%)	
		Syntactico-semantic	Name-neighbor
D1.3	13.8	64.2	22.0
D1.4	20.5	60.8	18.7
D1.5	19.5	57.0	23.5
D1.6	18.5	64.0	17.5

Feature makeup in the winning mix feature sets on Dataset1

In this perspective, meta-phrase features, especially syntactico-semantic features, are more contributing than word unigram features. For more intuitive inspection, I extract the top 10 features from the mix feature set for D1.3 (WHY).

('/ARG/', ('identity', 'n'))	('CIA', ('agent', 'n'))
(('insulting', 'a'), '/ARG/')	(('sell', 'v'), ('country', 'n'))
(('molestation', 'n'), '/ARG/')	(('sell', 'v'), ('technology', 'n'))
CIA	(('charge', 'n'), '/prep_/')
(('convict', 'v'), '/prep_/')	(('indictment', 'n'), '/PRED/')

The above 'a', 'n', 'v' are POS tags, /prep_/ is prepositional modifier. Only 1 (CIA) feature is a word unigram and among the other 9 meta-phrase features, only 1 (('CIA', ('agent', 'n'))) is a name-neighbor feature. Obviously meta-phrase features are more *interpretable* than word unigram features and challenging textual details such as "WHY" rely more on syntactic and semantic relations under the surface of text.

💡 Avoid absolute terms, e.g. "obviously" especially at the start of the sentence.

The last batch of tests are targeted at the BD extensions discussed in 3.2.1. Based on prior results, I use BestSets for individual textual details and evaluate the classification performance of stacking, chain, context, and two ensembles: the ensemble of chains (E_chain) and the ensemble of stacking, (one) chain, and context (E_scc). In our experiments, we set the convergence condition for stacking and context as: Hamming Loss (see below) difference < 0.0001. The chain result is the average of 10 random chains and E_chain is based on 10 random chains.

Besides macro-average F, I use 2 other popular metrics: Hamming Loss (Schapire and Singer, 2000) and Average Accuracy (Godbole and Sarawagi, 2004). The results are shown in the following. For Hamming Loss, smaller is better.

✓Explains how table should be interpreted, e.g. "smaller is better"

	Hamming Loss	Macro-average F	Average accuracy
BD	0.0704	0.6500	0.7271
Stacking	0.0666	0.6792	0.7645
Chain	0.0665	0.6881	0.7645
Context	0.0673	0.6767	0.7641
E_chain	0.0656	**0.7130**	**0.7694**
E_scc	**0.0655**	0.7010	0.7676

BD and its extensions on Dataset1

	Hamming Loss	Macro-average F	Average accuracy
BD	0.1395	0.6739	0.5729
Stacking	0.1321	0.7032	0.6063
Chain	0.1317	0.7075	0.6019
Context	0.1334	0.7045	0.6029
E_chain	0.1311	0.7137	0.6148
E_scc	**0.1303**	**0.7143**	**0.6172**

BD and its extensions on Dataset2

The three BD extensions – stacking, chain, and context – prove to work for both datasets. But the difference among them is very slight. My explanation is that they capture different aspects of the problem (class dependencies vs. object dependencies) with different iterative strategies (stacking vs. chain) and the experimental data are not biased toward a particular aspect. This also explains why the ensemble extensions successfully combine the strengths of individual extensions and outperform individual extensions in each category. In the best scenario, the performance gain on BD is approaching 10%. The superiority of E_chain over Chain is consistent with the results of (Read et al., 2009).

The results and Discussion section often concludes with a short summary of the section or chapter.

5. Plan for Future Work

As discussed in Section 3, the project is composed of three major parts: post-extractive ordering, coherence-based extraction, and coherence-based revision. Currently, I have completed post-extractive ordering and some preliminary results (4.2) have been published in the proceedings of a top-ranking conference (Zhang et al., 2010). I am now working on the first task of coherence-based extraction – recognizing sentences with textual details – and have submitted the preliminary results (4.3) to a conference. But the task is not fully completed both in algorithm design and in experimentation, which are high on my agenda.

✓Outlines the work done

✓Highlights publications, which are often included as an appendix

✓Explains work in progress

✓Explains what work will be done

✓Provides a clear table giving dates for future work

After the sentence-level detail recognition is completed, I will migrate from text classification to summarization by first working on the details of generating globally coherence textual details and then developing sentence ranking/selection methods on top of detail sentence recognition. This will complete the second part of the project.

My next emphasis is on the third part, coherence-based revision, which for now stays on paper but has much to borrow from the previously developed coherence-oriented algorithms and a large body of literature on sentence revision/collection/reduction, etc.

Presumably news is not the only or the best domain to demonstrate the advantages of coherence concerns in automatic text summarization. Texts with more compact narrative or expository structure are expected to better illustrate the benefit of coherence to summarization. After the core algorithms for all the three parts are set up, I will experiment with data from non-news domains. Human assessment and extrinsic evaluation will also be used because coherence is ultimately an effect from human-text interaction.

The following table lists the scheduled progress to complete the project for my PhD program.

2011	Jan – Mar	Improving algorithms and doing more experiments for sentence recognition with textual details (3.2.1)
	Apr – Jun	Migrating from sentence recognition with textual details to sentence selection with globally coherent textual details (3.2.2)
	Jul – Sept	Working on global coherence-based sentence revision (3.3.1)
	Oct – Dec	Working on local coherence-based sentence revision (3.3.2)
2012	Jan – Mar	Experimenting on non-news domain with a design that incorporates all the developed parts of the projects
	Apr – Jun	Writing PhD dissertation, first draft
	Jul – Aug	Refining PhD dissertation, second draft

Acknowledgements

💡 The acknowledgements section is often at the beginning of a report after the abstract.

💡Do not mix the language style. The language can be less formal in this section, but informal expressions such as “sticking with” are best avoided. “During the process of working on my project” is a better option. Very formal language, e.g. “profusely from the erudition of”, is also not used in this section. “Knowledge” or “scholarship” are better alternatives.

During the process of sticking with my project, I have benefited profusely from the erudition of my chief supervisor, Dr. Wenjie Li. The basic framework of the project was

established by her for me, when I was transitioning from a linguistics researcher to a computational linguistics student. In the past 15 months, she never ceased to inspire and encourage me.

My colleagues and friends have been supporting me all the time. Ouyang You brushes up my math knowledge whenever I'm trapped in equations and matrices. Dehong Gao generously helps with programming and testing. Xiaoyan Cai inspires me with her rich experience with machine learning methods.

The Pao Yue-kong Library has provided all the necessary reference books, journals, and e-resources for my research. The Department of Computing has provided easy access to world-class equipment, laboratories, offices, etc. that are indispensable to the completion of this project.

References

Alonso i Alemany, L. and Fuentes Fort, M. 2003. Integrating cohesion and coherence for automatic summarization. In *Proceedings of EACL2003*, 1–8. Budapest, Hungary.

Althaus, E., Karamanis, N., and Koller A. 2004. Computing Locally Coherent Discourses. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, 399–406, Barcelona, Spain.

Aone, C., Okurowski, M. E., Gorlinsky, J., and Larsen, B. 1999. A Trainable Summarizer with Knowledge Acquired from Robust NLP Techniques. In I. Mani & M. T. Maybury (eds.), *Advances in Automatic Text Summarization*. 71–80. Cambridge, Massachusetts: MIT Press.

Baldwin, B. and Morton, T. 1998. Dynamic Coreference-Based Summarization. In *Proceedings of the Third Conference on Empirical Methods in Natural Language Processing (EMNLP-3)*, 1–6. Granada, Spain.

Barzilay, R. and Elhadad, M. 1997. Using Lexical Chains for Text Summarization. In *Proceedings of the ACL Workshop on Intelligent Scalable Text Summarization*, 10–17.

Barzilay, R., Elhadad, N., and McKeown, K. 2002. Inferring Strategies for Sentence Ordering in Multidocument News Summarization. *Journal of Artificial Intelligence Research*, 17:35–55.

Barzilay, R. and Lapata, M. 2005. Modeling Local Coherence: An Entity-based Approach. In *Proceedings of the 43rd Annual Meeting of the ACL*, 141–148. Ann Arbor.

Barzilay, R. and Lapata, M. 2008. Modeling Local Coherence: An Entity-Based Approach. *Computational Linguistics*, 34:1–34.

Barzilay, R., and Lee, L. 2003. Learning to Paraphrase: An Unsupervised Approach Using Multiple-Sequence Alignment. In *Proceedings of HLT-NAACL 2003*, 16–23.

Barzilay, R., and Lee, L. 2004. Catching the Drift: Probabilistic Content Models, with Applications to Generation and Summarization. In *HLT-NAACL 2004: Proceedings of the Main Conference*. 113–120.

Barzilay R. and McKeown, K. 2005. Sentence Fusion for Multidocument News Summarization. *Computational Linguistics*, 31(3): 297–328.

Barzilay, R., McKeown, K., and 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*, 550–557. New Brunswick, New Jersey: Association for Computational Linguistic.

Beaver, D. 2004. The Optimization of Discourse Anaphora. *Linguistics and Philosophy*, 27:3–56.

Bollegala, D, Okazaki, N., and Ishizuka, M. 2006. A Bottom-up Approach to Sentence Ordering for Multi-document Summarization. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the ACL*, 385–392, Sydney, Australia.

Brandow, R., Mitze, K., and Rau, L. F. 1995. Automatic Condensation of Electronic Publications by Sentence Selection. *Information Processing and Management*, 31(5):675–685.

Brennan, S. E., Marilyn A. F., and Carl J. P. 1987. A Centering Approach to Pronouns. In *Proceedings of ACL 1987*, 155–162. Stanford, CA.

Brew, C. 1992. Letting the Cat out of the Bag: Generation for Shake-and-Bake MT. In *COLING-92*, 610–616.

Brin, S. and Page, L. 1998. The Anatomy of a Large-scale Hypertextual Web Search Engine. *Computer Networks and ISDN Systems*, 30:1–7.

Boutell, M. R., Luo, J., Shen, X., and Brown, C. M. 2004, Learning Multi-label Scene Classification, *Pattern Recognition*, 37(9):1757–71.

Bromberg, I. 2006. Ordering Sentences According to Topicality. *Presented at the Midwest Computational Linguistics Colloquium*.

Carbonell, J. and Goldstein, J. 1998. The Use of MMR, Diversity-Based Reranking for Reordering Documents and Producing Summaries. In *SIGIR-98*, 335–336.

Clarke, J., and Lapata, M. 2007. Modelling Compression with Discourse Constraints. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, 1–11. Prague.

Cohen, W. W., Schapire, R. E., and Singer, Y. 1999. Learning to Order Things. *Journal of Artificial Intelligence Research*, 10:243–270.

Conroy, J. M., Schlesinger, J. D., and Goldstein, J. 2006. CLASSY Tasked Based Summarization: Back to Basics. In *proceedings of the Document Understanding Conference (DUC-06)*.

Cremmins, E. T. 1996. *The Art of Abstracting*. Arlington, VA: Information Resources Press.

Cristea, D., Ide, N., and Romary L. 1998. Veins Theory: A Model of Global Discourse Cohesion and Coherence. In *Proceedings of COLING/ACL'98*, 281–285. Montreal.

Daumé III, H. and Marcu, D. 2006. Bayesian Query-Focused Summarization. In *Proceedings of ACL-2006*, 305–312, Sydney, Australia.

Davies, D. and Bouldin, D. 1979. A Cluster Separation Measure. In *IEEE Transactions on Pattern Analysis and Machine Intelligence* 1:224–227.

Dejong, G. 1982. An Overview of the FRUMP System. In Lehnert, W. G. and Ringle, M. H. (eds.), *Strategies for Natural Languages Processing*. Hillsdale, NJ: Erlbaum.

Dunning, T. 1993. Accurate Methods for the Statistics of Surprise and Coincidence. *Computational Linguistics*, 19(1): 61–74.

Edmundson, H. P. 1969. New Methods in Automatic Abstracting. *Journal of the Association for Computing Machinery*, 16(2):264-285.

Elhadad, N. and McKeown, K. 2001. Towards Generating Patient Specific Summaries of Medical Articles. In *Proceedings of the NAACL-01*, 32–40.

Endres-Niggemeyer, B. 1998. *Summarizing Information*. Berlin, Germany: Springer-Verlag.

Erkan, G. and Radev, D. 2004. LexRank: Graph-based Centrality as Salience in Text Summarization. *Journal of Artificial Intelligence Research*, 22: 457–479.

Farzinder, A. and Lapalme, G. 2004. Legal Text Summarization by Exploration of the Thematic Structure and Argumentative Roles. In *Proceedings of ACL04*, 27–34.

Fellbaum, C. 1998. *WordNet: An Electronic Lexical Database*. Cambridge, Massachusetts: MIT Press.

Firmin, T. and Chrzanowski, J. 1999. An Evaluation of Automatic Text Summarization Systems. In I. Mani and M. T. Maybury (eds.), *Advances in Automatic Text Summarization*, 325–336. Cambridge, Massachusetts: MIT Press.

Filatova, E. and Hatzivassiloglou, V. 2003. Domain-Independent Detection, Extraction, and Labeling of Atomic Events. In *Proceedings of RANLP*, 145–152, Borovetz, Bulgaria.

Filippova, K. and Strobe, M. 2007. Extending the Entity-grid Coherence Model to Semantically Related Entities. In *Proceedings of the 11th European Workshop on Natural Language Generation*, 139–142. Schloss Dagstuhl, Germany.

Foltz, P. W., Kintsch, W., and Landauer, T. K. 1998. Textual Coherence Using Latent Semantic Analysis. *Discourse Processes*, 25(2, 3):285–307.

Fum, D., Guida, G., and Tasso, C. 1982. Forward and Backward Reasoning in Automatic Abstracting. In *The Proceedings of COLING82*, 83–88.

Ganesan, K., Zhai, C, and Han, J. 2010. Opinosis: A Graph-Based Approach to Abstractive Summarization of Highly Redundant Opinions. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*. 340–348. Beijing, China.

Gibbs, F. 1993. Knowledge-Based Indexing in SIMPR: Integration of Natural Language Processing and Principles of Subject Analysis in an Automated Indexing System. *Document and Text Management*, 1(2):131–153.

Godbole, S. and Sarawagi, S. 2004. Discriminative Methods for Multi-labeled Classification. In *Proceedings of the 8th Pacific-Asia Conference on Knowledge Discovery and Data Mining*, 22–30.

Gong, Y. and Liu, X. 2002. Generic Text Summarization Using Relevance Measure and Latent Semantic Analysis. In *Proceedings of ACM SIGIR-02*, 19–25.

Grefenstette, G. 1998. Producing Intelligent Telegraphic Text Reduction to Provide an Audio Scanning Service for the Blind. In *AAAI-98 Spring Symposium on Intelligent Text Summarization*, 102–108.

Grosz, B. J., Aravind K. J., and Scott W. 1995. "Centering: A framework for Modeling the Local Coherence of Discourse". *Computational Linguistics*, 21(2):203–225.

Grover, C., Hackey, B., and Korycinski, C. 2003. Summarizing Legal Texts: Sentential Tense and Argumentative Rules. In *Proceedings of the HLT-NAACL-03*, 33–40.

Hahn, U. 1990. TOPIC Parsing: Accounting for Text Macro Structures in Full-Text Analysis. *Information Processing and Management*, 26(1):135–170.

Hahn, U. and Reimer U. 1999. Knowledge-Based Text Summarization: Salience and Generalization Operators for Knowledge Base Abstraction. In I. Mani and M. T. Maybury (eds.), *Advances in Automatic Text Summarization*, 215–232. Cambridge, Massachusetts: MIT Press.

Harabagiu, S. and Lacatusu, F. 2002. Generating Single and Multi-Document Summarization with GISTexter. In *Proceedings of DUC 2002*, 30–38.

Hasler, L. 2004. An Investigation into the Use of Centering Transitions for Summarization. In *Proceedings of CLUK'04*, 100–107. Birmingham, UK.

Hirst, G. and St-Onge, D. 1998. Lexical Chains as Representations of Context for the Detection and Correction of Malapropisms. In Fellbaum, C. (ed.), *WordNet: An Electronic Lexical Database and Some of its Applications*, 305–332. Cambridge, Massachusetts: MIT Press.

Hobbs, J. 1985. On the Coherence and Structure of Discourse. *Report No. CSLI-85-37*. Stanford, California: Center for the Study of Language and Information, Stanford University.

Hovy, E. 1988a. *Generating Natural Language under Pragmatic Constraints*. Hillsdale, NJ: Erlbaum.

Hovy, E. 1988b. Planning Coherent Multisentential Text. *Technical Report ISI/RS-88-208*. Marina del Rey, California: Information Sciences Institute.

Hovy, E. 2005. Automated Text Summarization. In R. Mitkov (ed.), *The Oxford Handbook of Computational Linguistics*, pp. 583–598. Oxford: Oxford University Press.

Hovy, E. and Lin, C-Y. 1999. Automated Text Summarization in SUMMARIST. In I. Mani and M. T. Maybury (eds.), *Advances in Automatic Text Summarization*. 81–94. Cambridge, Massachusetts: MIT Press.

Hovy, E., Lin, C-Y., and Zhou, L. 2005. Evaluating DUC 2005 Using Basic Elements. In *Proceedings of the Fifth Document Understanding Conference (DUC)*, Vancouver, Canada.

Hovy, E. and Maier, E. 1995. Parsimonious or Profligate: How Many and Which Discourse Relations? *Unpublished manuscript*. Marina del Rey, CA: Information Sciences Institute of the University of Southern California.

Jacobs, P. S. and Rau, L. F. 1990. SCISOR: Extracting Information from Online News. *Communications of the ACM*, 33(11):88–97.

Ji, P. D. and Pulman S. 2006. Sentence ordering with manifold-based classification in multidocument summarization. In *Proceedings of EMNLP 2006*, 526–533.

Jiang, J. and Conrath, D. 1997. Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy. In *Proceedings of ROCLING*, 19–33.

Jing, H. 2000. Sentence Reduction for Automatic Text Summarization. In *Proceedings of the 6th Applied Natural Language Processing Conference*, 310–315.

Jing, H. and McKeown, K. 1999. The Decomposition of Human-Written Summary Sentences. In *Proceedings of the 22nd International Conference on Research and Development in Information Retrieval (SIGIR99)*, 129–136.

Jing, H. and McKeown, K. 2000. Cut and Paste Based Text Summarization. In *Proceedings of the 1st North American Chapter of the Association for Computational Linguistics*, 178–185.

Jurafsky D. and Martin, J. H. 2009. *Speech and Language Processing, Second Edition*. Upper Saddle River, NJ: Pearson Education International.

Karamanis, N., 2001. Exploring Entity-based Coherence. In *Proceedings of CLUK4*, 18–26.

Karamanis, N. 2006. Entity versus Rhetorical Coherence for Information Ordering: Initial Experimentation. In *Proceedings of ESSLLI Workshop on Coherence for Generation and Dialogue*, 25–31. Malaga, Spain.

Karamanis, N. 2007. Supplementing Entity Coherence with Local Rhetorical Relations for Information Ordering. *Journal of Logic, Language and Information*, 16:445–464.

Karamanis, N. and Mellish C. 2005. Using a Corpus of Sentence Orderings Defined by Many Experts to Evaluate Metrics of Coherence for Text Structuring. *In Proceedings of ENLG05*, 174–179.

Karamanis, N., Mellish, C., Oberlander, J., and Poesio M. 2004a. A Corpus-Based Methodology for Evaluating Metrics of Coherence for Text Structuring. In A. Belz et al. (eds.), *INLG 2004, LNAI 3123*. 90–99. Heidelberg, Berlin: Springer-Verlag.

Karamanis, N., Poesio M., Mellish C., and Oberlander, J. 2004b. Evaluating Centering-based Metrics of Coherence for Text Structuring Using a Reliably Annotated Corpus. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, 391–398, Barcelona, Spain.

Karamanis, N., Poesio, M. Mellish, C., and Oberlander, J. 2008. Evaluating Centering for Information Ordering Using Corpora. *Computational Linguistics*, 35:29–46.

Kehler, A. 2002. *Coherence, Reference, and the Theory of Grammar*. Stanford, California: CSLI Publications.

Kibble, R. and Power, R. 2004. Optimizing Referential Coherence in Text Generation. *Computational Linguistics*, 30:401–416.

Klein, D. and Manning, C. D. 2003. Accurate Unlexicalized Parsing. In *Proceedings of the 41st Meeting of the Association for Computational Linguistics*, 423–430.

Kleinberg, J. M. 1999. Authoritative Sources in a Hyperlinked Environment. *Journal of the ACM*, 46(5):604–632.

Knight, K. 1999. Decoding Complexity in Word-Replacement Translation Models. *Computational Linguistics*, 25(4):607–615.

Knight, K. and Marcu, D. 2000. Statistics-Based Summarization — Step One: Sentence Compression. In *Proceedings of AAAI*, 703–710, Austin, Texas.

Knott, A., Oberlander J., O'Donnell M., and Mellish C. 2001. Beyond Elaboration: The Interaction of Relations and Focus in Coherent Text. In Sanders, T., Schilperoord, J. and Spooren, W. (eds.), *Text Representation: Linguistic and Psycholinguistic Aspects*, 181–196. Benjamins.

Kupiec, J., Pedersen, J., and Chen, F. 1995. A Trainable Document Summarizer. In *Proceedings of the 18th International Conference on Research and Development in Information Retrieval (SIGIR'95)*, 68–73. Seattle, Washington: Association for Computing Machinery.

Lacatusu, V. F., Parker, P., and Harabagiu, S. M. 2003. "Lite-GISTexter: Generating Short Summaries with Minimal Resources". In *Proceedings of DUC 2003*, 122– 128.

Landauer, T. and Dumais, S. 1997. A Solution to Plato's Problem: The Latent Semantic Analysis Theory of the Acquisition, Induction, and Representation of Knowledge. *Psychological Review*, 104:211–240.

Lapata, M. 2003. Probabilistic Text Structuring: Experiments with Sentence Ordering. In *Proceedings of the Annual Meeting of ACL*, 545–552. Sapporo, Japan.

Lapata, M. 2006. Automatic evaluation of information ordering: Kendall's tau. *Computational Linguistics*, 32(4):1–14.

Lapata, M. and Barzilay, R. 2005. Automatic Evaluation of Text Coherence: Models and Representations. In *Proceedings of the 19th International Joint Conference on Artificial Intelligence*. 1085–1090. Edinburgh.

Lehnert, W. G. 1999. Plot Units: A Narrative Summarization Strategy. In I. Mani and M. T. Maybury (eds.) *Advances in Automatic Text Summarization*, 177–214. Cambridge, Massachusetts: MIT Press.

Lesk, M. 1986. Automatic Sense Disambiguation: How to Tell a Pine Cone from an Ice Cream Cone. In *Proceedings of the 1986 Special Interest Group in Documentation*, 24–26.

Li, W., Wu, M., Lu, Q., Xu, W., and Yuan, C. 2006. Extractive Summarization Using Inter- and Intra- Event Relevance. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the ACL*, 369–376. Sydney.

Lin, C-Y. 2004. ROUGE: A Package for Automatic Evaluation of Summaries. In *ACL 2004 Workshop on Text Summarization Branches out , post-conference workshop of ACL 2004*, 74–81.

Lin, C-Y. and Hovy, E. 1997. Identifying Topics by Position. In *Proceedings of the Applied Natural Language Processing Conference*, 283–290, Washington, DC

Lin, C-Y. and Hovy, E. 2000. The Automated Acquisition of Topic Signatures for Text Summarization. In *COLING-00*, 495–501.

Lin, C-Y. and Hovy, E. 2002. Automated Multi-document Summarization in NeATS. In *Proceedings of the Human Technology Conference 2002*, 50–53.

Lin, C-Y. and Hovy, E. 2003. Automatic Evaluation of Summaries Using N-gram Co-Occurrence Statistics. In *Proceedings of the Human Technology Conference 2003 (HLT-NAACL-2003)*, 71–78, Edmonton, Canada.

Luhn, H. P. 1958. The Automatic Creation of Literature Abstracts. *IBM Journal of Research and Development*, 2(2):159–165.

Madnani, N., Zajic, D., Dorr, B., Ayan, N. F., and Lin, J. 2007. Multiple Alternative Sentence Compressions for Automatic Text Summarization. In *Proceedings of the HLT/NAACL Document Understanding Conference Workshop*, Rochester, New York.

Manabu, O. and Hajime M. 2000. Query-biased Summarization Based on Lexical Chaining. *Computational Intelligence*, 16(4):578–585.

Mani, I. 2001. *Automatic Summarization*. Amsterdam/Philadelphia: John Benjamins.

Mani, I. and Bloedorn E. 1999. Summarizing Similarities and Differences among Related Documents. *Information Retrieval*, 1:35–67.

Mani, I., Firmin, T., House, D., Klein, G., Sundheim, B., and Hirschman, L. 2002. The TIPSTER SUMMAC text summarization evaluation. *Natural Language Engineering*, 8(1):35–67.

Mani, I., Gates, B., and Bloedorn, E. 1999. Improving Summaries by Revising Them. In *Proceedings of ACL99*, 558–565, College Park, Maryland.

Mani, I. and Maybury M. 1999. *Advances in Automatic Text Summarization*. Cambridge, Massachusetts: MIT Press.

Mann, W. C. and Thompson, S. 1988. Rhetorical Structure Theory: Toward a Functional Theory of Text Organization. *Text*, 8(3):243–281.

Marcu, D. 1997. The Rhetorical Parsing of Natural Language Texts. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics (ACL-97)*, 96–103.

Marcu, D. 1999. Discourse Trees Are Good Indicators of Importance in Text. In I. Mani and M. T. Maybury (eds.), *Advances in Automatic Text Summarization*. 123–136. Cambridge, Massachusetts: MIT Press.

Marcu, D. 2000. *The Theory and Practice of Discourse Parsing and Summarization*. Cambridge, MA: The MIT Press.

Maybury, M. 1999. Generating Summaries from Event Data. In I. Mani and M. T. Maybury (eds.), *Advances in Automatic Text Summarization*. 265–281. Cambridge, Massachusetts: MIT Press.

McDonald, R. 2006. Discriminative Sentence Compression with Soft Syntactic Constraints. In *EACL-06*, 297–304.

McKeown, K., Barzilay, R., Evans, D., Hatzivassiloglou, V., Klavans, J. L., Nenkova, A., Sable, C., Schiffman, B., and Sigelman, S. 2002. Tracking and Summarizing News on a Daily Basis with Columbia's Newsblaster. In *Proceedings of HLT 2002*, 280–285, San Diego, California.

McKeown, K., Jordan, D., and Hatzivassiloglou, V. 1998. Generating Patient-Specific Summaries of Online Literature. In *Proceedings of the AAAI-98*, 34–43.

McKeown, K. and Radev, D. R. 1995. Generating Summaries of Multiple News Articles. In *Proceedings, 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 74–82.

McKeown, K., Robin J., and Kukich, K. 1995. Generating Concise Natural language Summaries. *Information Processing and Management*, 31:703–733.

Mihalcea, R. 2004. Graph-based Ranking Algorithms for Sentence Extraction, Applied to Text Summarization. In *The Companion Volume to the Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics*, 170–173.

Mihalcea, R. 2006. Random Walks on Text Structures. In *CICLing 2006, LNCS 3878*, 249–262.

Mihalcea, R. and Tarau, P. 2004. TextRank: Bringing Order into Texts. In *Proceedings of EMNLP 2004*, 404–411, Barcelona, Spain.

Moens, M-F. and Dumortier, J. 2000. Use of a Text Grammar for Generating Highlight Abstracts of Magazine Articles. *Journal of Documentation*, 56:520–539.

Morris, A. H., Kasper, G. M., and Adams, D. A. 1992. The Effects and Limitations of Automated Text Condensing on Reading Comprehension Performance. *Information Systems Research*, 3(1):17–35.

Morris, J. and Hirst, G. 1991. Lexical Cohesion Computed by Thesaural Relations as an Indicator of the Structure of Text. *Computational Linguistics*, 17(1):21–48.

Myaeng, S. H. and Jang, D-H. 1999. Development and Evaluation of a Statistically-Based Document Summarization System. In I. Mani and M. T. Maybury (eds.) *Advances in Automatic Text Summarization*, 61–70. Cambridge, Massachusetts: MIT Press.

Nahnsen, T. 2009. Domain-Independent Shallow Sentence Ordering. In *Proceedings of the NAACL HLT Student Research Workshop and Doctoral Consortium*, 78–83. Boulder, Colorado.

Nenkova, A. 2008. Entity-driven Rewrite for Multi-Document Summarization. In *The Third International Joint Conference on Natural Language Processing (IJCNLP08)*, 118–125, Hyderabad, India.

Nenkova, A. and Vanderwende, L. 2005. The Impact of Frequency on Summarization. *Technical Report MSR-TR-2005-101*, Microsoft Research, Redmond, WA.

Okazaki, N., Matsuo, Y., and Ishizuka, M. 2004. Improving Chronological Ordering by Precedence Relation. In *Proceedings of 20th International Conference on Computational Linguistics (COLING 04)*, 750–756.

Okazaki, N., Matsuo, Y., Matsumura, N., and Ishizuka, M. 2003. Sentence Extraction by Spreading Activation with Refined Similarity Measure. In *Proceedings of FLAIRS Conference*, 407–411.

Orasan, C. 2003. An evolutionary Approach for Improving the Quality of Automatic Summaries. In *Proceedings of the Multilingual Summarization and Question Answering — Machine Learning and Beyond Workshop*, 37–45. Sapporo, Japan.

Ouyang, Y., Li, W., Lu, Q., and Zhang, R. 2010. A Study on Position Information in Document Summarization. In *COLING 2010*: Poster Volume, 919–927, Beijing.

Over, P. and Yen, J. 2003. An Introduction to DUC 2003. *Presented at the 2003 Human Language Technology Conference*.

Paice, C. D. and Jones, P. A. 1993. The Identification of Important Concepts in Highly Structured Technical Papers. In *Proceedings of the 16th International Conference on Research and Development in Information Retrieval (SIGIR93)*, 69–78.

Passonneau, R. J., Nenkova, A., McKeown, K., and Sigelman, S. 2005. Applying the Pyramid Method in DUC 2005. In *Proceedings of the Fifth Document Understanding Conference (DUC)*, Vancouver, Canada.

Poesio, M., Stevenson, R., Di Eugenio B., and Hitzeman, J. 2004. Centering: A Parametric Theory and Its Instantiations. *Computational Linguistics*, 30:309–363.

Polanyi, L., Culy, C., van den Berg, M., Thione, G. L., and Ahn, D. 2004. A Rule-Based Approach to Discourse Parsing. In *Proceedings of the Fifth SIGdial Workshop on Discourse and Dialogue, ACL*, 108–117.

Pollock, J. J. and Zamora, A. 1975. Automatic Abstracting Research at Chemical Abstracts Service. *Journal of Chemical Information and Computer Sciences*, 15(4):226–232.

Radev, D., Jing, H., and Budzikowska, M. 2000. Centroid-Based Summarization of Multiple Documents: Sentence Extraction, Utility-Based Evaluation, and User Studies. In *Proceedings of the ANLP/NAACL-00*, 21–30.

Radev, D., Allison, T., Blair-Goldensohn, S., Blitzer, J., Çelebi, A., Dimitrov, S., Drabek, E., Hakim, A., Lam, W., Liu, D., Otterbacher, J., Qi, H., Saggion, H., Teufel, S., Topper, M., Winkel, A., and Zhang, Z. 2004a. MEAD — A Platform for Multidocument Multilingual Text Summarization. In *Proceedings of LREC 2004*, 55–57.

Radev, D., Jing, H., Styś, M., and Tam, D. 2004b. Centroid-Based Summarization of Multiple Documents. *Information Processing and Management*, 40: 919–938.

Rath, G. J., Resnick, A., and Savage, T. R. 1961. The formation of abstracts by the selection of sentences. *American Documentation*, 12(2):139–143.

Read, J., Pfahringer, B., and Holmes, G. 2008. Multi-label Classification Using Ensembles of Pruned Sets. In *ICDM'08: Eighth IEEE International Conference on Data Mining*, 995–1000.

Read, J., Pfahringer, B., Holmes, G., and Frank, E. 2009. Classifier Chains for Multi-Label Classification. In *ECML/PKDD 2009*, 254–269.

Riezler, S., King, T. H., Crouch, R., and Zaenen, A. 2003. Statistical Sentence Condensation Using Ambiguity Packing and Stochastic Disambiguation Methods for Lexical-Functional Grammar. In *HLT-NAACL-03*, 118–125. Edmonton, Canada. Rush, J. E., Salvador, R., and Zamora, A. 1971. Automatic Abstracting and Indexing II: Production of Indicative Abstracts by Application of Contextual Inference and Syntactic Coherence Criteria. *American Society for Information Science*, 22(4):260–274.

Saggion, H. and Lapalme, G. 2002. Generating Indicative-Informative Summaries with SumUM. *Computational Linguistics*, 28(4):497–526.

Salton, G., Singhal, A., Mitra M., and Buckley, C. 1997. Automatic Text Structuring and Summarization. *Information Processing and Management*, 33(2):193–207.

Schapire, R. E. and Singer, Y. 2000, Boostexter: a Boosting-Based System for Text Categorization, *Machine Learning*, 39(2/3):135–68.

Silber, H. G. and McCoy, K. F. 2000. Efficient Text Summarization Using Lexical Chains. In *Proceedings of the 5th International Conference on Intelligent User Interfaces*, 252–255. New Orleans, Louisiana.

Skorohodko, E. F. 1971. Adaptive Method of Automatic Abstracting and Indexing. In *Proceedings of IFIP Conference, Lubljana, booklet TA-6*, 133–137.

Soricut, R. and Marcu D. 2006. Discourse Generation Using Utility-Trained Coherence Models. In *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, 803–810.

Spärck Jones, K. 1972. A Statistical Interpretation of Term Specificity and Its Application in Retrieval. *Journal of Documentation*, 28(1):11–21.

Spärck Jones, K. 1999. Automatic Summarizing: Factors and Directions. In I. Mani and M. T. Maybury (eds.), *Advances in Automatic Text Summarization*. 1–12. Cambridge, Massachusetts: MIT Press.

Spärck Jones, K. 2007. Automatic Summarising: The State of the Art. *Information Processing and Management*, 43:1449–1481.

Steinberger, J. and Křišťan, M. 2007. LSA-Based Multi-Document Summarization. In *Proceedings of 8th International Workshop on Systems and Control*, 87–91.

Stevenson, M. and Greenwood, M. A. 2005. A Semantic Approach to IE Pattern Recognition. In *Proceedings of the 43rd Annual Meeting of the ACL*, 379–386.

Strzalkowski, T., Stein, G., Wang, J., and Wise, B. 1999. A Robust Practical Text Summarizer. In I. Mani and M. T. Maybury (eds.), *Advances in Automatic Text Summarization*. 137–154. Cambridge, Massachusetts: MIT Press.

Tapiero, I. 2007. *Situation Models and Levels of Coherence: Towards a Definition of Comprehension*. Mahwah, New Jersey: Lawrence Erlbaum Associates.

Teufel, S. and Moens, M. 1999. Argumentative Classification of Extracted Sentences as a First Step towards Flexible Abstracting. In I. Mani and M. T. Maybury (eds.), *Advances in Automatic Text Summarization*. 155-171. Cambridge, Massachusetts: MIT Press.

Tsoumakas, G. and Katakis, I. 2007. Multi label classification: An overview. *International Journal of Data Warehousing and Mining*, 3(3):1–13.

Tsoumakas, G. and Vlahavas, I. P. 2007. Random K-labelsets: An Ensemble Method for Multilabel Classification. In *ECML '07: 18th European Conference on Machine Learning*, 406–17.

Wan, X. and Yang, J. 2008. Multi-Document Summarization Using Cluster-Based Link Analysis. In *Proceedings of SIGIR-08*, 299–306

Wei, F., Li, W., Qin, L., and He, Y. 2009. A Document-sensitive Graph Model for Multi-document Summarization. In *Knowledge and Information Systems*, 22(2):245–259.

White, M. and Cardie, C. 2002. Selecting Sentences for Multi-Document Summaries with Randomized Local Search. In *Proceedings of ACL02*, 9–18.

Wilpon, J. and Rabiner, L. 1985. A Modified K-means Clustering Algorithm for Use in Isolated Word Recognition. In *Proceedings of IEEE Trans. Acoustics, Speech, Signal, ASSP-33*, 587–594.

Wolf, F. and Gibson, E. 2004. Paragraph-, Word-, and Coherence-based approaches to Sentence Ranking: A Comparison of Algorithm and Human Performance. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, 383–390. Barcelona, Spain.

Wolf, F. and Gibson, E. 2006. *Coherence in Natural Language*. Cambridge, MA: MIT Press.

Wolpert, D. H. 1992. Stacked Generalization, *Neural Networks*, 5: 241-59.

Zhang R., Li, W., and Lu, Q. 2010. Sentence Ordering with Event-Enriched Semantics and Two-Layered Clustering for Multi-Document News Summarization. In *COLING 2010: Poster Volume*, 1489–1497, Beijing.