

<http://researchcommons.waikato.ac.nz/>

Research Commons at the University of Waikato

Copyright Statement:

The digital copy of this thesis is protected by the Copyright Act 1994 (New Zealand).

The thesis may be consulted by you, provided you comply with the provisions of the Act and the following conditions of use:

- Any use you make of these documents or images must be for research or private study purposes only, and you may not make them available to any other person.
- Authors control the copyright of their thesis. You will recognise the author's right to be identified as the author of the thesis, and due acknowledgement will be made to the author where appropriate.
- You will obtain the author's permission before publishing any material from the thesis.

Department of Computer Science



Hamilton, New Zealand

Concept-based text clustering

by

Lan Huang

This thesis is submitted in partial fulfillment of the requirements
for the degree of
Doctor of Philosophy in Computer Science
at The University of Waikato

April 2011

© 2011 Lan Huang

Abstract

Thematic organization of text is a natural practice of humans and a crucial task for today’s vast repositories. Clustering automates this by assessing the similarity between texts and organizing them accordingly, grouping like ones together and separating those with different topics. Clusters provide a comprehensive logical structure that facilitates exploration, search and interpretation of current texts, as well as organization of future ones.

Automatic clustering is usually based on words. Text is represented by the words it mentions, and thematic similarity is based on the proportion of words that texts have in common. The resulting *bag-of-words* model is semantically ambiguous and undesirably orthogonal—it ignores the connections between words.

This thesis claims that using concepts as the basis of clustering can significantly improve effectiveness. Concepts are defined as *units of knowledge*. When organized according to the relations among them, they form a *concept system*. Two concept systems are used here: WordNet, which focuses on word knowledge, and Wikipedia, which encompasses world knowledge.

We investigate a clustering procedure with three components: using concepts to represent text; taking the semantic relations among them into account during clustering; and learning a text similarity measure from concepts and their relations. First, we demonstrate that concepts provide a succinct and informative representation of the themes in text, exemplifying this with the two concept systems. Second, we define methods for utilizing concept relations to enhance clustering by making the representation models more discriminative and extending thematic similarity beyond surface overlap. Third, we present a similarity measure based on concepts and their relations that is learned from a small number of examples, and show that it both predicts similarity consistently with human judgement and improves clustering. The thesis provides strong support for the use of concept-based representations instead of the classic bag-of-words model.

Acknowledgement

First, I would like to thank my chief supervisor, Ian Witten. Thank you, Ian, for taking me as a PhD student when I first applied with little research experience, and teaching me since then how to choose good research topics, how to do research, and how to write and present it. Thank you for all your guidance, encouragements and patience: I would never have been able to complete this thesis without your continuous support. Thank you and Pam for the lovely dinners at your place, wonderful sailing lessons and music concerts. I am grateful to have you as such a great mentor and example, both in career and in life.

I am also grateful to my other supervisor Eibe Frank. Thank you, Eibe, for always being supportive, patient, and willing to discuss my research. I appreciate all your insightful comments and advice on machine learning aspects. Thank you for teaching me the technical skills on using Weka and how to run experiments more efficiently.

I am fortunate and honoured to have two truly wonderful supervisors. As a result, I am lucky to have been working with a group of really bright and dynamic fellow PhD students, who have helped and supported me during my PhD. My special thanks goes to David Milne and Olena Medelyan, whose enthusiastic and creative work on exploring Wikipedia for knowledge inspired this research. Especially, thank you, David, for developing Wikipedia Miner and making it so easy to use. I warmly thank everyone who has been in this group in the past few years, for listening to my practice presentations and giving me feedback, even when they are unrelated to your research topics and not interesting.

I have learned a lot about academic writing from many people during my PhD. I would like to thank them for their comments and advice on my previous writings and this thesis: Antti Puurula, Carl Hadley, Craig Schock, David Milne, Kathryn Hempstalk, Michael Walmsley, Olena Medelyan, Rob Akscyn, Shaoqun Wu, Tina Chen, Veronica Liesaputra and of course Ian and Eibe! I especially thank Veronica for sharing so many useful tips on Latex, which helps to make writing this thesis a lot easier; and Shaoqun and Xiaofeng for their FLAX web collocation collection, which I frequently consulted for correcting and enriching

my non-native English writing.

I would like to thank the University of Waikato for the generous Doctoral Scholarship that funds this research, and BuildIT for funding my travels to the international conferences. I would also like to thank the European Media Laboratory, especially Michael Strube, for the opportunity of a two-month internship, which was a meaningful and nice break from my PhD.

I am profoundly grateful to my supportive and loving family. To my husband, Jun, thank you for your faith in me, your patience and encouragements, and being such a great partner. Mom and Dad, I am blessed to be your child. Thank you for encouraging me to pursue my interests and your always unconditional love and support.

Table of contents

Abstract	iii
Acknowledgement	v
List of Tables	xi
List of Figures	xiii
1 Introduction	1
1.1 Motivation	3
1.2 Thesis statement	5
1.3 Research questions	8
1.4 Contributions	12
1.5 Thesis structure	14
2 Background	17
2.1 Text representation	18
2.1.1 Phrases and linguistic features	19
2.1.2 Term clusters and combinations	22
2.1.3 Concepts	26
2.1.4 Utilizing different features	34
2.2 Similarity measures	36
2.3 Clustering techniques	37
2.3.1 Hierarchical algorithms	38
2.3.2 Partitional algorithms	40
2.4 Evaluation measures	41
2.4.1 Cluster purity	42
2.4.2 Structural quality	43
2.4.3 Pairwise relations	44
2.5 Summary	44
3 Experimental method	47
3.1 Experimental datasets	47
3.2 Clustering methods	49

3.3	Evaluation methodology	50
4	Concept-based document representation	53
4.1	Identifying WordNet concepts from text	54
4.1.1	Identifying candidate concepts	54
4.1.2	Sense disambiguation	55
4.2	Identifying Wikipedia concepts from text	59
4.2.1	Identifying candidate concepts	59
4.2.2	Sense disambiguation: unsupervised vs. supervised	62
4.2.3	Milne and Witten’s sense disambiguation algorithm	63
4.3	Experimental design	65
4.4	Experimental results	67
4.4.1	An example of representations	67
4.4.2	Comparison of dimensionalities	70
4.4.3	Evaluating against category structure	71
4.4.4	Evaluating with different numbers of clusters	76
4.4.5	Evaluating the clustering algorithms	83
4.5	Combining different representations	84
4.6	Summary	86
5	Semantically enriched clustering	89
5.1	Semantic concept relatedness and clustering	90
5.2	Measures for semantic concept relatedness	93
5.3	Concept reweighting based on centrality	94
5.3.1	Context centrality	94
5.3.2	Local centrality	96
5.3.3	Relative centrality	98
5.3.4	Discussion	99
5.4	Beyond surface similarity	101
5.4.1	Existing methods	101
5.4.2	Katoa’s semantically enriched document similarity	102
5.5	Experimental design	104
5.6	Experimental results	105

5.6.1	Overall effectiveness	106
5.6.2	Effectiveness with different clustering algorithms	108
5.6.3	Effectiveness of the enriched document similarity	110
5.6.4	Binary and weighted schemes	111
5.7	Summary	113
6	Learning document similarity	115
6.1	Manually assigned document similarity	117
6.2	Features	118
6.2.1	Overall similarity	119
6.2.2	Context centrality	121
6.2.3	Strongest connection	122
6.2.4	Concept groups	122
6.3	Learning algorithms	127
6.4	Evaluation against human judgement	130
6.4.1	Baselines	130
6.4.2	Experimental design	131
6.4.3	Effectiveness of individual features	132
6.4.4	Effectiveness of combinations of features	136
6.5	Evaluation with text clustering	139
6.5.1	Experimental design	139
6.5.2	Experimental results	140
6.6	Discussion	143
6.7	Summary	144
7	Conclusion and future work	147
7.1	Revisiting the thesis hypothesis	148
7.1.1	Representing texts by concepts	149
7.1.2	Utilizing relations among concepts	150
7.1.3	Learning document similarity with concepts	154
7.1.4	Summary	156
7.2	Answering the research questions	157
7.3	Future work	159

7.4	Closing remarks	160
	References	163
A	Example documents and representations	181
B	Katoa: A toolkit for concept-based text clustering	185
B.1	Third party software	185
B.2	Concept-based representation creators	187
B.3	Similarity measures	190
B.4	Clustering algorithms	192
B.5	The Katoa toolkit	192
C	Category distribution of the experimental datasets	195
D	Results of different disambiguation techniques on WordNet	199
E	Configuration of the regression algorithms	201

List of Tables

3.1	Statistics of the four experimental datasets	48
4.1	Example of features in different representations	68
4.2	Wikipedia concepts identified by other approaches	70
4.3	Dimensionality of different representations	71
4.4	Performance of different representations with the k -means clustering algorithm	72
4.5	Performance of other approaches on the SmallReuters dataset with the k -means clustering algorithm	75
4.6	Relative performance of different clustering algorithms across datasets	83
4.7	Performance of different hybrid representations with the k -means clustering algorithm	85
5.1	Example of local context centrality	97
5.2	Example of relative context centrality	97
5.3	Overall performance of semantically enriched clustering methods .	106
5.4	Performance of semantically enriched clustering with the k -means clustering algorithm	109
5.5	Performance of semantically enriched clustering with hierarchical agglomerative clustering with group-average link	110
5.6	Relative performance of the binary and the weighted schemes of the semantically enriched clustering methods	112
6.1	Features used for learning document similarity	119
6.2	Relative performance of different regression algorithms	128
6.3	Performance (consistency with human judgement) of other approaches on the HE50 dataset	130
6.4	Predictive value of features generated with Wikipedia concepts . .	133
6.5	Predictive value of features generated with WordNet concepts . .	134

6.6	Performance (consistency with human judgement) of the learned measure on the HE50 dataset	137
6.7	Performance (normalized mutual information) of the learned measure in clustering the four experimental datasets	140
A.1	Document from SmallReuters dataset's <i>oil</i> category	182
A.2	Document from Med100 dataset's <i>Nervous System Diseases</i> category	183
A.3	Document from NewsSim3 dataset's <i>comp.windows.x</i> category . .	184
A.4	Document from NewsDiff3 dataset's <i>sci.space</i> category	184
C.1	Category distribution of the NewsSim3 and NewsDiff3 datasets . .	196
C.2	Category distribution of the SmallReuters and Med100 datasets .	196
C.3	Category distribution of the original OHSUMed datasets	197
D.1	Performance of the most-common-sense-rule and context-based disambiguation strategies for WordNet	200

List of Figures

1.1	Framework for a concept system	4
2.1	A clustered search results of the query <i>clustering</i>	21
2.2	The conceptual model of a document in the Reuters collection . .	26
2.3	Encyclopedia entries for the concept <i>computer</i>	29
2.4	Lexical entries for the concept <i>computer</i>	31
4.1	Fragment of WordNet’s concept taxonomy	57
4.2	Distribution of the keyphraseness of Wikipedia anchor phrases . .	61
4.3	Discriminative power of concepts in the NewsSim3 and NewsDiff3 datasets	73
4.4	Performance of the <i>k</i> -means clustering algorithm across datasets .	77
4.5	Performance of hierarchical agglomerative clustering with group- average-link across datasets	78
4.6	Performance of hierarchical agglomerative clustering with average- link across datasets	78
4.7	Performance of hierarchical agglomerative clustering with complete- link across datasets	79
4.8	Performance of hierarchical agglomerative clustering with single- link across datasets	79
4.9	Distribution of pairwise document distances in each dataset . . .	80
5.1	Example documents on <i>smoking and health</i>	91
5.2	Concept-document matrix of the example documents in Figure 5.1Ex- ample documents on <i>smoking and health</i> figure.caption.44	92
5.3	Concept graph of the first example document in Figure 5.1Example documents on <i>smoking and health</i> figure.caption.44	95
5.4	Distribution of concept’s local and relative context centrality on the SmallReuters dataset	100
5.5	Plain and semantically enriched document similarity	104

5.6	Distribution of concept relatedness on the SmallReuters dataset .	108
5.7	Performance of the enriched similarity measure's components on the Med100 dataset	111
6.1	Local and relative context centrality of concepts in two sample documents from the HE50 dataset	123
6.2	Concept groups in the example documents in Figure 6.1Local and relative context centrality of concepts in two sample documents from the HE50 datasetfigure.caption.57	125
B.1	Weka's organization structure of data preprocessors	187
B.2	Options of the filters for creating concept-based text representations	188
B.3	Options of the plain cosine and the semantically enriched distance functions	190
B.4	Options of the DocumentPair filter and the LearnedDistance function	191
B.5	Options of the SimpleKMeansReweighted clustering algorithm . . .	193

1

Introduction

Thematic categorization of text dates back to the Seven Epitomes (Qilue), the first Chinese classification system created in 26 B.C.E. and finished 20 years later (H.-L. Lee, 2010). It consists of six topic categories: six arts, masters, lyrics and rhapsodies, military texts, divination and numbers, and formulae and techniques; which are further refined into thirty-eight subcategories. All existing books in Chinese at that time—there were only 13,269 of them—were manually classified into categories in this system. People realized the value of thematic organization in facilitating information access, even for a corpus that seems scant nowadays.

Our capacity to accumulate and store information has evolved since then, particularly with the aid of computers and information technologies. Assessing the main topics of texts and organizing them into meaningful structures is a labour-intensive and time-consuming process, which has become infeasible with the sheer volume of electronically available information in today's world. A revolution is required in how we organize information, in order to match our capacity to understand and interpret information with our increasing capacity to collect it.

Clustering is a technique that automatically analyzes the relations among texts and organizes them to form thematically coherent structures—clusters of texts that share similar topics. Automatic clustering does not require any human intervention, nor does it need any prior knowledge about the texts, which makes it a widely applied method for information analysis. The clusters that are discovered can facilitate exploring, searching and interpreting a body of texts (Hearst and Pedersen, 1996; Zamir and Etzioni, 1998; Dhillon, 2001), as well as organizing

new texts that appear in the future (Slonim et al., 2002; Z. H. Zheng et al., 2005).

Traditionally, each text—such as a news article, a scholarly publication, a document in a digital library collection, or even a fragment of a document—is represented by the words it mentions, each being weighted according to how often it occurs in the text. Their positions and order of occurrences are not considered. This representation is called the *bag-of-words* model (van Rijsbergen, 1979), and has been the most popular way of representing textual content for information retrieval, text classification and clustering. Basically, texts are considered as thematically similar if they have enough words in common.

Compare this computerized model with how humans perceive and digest information from text. It has two shortcomings. First, the model is ambiguous: it ignores the fact that different words can have the same meaning while the same word might have different meanings in different contexts. Humans can easily resolve the intended meaning of an ambiguous word, either consciously or sub-consciously, using extensive knowledge obtained from previous experience (McDonald and Ramsar, 2001). Thus our interpretation is unambiguous. Second, the model is orthogonal: it assumes that words are independent of each other. In fact, they never exist as isolated language units but always relate to each other to form meaningful lexical structures or to continuously convey an idea. When comprehending text our thoughts constantly utilize the relations between words to facilitate understanding (Altmann and Steedman, 1988).

This thesis explores methods for overcoming shortcomings of the bag-of-words model by using *concepts* instead of words as descriptors of text contents, the goal being to improve text clustering effectiveness. Concepts are defined as *units of knowledge* that abstract and represent a set of perceivable objects with the same characteristics, according to the *International Standard for Terminology Work—Principles and Methods* (ISO, 2009). Concepts are unambiguous. Objects with different characteristics are abstracted into different concepts, even though they might be referred to using the same literal expression. For example, *orange* can refer to the fruit, the colour, a bicycle maker, the longest river in South Africa and the city of Orange located in California; and each meaning is represented by a different concept. Concepts have relations with each other, and, because they are unambiguous, the relations can be explicitly defined. A set of concepts structured

according to the relations among them forms a *concept system* (ISO, 2009).

We call methods that use concepts and their relations to facilitate clustering *concept-based text clustering*, to differentiate them from the traditional word-based paradigms. This thesis investigates how concept systems can be effectively and efficiently exploited to assist concept-based text clustering, and develops a toolkit called Katoa¹ (knowledge assisted text organization algorithms) that represents texts by the concepts they mention, and enriches the clustering process to take the semantic relations among concepts into account.

1.1 Motivation

Concept systems provide precisely the knowledge needed to overcome the ambiguity and orthogonality problems of the bag-of-words model: a mapping between objects and concepts, and diverse relations among concepts. Figure 1.1 illustrates the components of a concept system and the relations between them, according to the ISO (2009) standard. Objects are perceived and abstracted into concepts, which are designated, defined and explained.

Katoa aims to recognize any *object* that can occur in natural language texts. Objects can be material and concrete such as rivers, immaterial and abstract such as the error rate of an algorithm, or imaginary such as a fictional character. The purpose of recognizing objects is to group them into meaningful units—their corresponding concepts—because to analyze the topics in text, there is no need to differentiate every single object if they convey the same meaning. For example, *U.S.A.*, *the States* and *United States of America* are considered as different objects with the same meaning, whose occurrences indicate that the text is talking about the country.

Designations can be regarded as handles by which concepts can be uniquely, succinctly and conveniently referenced. For example, 3434750 for the concept *North American republic containing 50 states—48 conterminous states in North America plus Alaska in northwest North America and the Hawaiian Islands in the Pacific Ocean; achieved independence in 1776*, which represents the above objects. They are usually appellations: names or titles in running text that

¹Katoa is a common Māori word meaning *everybody*.

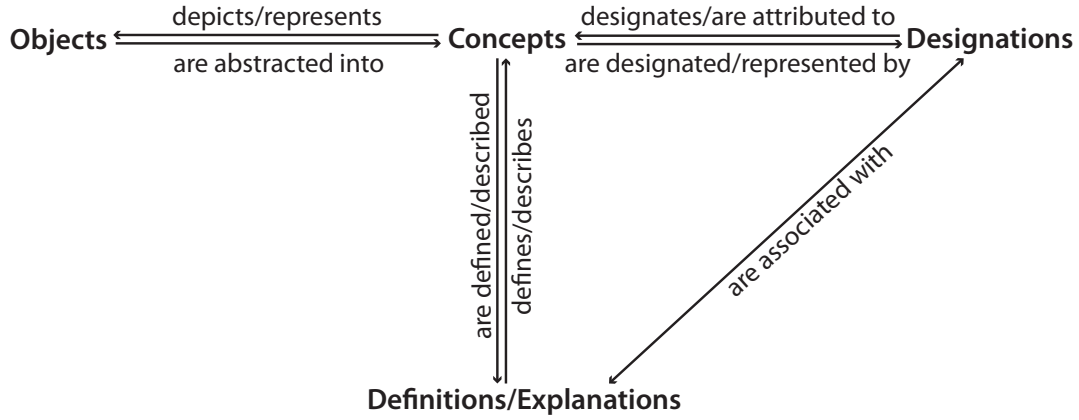


Figure 1.1: Framework for a concept system

evoke the concepts; they can also be symbolic or numeric identifiers, which is common in computer-based concept systems. Being unique concept identifiers, they usually become the actual features used by a concept-based representation model to succinctly denote their associated concepts.

A concept is clarified by a definition and sometimes an explanation, both of which depict the reason for its existence: the characteristics that define which objects are grouped together to form this particular concept. Furthermore, explanations usually provide information about related concepts. For example, explaining *New Zealand* is likely to involve *country*, *Pacific Ocean*, and *Māori*, which are all closely related to *New Zealand*. Although a concept system's structure, by definition, is constructed based on relations among concepts, it usually encodes only one or two specific types of relations (see the discussion below). In contrast, definitions and explanations usually involve a wide range of related concepts that stem from various perspectives, and thus present rich additional information on how closely concepts are related.

ISO (2009) categorizes relations among concepts into three types: *generic*, *partitive* and *associative* relations. The first two are hierarchical: structures created based on them form hierarchies that relate *superordinate* concepts that are higher up in the concept hierarchy to their *subordinate* concepts. Generic relations are also known as *is-a* relations: for example, *New Zealand* is a *country*. Partitive relations denote that a subordinate concept is a part of its superordinate concept: for example, *wheel* is part of *car*. Associative relations are non-hierarchical and

cover more general aspects, for example, *New Zealand* is related to *Pacific Ocean* because of its geographical location, and to *Māori* because it is where they dwell. Concept systems are structured based on these relations.

Information encoded in concept systems is valuable and can be exploited in concept-based text representation and clustering. However, there are many challenges. First, although we have discussed shortcomings of the bag-of-words model, it is not clear whether it is necessary to make computers model texts the way that people do: in fact, this simple model still prevails in practical applications. Second, the best way to apply such information is unknown, especially considering that texts can come from any domain and cover diverse topics. Third, each concept system has its own perspective on what kind of concepts and relations it should include and how they should be organized, and it is unclear how such differences might impact the applicability and effectiveness for different texts. This thesis investigates methods for effectively addressing these challenges.

1.2 Thesis statement

This thesis claims that

Representing text by concepts and taking account of the relations among them can significantly improve text clustering over the bag-of-words representation, using standard clustering algorithms.

Establishing evidence to support this claim involves two components: first, use concepts to represent texts; and second, take their relations into account. In fact, Katoa implements concept-based text clustering in three steps. First, it creates a *plain* concept-based representation—the *bag-of-concepts* model—that utilizes concepts instead of words as features. The second step is to implement an *enriched* method that takes concept relations into account during clustering. The third step uses machine learning techniques to combine the first two approaches, so as to further enhance the effectiveness of concept-based clustering. Each step is evaluated individually against the bag-of-words model, and their relative performance is also investigated. The results of these evaluations will provide justification for the thesis statement. Standard clustering methods are used throughout:

the contribution of this thesis is not a new clustering algorithm but an effective text representation and similarity measure that will benefit *any* similarity-based clustering algorithm.

Evaluation of text clustering performance can be categorized into two types: subjective and objective. Subjective evaluation employs humans to assess the quality of the text clusters. This seems to be a natural choice, because clustering is indeed a subjective task: even humans can have different opinions about how similar two texts are in terms of sharing the same topics, and about how they should be organized into clusters (Macskassy et al., 1998; M. D. Lee et al., 2005). However, as with most tasks that require human involvement, subjective evaluation is labour-intensive and expensive. This makes it impractical in many situations; for example, digital libraries often contain many thousands of documents. Furthermore, subjective assessment of overall clustering quality is a complicated process. It can be broken down into three steps, each of which is a challenging task in its own right: determining the main topics of each text, determining the similarity between them, and creating clusters accordingly. Thus subjective evaluations of clustering quality are rare, and usually involve very few documents: for example, only 10 to 16 in Macskassy et al. (1998)’s evaluation.

Objective evaluations are more practical. In fact, many texts are categorized in some way, either explicitly or implicitly. Sometimes they are assigned topics by humans, which provides a basis for grouping. For example, documents in digital libraries usually have manually assigned subject metadata, and news articles on the web can be tagged by readers. Sometimes topics can be identified automatically: for example, posts to a mailing list on *automobiles* usually pertain to this subject (assuming that spam is filtered), and documents returned by querying a search engine are considered relevant to the query topic. In either case, texts can be categorized based on their themes in a cheap and automatic fashion, and often this also reflects human judgements. More importantly, these categorizations provide a well-grounded and efficient basis to perform objective evaluation, especially for large numbers of documents.

This thesis performs objective evaluations only, and evaluates clustering quality in terms of the goodness-of-fit of the resulting clusters against the existing categories. We employ four standard datasets that are widely used for evaluating

clustering algorithms (see Section 3.1): one from the Reuters collection, another from the OHSUMed collection of medical abstracts, and two from the 20News-group collection. Each dataset has its distinct characteristics, and all come with predefined categorizations. For the first two, categorizations are formed based on manually assigned topics, while the others contain posts to mailing lists, so categorizations are based on the subject of the list. Although this thesis does not contribute any new subjective evaluations, Katoa is assessed against human judgements whenever they are available, which include the categorizations created based on manually assigned metadata described above, and the manually assigned thematic similarities between texts that are used for testing Katoa’s similarity measures (see Section 6.1).

These comparative evaluations involve several design considerations: the concept system, the kind of text, the clustering algorithm, and how performance is measured. This thesis tests Katoa with two concept systems (WordNet and Wikipedia, see Section 1.3), on the four datasets above, across different clustering algorithms (see Section 2.3), and uses four standard measures of clustering performance (see Section 2.4). The goal is to systematically evaluate Katoa under various conditions, so as to provide a guide for its application in practice. The impact of each factor—concept system, dataset, and clustering algorithm—is evaluated individually, so that users can choose the most effective method for each according to the actual requirements and circumstances.

Clustering consists of five components: representation model, similarity measure, clustering method, cluster representation, and validation (Rasmussen, 1992; Jain et al., 1999). Katoa’s concept-based methods address the first two components, and methods targeting each component are evaluated individually. This thesis does not involve presenting the resulting text clusters to users, and thus the cluster representation component is not considered. Apart from this, all the other components are involved.

1.3 Research questions

The preceding thesis statement poses three research questions, each focusing on one important factor in concept-based text clustering. This section discusses them.

1. What kind of concept system can be used?

Concept systems can be as simple as a controlled vocabulary of terms, or as complex as the Cyc project that aims to encompass everyday common-sense knowledge (Lenat and Guha, 1989). There are several ways to categorize them (see Section 2.1.3). For example, a system can be categorized as *lexical* or *encyclopedic*, based on whether it encodes *word* or *world* knowledge (Fellbaum, 1998).

Different systems have distinct domain focus, coverage, comprehensiveness, granularity and accessibility. This thesis aims to develop methods that are generic and applicable to general domains, so the systems that they consult need to have wide coverage and be comprehensive. Those that focus on specific topic domains are not considered. Developing an open-source toolkit for concept-based clustering is a secondary goal of this research; thus the concept systems used must be openly accessible.

Based on these considerations, we examine two systems—WordNet, a lexical system, and Wikipedia, an encyclopedic system. WordNet is a large lexical database developed and maintained by Princeton University since the 1980s (Miller, 1985). Wikipedia is a burgeoning online encyclopedia collaboratively created and actively maintained by hundreds of thousands of Internet users worldwide. Both are representative in their comprehensive coverage of information and broad applicability, and both are openly accessible. Appendix A shows examples of the concepts identified in documents from the four experimental datasets.

Concepts are word senses in WordNet, and each concept consists of a group of interchangeable synonyms—terms that have the same meaning. For example, the concept for *a machine for performing calculations automatically* contains six synonyms, including *computer*, *computing device* and *data processor*. Concepts are organized hierarchically based on generic relations (called *hyponymy* and *troponymy* in WordNet) and partitive relations (called *meronymy* and *entailment* in

WordNet). For example, *robin* is a kind of *bird*, and *lollop*, *toddle* and *stumble* are different types of *walk*, which are connected through the generic relations. Associative relations are auxiliary in WordNet: there are only a few of them, such as the *coordination* relation between sibling concepts that share the same superordinate concept, for example, between *lollop*, *toddle* and *stumble*.

Concepts are individual articles in Wikipedia, each describing a particular concept. For example, the Wikipedia article *Computer* succinctly describes the history of computing, the basic components of a computer, and a variety of other related topics. Each Wikipedia article is usually assigned to one or more subject categories. For example, *Computer* belongs to two categories: *Computers* and *Computing*. The Wikipedia category structure is usually considered as the counterpart to WordNet’s hierarchical structures (Strube and Ponzetto, 2006): categories are included in one or more general categories. However, this structure is not a hierarchy but an acyclic graph, and sometimes there are also cyclic inclusions, even though the Wikipedia style guide explicitly advises against this. In fact, hierarchical concept relations are not the dominant type in this system. Associative relations outside the category structure are ubiquitous in Wikipedia, thanks to the extensive hyperlinks that exist between Wikipedia articles. These inter-article hyperlinks present various kinds of relations. For example, the article *New Zealand* points to about 630 other Wikipedia articles, covering concepts that are related from the perspectives of history, politics, environment, economy, demography and culture.

Each concept system’s characteristics might impact its overall effectiveness in clustering. For example, their distinct structures require different methods for mapping objects in running text to concepts in these systems, and for identifying and utilizing the relations among concepts. How these distinctions might affect clustering is a question that needs to be explored. Their relative performance is also worth investigating, especially considering the various factors involved, such as the impact of different subject domains and clustering algorithms. Knowing which system is more likely to be effective is valuable, especially for choosing the right one to consult in practical applications.

2. What kind of text can be handled?

Katoa aims to handle all kinds of texts and topics, thus its methods and the concept systems used are independent of any particular domain. Nevertheless, the domain specificity of topics *can* affect the match between texts and information in the concept system. For example, if a text collection is mainly about sports with topics that are specific to this particular domain, whereas the concept system only has a limited coverage of sports-related concepts, it is likely that only a few useful concepts can be identified in these texts. Although Katoa is designed to be domain independent, this thesis tests it with both types of text: three datasets cover assorted domains, while the fourth focuses on the medical domain. Tests will show whether Katoa’s concept-based clustering methods can be effectively applied in both cases.

Intuitively, the problem of *ambiguity* is more likely to occur when the topics are diverse, in which case it is more likely that the same word will be mentioned in different contexts, and have different meanings in the same collection. The experimental datasets have various distributions of topics: two contain a broad range (23 and 30 topic categories), and the other two only cover three categories: one with three closely related topics and the other with three completely unrelated topics. In contrast, the problem of *orthogonality* is likely to manifest itself when texts are short, for then they are less likely to have any surface overlap. Documents in the experimental datasets vary from 1 to 16232 words per document, the average for each dataset ranging from 157 to 338 words. Section 3.1 and Appendix C provide more details about these datasets. Because all these experimental datasets consist of distinct documents that are clearly delimited from each other, this thesis uses *text* and *document* interchangeably.

The variety of topic domains, distributions and document lengths helps us to investigate whether these factors will affect the effectiveness of concept-based clustering, and if so, how. In particular, they might influence the relative effectiveness of the two concept systems. These open questions are investigated in this thesis.

3. How can clustering utilize relations among concepts?

Methods for utilizing concept relations can be categorized into two general types: those that use a specific type of relation (Hotho et al., 2003; Hu et al., 2008) and those that use the quantified overall relatedness among concepts (Budanitsky and Hirst, 2001; Milne et al., 2007). For example, if concepts from different texts share the same superordinate concept, such as *robin* and *sparrow*, this reveals a semantic connection between these texts, even though they might have no concepts in common at all. In this case, *robin* and *sparrow* can be connected through a specific type of relation: their *is-a* relations with *bird*. However, concepts can relate to each other in many aspects and through many kinds of relations. For example, *sparrow* is related to its food *seed* and *insect*, its origin in *Europe*, *Africa* and *Asia*, and even the novel *The Dark Half* by Stephen King, which depicts sparrows as psychopomps² that guide souls into the next world. Restricting to one specific relation or a set of relations will undoubtedly miss out a variety of other relations that might also contribute to the thematic connections between different texts.

Quantifying semantic relatedness among concepts, regardless of the specific types of relations they have, is a more flexible and generic approach. After all, clustering is affected more by how closely texts are related to each other than by why they are related. Thus Katoa uses quantified semantic relatedness instead of specific relation types to relate concepts and to connect texts beyond their surface overlap.

The measure for assessing concept relatedness usually differs for each concept system, due to its distinct structure. Developing effective concept relatedness measures is a research problem in its own right (Resnik, 1995; Leacock and Chodorow, 1997; Strube and Ponzetto, 2006; Gabrilovich and Markovitch, 2007; Milne and Witten, 2008a). This thesis draws on extensive prior work for each concept system used—WordNet and Wikipedia. It does not develop new relatedness measures, but uses existing measures that are both accurate and efficient. One common way to evaluate the accuracy of a relatedness measure is to compare its predictions with human judgements—relatedness manually assigned by human raters—and the more consistent the better. The measures used in Katoa have been evaluated

²Psychopomps are creatures, spirits, or angels that escort newly deceased souls to the after-life.

this way. Nevertheless, their impact on Katoa’s concept-based clustering methods are unknown and require investigation.

It is worth clarifying that the *synonym* relation is not a type of concept relation, because it only exist between objects. Terms that have the same meaning are synonyms. Concepts do not have such a relation, because each concept is unique in the meaning and knowledge it represents: no concept will be a synonym of another concept. Synonym relations are handled during the mapping process, when synonyms are grouped and mapped to the concept that corresponds to their meaning.

1.4 Contributions

The thesis makes the following research contributions.

- Techniques for linking texts to concepts in WordNet and Wikipedia are compared. Each technique is discussed, implemented and evaluated in the task of text clustering.
- New text representation models that use concepts instead of words are developed, and their effectiveness in text clustering is evaluated.
- Three new methods are developed for utilizing semantic relations between concepts to improve text clustering, and their effectiveness is evaluated.
- A new similarity measure is defined that employs machine learning to combine various channels of information on thematic similarity between texts. The learned measure is evaluated both against human judgements on pairwise text similarity and in the clustering task, on the four experimental datasets.
- The two most commonly used clustering algorithms, k -means and hierarchical agglomerative clustering (see Section 2.3), are compared in concept-based text clustering.
- The impacts of several relevant factors—representation model, dataset characteristics (domain specificity and topic distributions), the concept sys-

tem, the clustering algorithm and the method for incorporating concept relations—on each other are investigated.

- Katoa, an open source toolkit, is created that implements the new representation models, the semantically enriched clustering methods and the machine learned similarity measure.

Katoa is written in Java and developed on top of the Weka machine learning workbench (Witten et al., 2011). Appendix B provides a detailed description of the system.

Five publications have appeared in peer-reviewed national and international conferences:

- Huang, A. (2011) Learning document similarity. In *Proceedings of the Ninth New Zealand Computer Science Research Student Conference*, Palmerston North, New Zealand.
- Huang, A. (2010) Combining global semantic relatedness and local analysis for document clustering. In *Proceedings of the Eighth New Zealand Computer Science Research Student Conference*, Wellington, New Zealand.
- Huang, A., Milne, D., Frank, E., Witten, I. H. (2009) Clustering documents using a Wikipedia-based concept representation. In *Proceedings of the 13th Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 628–636, Bangkok, Thailand.
- Huang, A., Milne, D., Frank, E., Witten, I. H. (2008) Clustering documents with active learning using Wikipedia. In *Proceedings of the Eighth IEEE International Conference on Data Mining*, pp. 839–844, Pisa, Italy.
- Huang, A. (2008) Similarity measures for text document clustering. In *Proceedings of the Seventh New Zealand Computer Science Research Student Conference*, Christchurch, New Zealand.

1.5 Thesis structure

This thesis is structured as follows. Chapter 2 provides the background for text clustering in general, and specifically how information encoded in concept systems can be applied to the task. First each component of text clustering is analyzed, problems with traditional methods are identified, and how concept-based clustering can tackle them is described. Then related work on the most relevant two components to this thesis—representation models and similarity measures—is reviewed. These reviews and discussions will clearly depict the motivation for this research.

There are various ways to perform, test and evaluate clustering. Chapter 3 specifies the experimental setup, including the experimental datasets, the clustering methods and the evaluation methodology. The purpose of this short chapter is to establish a baseline for the research, to which the methods in Katoa are compared.

Chapters 4 to 6 each examines one aspect of the thesis statement. Chapter 4 investigates the hypothesis that *representing text by concepts provides a more effective basis for text clustering than the traditional bag-of-words model*. It starts by discussing the two concept systems used in this thesis, WordNet and Wikipedia; their characteristics; and methods for identifying objects in running text and mapping them to concepts in these systems. This involves issues like handling morphological variation and sense disambiguation—identifying the intended sense of a word or phrase. The resulting concept-based representation models are tested by clustering standard text collections, and evaluated against human judgement: namely, the existing categories in these datasets. The impact of the various factors—concept system, experimental dataset, and clustering algorithm—are evaluated. Lexical and encyclopedic knowledge can potentially complement each other, because our mental lexicons contain both word and world knowledge (Fellbaum, 1998), and Section 4.5 investigates hybrid models that combine information from each system.

Whereas Chapter 4 focuses on solving the ambiguity problem suffered by the bag-of-words model, Chapter 5 investigates the orthogonality problem, and analyzes how to utilize semantic relations among concepts for better clustering. It

investigates the hypothesis that *utilizing semantic concept relatedness can help to further enhance concept-based text clustering*. We call the models and clustering methods in Chapter 4 the *plain* methods for concept-based text clustering, because they use concepts as orthogonal features that are independent to each other. Chapter 5 develops *enriched* methods for concept-based clustering, the purpose being to improve the plain methods (which is a better baseline than the bag-of-words model). We develop two types of method for utilizing concept relatedness: one uses it to improve the discriminative power of the concept-based representation models, while the other uses it to extend the similarity measure beyond surface overlap of concepts. All these methods are evaluated against both the bag-of-words model and the plain clustering methods.

This thesis explores thematic text similarity from various perspectives, for example, similarities computed based on the different representation models each constituting a single aspect. The next question is whether these aspects can be combined into a better similarity measure, and if so, how. Normally such combinations use handcrafted formulas based on heuristics (Hammouda and Kamel, 2004; Hu et al., 2008; Song et al., 2009; Zhu et al., 2009). Chapter 6 describes a more principled technique that employs machine learning to learn the best formula for combining these aspects. It learns from *training data*, which consists of texts whose thematic similarity to each other is already known. This chapter uses a small dataset with 50 documents, which provides 1225 pairs (excluding self similarities)—that is, 1225 training examples. The similarity for every pair is the average of the similarities manually assigned by several human raters. From these, the machine learning algorithms learn the best formula to map the various perspectives into human judgement. The learned measure is evaluated in two ways: first, against manually assigned similarities, to test its consistency with human judgement; and second, in the text clustering task, to test whether the combination effectively outperforms the individual aspects. In the latter, the learned model is tested on previously unseen texts: the four experimental datasets (see Section 3.1). The hypothesis is that *with machine learning, the learned measure can predict thematic similarity between texts as consistently as human judgement, and is more effective than the cosine measure in text clustering*.

Chapter 7 concludes the thesis and discusses future work.

2

Background

This chapter presents the background to text clustering, with a particular focus on how concept systems can benefit it. This thesis is by no means the first that exploits concept systems to help computers interpreting and processing natural language texts, and this chapter reviews related work in this area.

The clustering process consists of the following basic components (Rasmussen, 1992; Jain et al., 1999) where we have specialized them to apply to text:

- Text representation: generating the features that represent the thematic content of a text, and weighting them appropriately.
- Similarity measure: a function that determines how similar or dissimilar two texts are. Choosing an appropriate similarity measure is of no less importance than determining the representation (Hartigan, 1975).
- Clustering method: an algorithm that effectively organizes texts according to their similarity, putting similar ones into the same cluster and assigning different ones to different groups.
- Cluster representation: a succinct summary of a cluster's content. This is optional, but indispensable if the clusters are to be presented to users, as in a clustering search engine (Carpineto et al., 2009).
- Cluster validation: validating the quality of the results, either manually by employing human evaluators (Macskassy et al., 1998) or automatically using evaluation measures against certain gold standard (see Section 2.4).

The first two components—representation model and similarity measure—are particularly relevant to this research, and the following sections discuss them in detail. Clustering algorithms (Section 2.3) and cluster validation measures (Section 2.4) are not the focus of this thesis, yet they are crucial for fair evaluation. The only component that is not involved in this thesis is cluster representation, although we sketch the potential of concepts as cluster labels in Section 2.5.

2.1 Text representation

Texts are usually represented using the *vector space model* (Salton et al., 1975): each text is expressed as a weighted high dimensional vector, each dimension corresponding to a feature such as a word or concept. Words are the most commonly used feature for describing a text’s content, and the resulting representation is called the *bag-of-words* model. It has been widely applied in almost every field that involves text analysis, including information retrieval, categorization and clustering.

However, the bag-of-words model has certain limitations.

- First is the term mismatch problem (i.e. synonymy): different texts use different words to express the same concept, and the bag-of-words model does not connect synonyms. For example, *New Zealanders* are also known as *kiwis*, but without external knowledge they will be treated as two different features.
- Second is semantic ambiguity (i.e. polysemy): a word can have different meanings depending on its surrounding context, and the bag-of-words model does not capture such differences. For example, *kiwi* can also mean a particular kind of flightless bird in New Zealand or a kind of fuzzy brown egg-shaped fruit, yet it will be treated as a single feature irrespective of its intended meaning.
- The third problem, manifested by the first two, is that the bag-of-words model ignores the connections between words: it assumes that they are independent of each other. The connections include not only the synonymy

and polysemy relations shown above, but also extend to the more general sense of relatedness between words—for example, the extent to which *kiwi* relates to *New Zealand*. Therefore the bag-of-words model only represents texts at the surface level.

- The above three problems not only affect the accuracy of the model and hence the similarity computed based on it, but also incur a fourth problem: the bag-of-words model is not robust, especially with respect to new texts. As an extreme example, when assigning a new text to one of a group of clusters, it will not be properly placed if it does not mention any words in the existing clusters, for it has no surface overlap with any of them.

These shortcomings have been known for a long time, and many methods have been explored to overcome them. The following sections survey these methods and discuss their strengths and limitations.

2.1.1 Phrases and linguistic features

Phrases spring readily to mind. They are more specific than words and therefore less likely to be ambiguous (Lewis and Croft, 1990). For example, *clustering* is a general term that can be further focused by specifying the object being clustered: *text clustering*, *image clustering*, *query clustering*, and *search result clustering*; or the type of method it uses: *hierarchical clustering*, *flat clustering* and *partitional clustering*; or the characteristics of the method: *incremental clustering* and *iterative clustering*; and so on. Two texts that both mention the same phrase, such as *query clustering*, are more similar than those that only share the word *clustering*. Phrases also help to solve the ambiguity problem. For example, *plane* has several possible meanings, while *plane ticket* and *Euclidean plane* are unambiguous.

Phrases (including n-grams) have been extensively investigated to supplement or replace words in text categorization (Caropreso et al., 2001) and information retrieval (Zhai et al., 1997). In the context of clustering, utilization of phrases can be categorized into two types, as we discuss below.

Phrases as alternative features

The first type uses phrases as alternatives to words. Texts are represented by several models; for example, one with words and another with phrases. Similarity values computed from each model are combined to give the overall similarity between two texts. For example, Hatzivassiloglou et al. (2000) investigate noun phrase heads and proper names as features, and combine three similarities: one based on the former, another on the latter, and the third on traditional word vectors, all calculated using the cosine measure of vector similarity (see Section 2.2). Their experimental results show that considering linguistic features improves the overall quality of the resulting text clusters and benefits the subsequent task, which in their case is *topic detection and tracking*.

Furthermore, phrases provide more information about how similar the topics in two texts are. For example, texts tend to be more similar when they share longer phrases. Based on such considerations, Hammouda and Kamel (2004) specifically design a similarity measure for phrases, which combines four aspects with an ad hoc formula: the number and the lengths of matching phrases, their number of occurrences and their significance level in both texts (e.g., phrase matches in titles are more significant). The resulting similarity is then combined with the cosine similarity of the word vectors. Their experiments with the hierarchical agglomerative clustering algorithm (see Section 2.3) also suggest that including phrases benefits clustering: they achieve a 29% improvement over the bag-of-words model on a subset of the 20Newsgroup collection (see Section 3.1 for a description of the collection).

Phrases as cluster labels

Based on the observation that phrases make better cluster labels than words, the second approach for utilizing them prioritizes identifying phrases that are likely to be good labels and then groups texts around them (Zamir and Etzioni, 1998; Pantel and Lin, 2002; Hammouda and Kamel, 2004; Zeng et al., 2004; Stefanowski and Weiss, 2003). This is called *descriptive clustering* (Stefanowski and Weiss, 2003) or *description-centric clustering* (Carpineto et al., 2009), to highlight its underlying motivation: to generate high quality descriptions of the resulting text

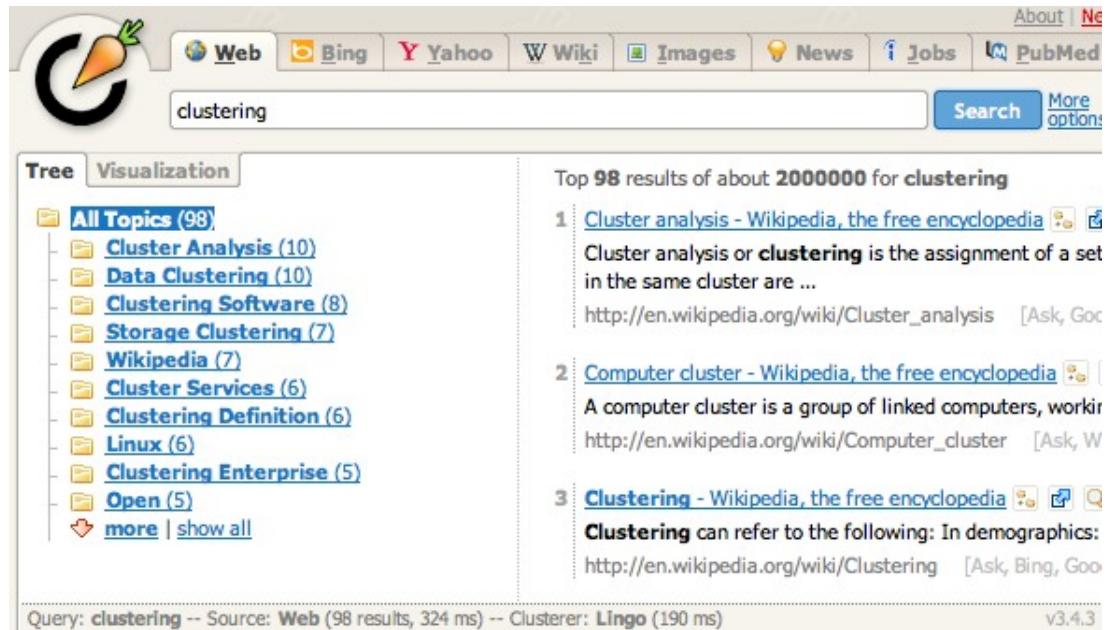


Figure 2.1: A clustered search results of the query *clustering*

clusters. Descriptive clustering dates back to at least Zamir and Etzioni (1998)’s work on suffix tree clustering, which uses the suffix tree data structure to index and organize texts by the phrases they mention.

Good labels are particularly important if the clusters are to be presented to users, and much research focuses on the task of clustering search results. Various techniques such as machine learning (Zeng et al., 2004) and matrix decomposition methods (Stefanowski and Weiss, 2003) have been explored for finding discriminative phrases in a given text collection, and have been shown to be effective. For example, Stefanowski and Weiss’ method has become the foundation of the clustering search engine Carrot¹ shown in Figure 2.1. The cluster labels—such as *cluster analysis* and *storage clustering* in Figure 2.1—are identified as the most dominant topics mentioned in the returned search results, using the *singular value decomposition* technique.

¹The Carrot search engine is available at <http://www.carrot2.org>.

Discussion

The effectiveness of phrases suggests that words are probably not the best features for describing a text's topics. The main reason for preferring phrases over words is that they are semantically more specific and less likely to be ambiguous. Their success indicates that the ambiguity of words reduces the effectiveness of the orthodox bag-of-words paradigm in clustering.

However, phrases are by nature merely sequences of words (or text fragments for languages like Chinese and Arabic), so they *can* be ambiguous. For example, *access point* usually refers to the device to connect to a wireless network, yet it can also mean a rocky point on the Anvers Island of Antarctica. Unless external concept systems are consulted, they provide no semantic information such as the relations between two phrases. Furthermore, they are sparser than words: the number of matching phrases is usually much smaller than the number of matching words (Lewis and Croft, 1990). All these limitations restrict phrases from being a satisfactory solution to the problems in the bag-of-word model.

2.1.2 Term clusters and combinations

Similar words tend to occur in the same documents (Senellart and Blondel, 2003) and within similar contexts (Pereira et al., 1993). Thus words can be grouped based on their co-occurrence in a text collection, each group being viewed as a *latent* concept or topic that is hidden in the collection. This section reviews two widely applied approaches for utilizing such associations between words: term clustering and dimensionality reduction techniques.

Term clustering

Term clustering uses clustering techniques to group terms based on their distributions in a given text collection. Each of the resulting term clusters consists of terms that co-occur frequently with each other. The underlying assumption is that such terms are either similar or closely related, and there is no need to distinguish between them for the clustering task. In contrast, term clusters provide compact and efficient representation of texts.

For example, consider clustering texts about *sports* into categories by individual sport, such as *basketball* and *hockey*. Words like *rebound* and *dunk* only occur in the *basketball* category, and thus they do not need to be distinguished for the clustering task. All words that are strongly indicative of the *basketball* category will be grouped together. This usually drastically reduces the dimensionality, from several thousand words to less than a hundred clusters, significantly reducing the redundancy and noise in the word space (Slonim and Tishby, 2000; El-Yaniv and Souroujon, 2001; Rooney et al., 2006; Chee and Schatz, 2007).

Texts are then represented by the term clusters, and each word’s weight is aggregated to the term cluster it belongs to. This connects texts with related topics yet distinct vocabularies, which is particularly helpful for short texts, such as queries and search result snippets. Indeed, term clustering has been extensively investigated in information retrieval, dating back to the work of Spärck Jones in the 1970-80s (Jones, 1971; Lewis and Croft, 1990). For longer documents, however, studies show that using term clusters yields limited success in the text categorization task (Baker and McCallum, 1998; Bekkerman et al., 2003).

In contrast, in the context of text clustering, term clustering has been shown to be quite effective. Slonim and Tishby (2000) extend their information theoretic clustering method called the *information bottleneck* (Tishby et al., 2000) method to term clustering. Their method works in two steps—first grouping terms based on their occurrence in the texts, and then representing and clustering the texts by the resulting term clusters. Their experiments on the 20Newsgroup collection show a 17% improvement over the complete-link hierarchical agglomerative clustering (see Section 2.3.1) with the bag-of-words model. Term clusters reduce the dimensionality of the feature space from 2000 (words) to 10-50 (term clusters), which contributes to the method’s success.

Frequent itemset clustering shows a similar effect (Beil et al., 2002; Fung et al., 2003). It first applies association rule learning to identify the frequent itemsets of words in a text collection (Agrawal and Srikant, 1994). Each frequent itemset can be regarded as a term cluster consisting of words that occur together in a minimum fraction of the texts. The intuition is that texts belonging to the same category share many frequent itemsets and those from different categories (i.e., topics) share few.

The work of Slonim and Tishby (2000) and Beil et al. (2002) represents the standard double clustering procedure—creating the term clusters first and then the text clusters afterwards (El-Yaniv and Souroujon, 2001; Rooney et al., 2006; Chee and Schatz, 2007). Alternatively, terms can be clustered at the same time as the texts, which is called *co-clustering* (Dhillon, 2001; Rege et al., 2006; Yoo et al., 2006), or after the text clusters are formed (Dhillon and Modha, 2001). For the former, the purpose is to simultaneously utilize the associations among words, texts and between words and texts. For the latter, term clusters are created to generate better cluster descriptions and to facilitate clustering future texts.

Despite the differences in procedure, there are three benefits of using term clusters as features. First, they connect distinct terms that are indicative of the same topic. Second, they provide more compact and efficient text representation. Third, they generate smaller clustering models for future use.

Dimensionality reduction methods

Dimensionality reduction methods first project texts into a new space where each dimension is considered as a *latent* topic in the text collection, and then perform traditional clustering algorithms in the transformed space. These include *latent semantic analysis* (Deerwester et al., 1990; Schütze and Silverstein, 1997), its probabilistic version *probabilistic latent semantic indexing* (Hofmann, 1999), *non-negative matrix factorization* (Xu et al., 2003) and *principal component analysis* (Jolliffe, 2002; Saerens et al., 2004).

With the vector space model, a text collection can be represented by a matrix, where each row corresponds to a term in the collection and the column vectors are the term vectors of the texts therein: the *term-document matrix* of the collection. Take latent semantic analysis (Deerwester et al., 1990) for example. It uses *singular value decomposition* to transform the word space. Given such a term-document matrix A , singular value decomposition breaks it into three matrices: U , S and V in such a way that $A = USV^T$ (Golub and Loan, 1996), where S is a diagonal matrix. The top r columns of matrix U correspond to the r largest singular values of A , which also form an orthogonal approximated space for the term space in A . Each of the r columns is a linear combination of terms, and represents

a *latent topic* (Deerwester et al., 1990). Each document is then transformed to the new space and represented by a weighted vector of the latent topics.

The other dimensionality reduction methods differ in the techniques employed to find the latent topics, yet the topics usually take the same form as linear combinations of terms, such as $0.452 \times \textit{Export} + 0.2013 \times \textit{Employment} + \dots$, where the values of the variables *Export* and *Employment* are the number of times these concepts occur in the document. The main advantage of these methods is to improve efficiency, by drastically reducing the dimensionality of the feature space. Furthermore, the projection is performed in principled ways, so as to retain the most valuable information in the original space. Indeed, the studies mentioned above show that these methods can effectively speed up tasks like text clustering and categorization, usually with little or even no loss in accuracy.

Discussion

Both term clustering and dimensionality reduction depend heavily on the input data, and this cause several restrictions. First, it is difficult to generalize the latent topics—term clusters for the former and term combinations for the latter—to new data, especially for previously unseen terms. An extreme case is when the new texts have completely different topics from the data used for constructing the clusters: for example, when building the clusters with texts on *sports* when most of the new texts discuss *politics*. In such cases, it is necessary to re-generate term clusters and combinations based on both existing and new texts.

Second, different text collections result in different term clusters and combinations, making it difficult to connect clusters derived from different collections. For example, *puck* and *dunk* will be assigned to two groups if we are clustering texts on sports by individual sport; however, if half the texts are about sports while the other half are about politics, they may instead be assigned to the same group.

Third, although these methods reduce redundancy and dimensionality of the input term space, they tend to broaden the meaning by combining terms together, which does not necessarily contribute to semantic clarity. For example, it is difficult for humans to derive the actual topic represented by a linear combination

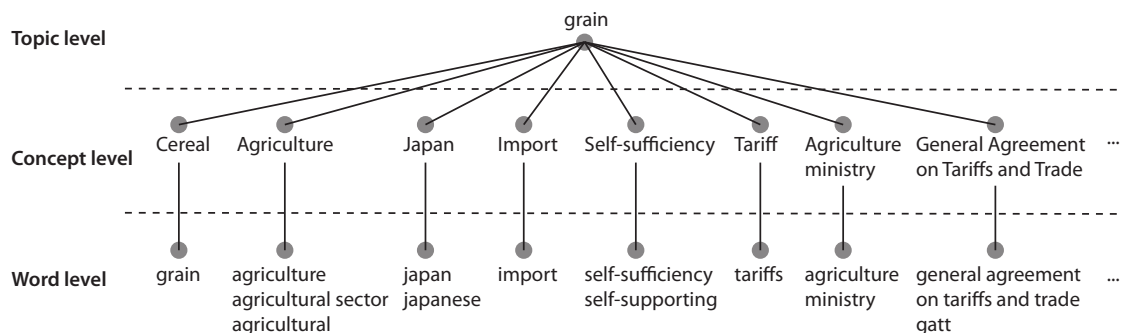


Figure 2.2: The conceptual model of a document in the Reuters collection

of terms, such as in the preceding example.

2.1.3 Concepts

The preceding survey exposed the trade-off between statistical redundancy and semantic clarity, and we have reviewed methods that tackle the problem at both ends of the spectrum: phrases are less ambiguous but sparser, while term clusters and combinations are more compact but less generic and specific. Concepts—units of knowledge—provide a solution that balances this trade-off.

Concepts are unambiguous: each concept represents a unique meaning. Synonyms are mapped to the same concept while terms with multiple meanings are mapped to different concepts corresponding to their intended meanings. For example, *the U.S.* and *United States* are recognized as one feature, while texts about Greek mythology and astronomy will not be mistakenly connected if both of them mention *pluto*. Concepts also provide a compact and efficient representation: the number of concepts in texts is usually far fewer than words (see Section 4.4).

The conceptual model in Figure 2.2 illustrates why concepts are better thematic descriptors than words. It shows document #12338 in the Reuters collection (see Section 3.1), which discusses the Japanese government’s urge to increase international trade in its agricultural market, and is assigned to the *grain* category. Figure 2.2 clearly shows that concepts provide a more succinct and concise description of the document’s content than words, especially considering that phrases *agricultural sector*, *agricultural ministry* and *general agreement on tariffs*

and trade are split into single word terms. Meanwhile, concepts are more expressive than topics: the topic *grain* is elaborated from several aspects, including both trading-related concepts and agriculture-related concepts. Thus the concept level provides an appropriate granularity of both abstraction and comprehensiveness.

Furthermore, semantic connections among concepts can be used to capture similarity between documents beyond their surface forms. For example, documents that mention *crop* and *export* will be considered similar (to a certain extent) to this one, because both concepts are closely related to those in Figure 2.2, although the documents might have no concepts in common at all. By taking the relatedness between concepts into account, new documents with previously unseen concepts can connect to existing clusters—a problem that methods reviewed so far fail to solve.

Creating a concept-based representation is more complicated than creating the bag-of-words representation, and is an ongoing research problem in its own right (Gabrilovich and Markovitch, 2006; Hu et al., 2008; Zhu et al., 2009). In general, concepts are selected from an external concept system and assigned to a text based on its content. Every concept system has its unique characteristics in terms of the kind of concept and concept relation encoded, its structure and organization of concepts, and coverage and comprehensiveness. The rest of this section discusses these characteristics of several commonly used concept systems, and research that utilizes them for text analysis.

Encyclopedias vs. lexical resources

Given the standard definition of concepts as units of knowledge, encyclopedias like the Encyclopedia Britannica and Wikipedia are good sources of concept knowledge. They cover extensive concepts in all branches of knowledge, and have a particular focus on factual explanation of the concepts (Hartmann and James, 1998). For example, Figure 2.3 shows the articles in Britannica and Wikipedia that explain the concept *computer*.² Both provide a detailed description of the concept and a rich assembly of related concepts.

²Pages retrieved from the online Britannica at <http://www.britannica.com/EBchecked/topic/130429/computer>, and Wikipedia at <http://en.wikipedia.org/wiki/Computer>.

Both resources are comprehensive. The most recent 2010 version of Britannica contains 65,000 articles, with extensive explanations that use 44 million words (Britannica, 2011). The current version of Wikipedia has about 2.7 million articles.³ However, Britannica is commercial whereas Wikipedia is freely accessible.

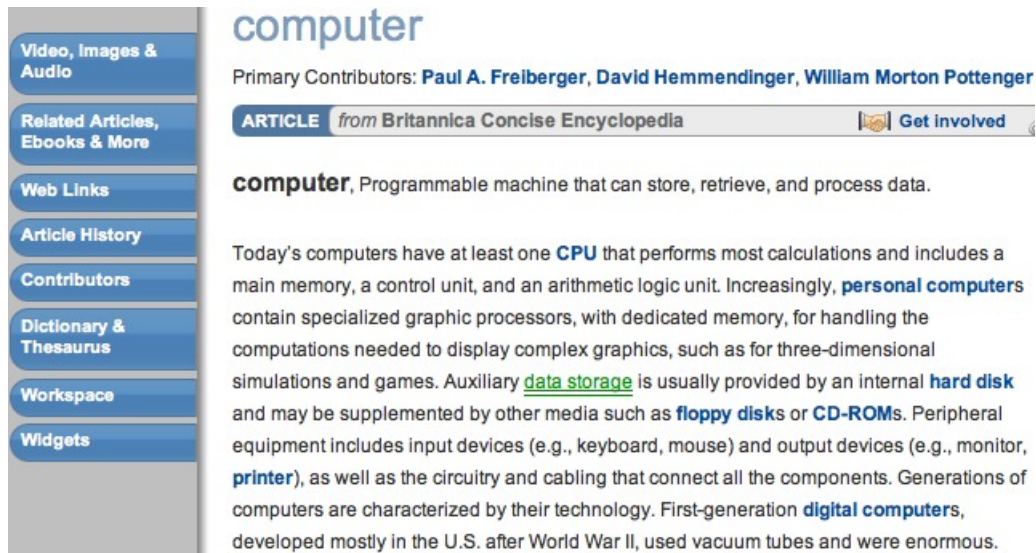
Indeed, Wikipedia, because of its open accessibility and comprehensive world knowledge, has been extensively and effectively exploited to facilitate better understanding of text. Studies show that representing text with Wikipedia concepts is more effective than merely using word vectors when assessing the semantic relatedness between texts (Gabrilovich and Markovitch, 2007; Yeh et al., 2009), and for information retrieval (Milne et al., 2007; Potthast et al., 2008), text categorization (Gabrilovich and Markovitch, 2006; Wang and Domeniconi, 2008) and clustering (Banerjee et al., 2007; Hu et al., 2008).

All these studies take Wikipedia articles such as the one shown in Figure 2.3(b) as *concepts*. Words and phrases in running text are mapped to Wikipedia articles that correspond to their intended meaning. These articles become features in the representation model, for example, by using their unique identifiers in Wikipedia.

Obviously, the effectiveness of the resulting representation depends on how concepts are identified. Strube and Ponzetto (2006) use string matching against article titles, which only counts surface matches and thus is extremely restrictive. Mihalcea and Csomai (2007) and Milne and Witten (2008b) derive an intermediate vocabulary from Wikipedia, and connect terms in running text to Wikipedia articles via this vocabulary (see Section 4.2). Gabrilovich and Markovitch (2006) employ full-text level analysis to assess the relevance between a Wikipedia article and a given text, which essentially indexes the text with all the concepts in Wikipedia, yielding a weight vector of millions of features. Truncation is usually applied to reduce the extremely high dimensionality.

Lexical resources such as WordNet have also been explored to identify concepts in running text. These resources predominantly provide information about individual words, rather than general conceptual knowledge (Gabrilovich and Markovitch, 2009). Take WordNet for example. Words and phrases in text are

³All statistics of Wikipedia in this thesis are computed based on the snapshot taken on March 6, 2009, unless otherwise specified.



computer

Primary Contributors: [Paul A. Freiburger](#), [David Hemmendinger](#), [William Morton Pottenger](#)

ARTICLE from [Britannica Concise Encyclopedia](#) [Get involved](#)

computer, Programmable machine that can store, retrieve, and process data.

Today's computers have at least one **CPU** that performs most calculations and includes a main memory, a control unit, and an arithmetic logic unit. Increasingly, **personal computers** contain specialized graphic processors, with dedicated memory, for handling the computations needed to display complex graphics, such as for three-dimensional simulations and games. Auxiliary [data storage](#) is usually provided by an internal **hard disk** and may be supplemented by other media such as **floppy disks** or **CD-ROMs**. Peripheral equipment includes input devices (e.g., keyboard, mouse) and output devices (e.g., monitor, **printer**), as well as the circuitry and cabling that connect all the components. Generations of computers are characterized by their technology. First-generation **digital computers**, developed mostly in the U.S. after World War II, used vacuum tubes and were enormous.

(a) Britannica

Computer

From Wikipedia, the free encyclopedia

For other uses, see [Computer \(disambiguation\)](#).

"Computer technology" redirects here. For the company, see [Computer Technology Limited](#).

A **computer** is a programmable [machine](#) designed to sequentially and automatically carry out a sequence of arithmetic or logical operations. The particular sequence of operations can be changed readily, allowing the computer to solve more than one kind of problem.

Conventionally a computer consists of some form of [memory](#) for data storage, at least one element that carries out arithmetic and logic operations, and a sequencing and control element that can change the order of operations based on the information that is stored. Peripheral devices allow information to be entered from external source, and allow the results of operations to be sent out.

A computer's processing unit executes series of instructions that make it read, manipulate and then store [data](#). Conditional instructions change the sequence of instructions as a function of the current state of the machine or its environment.



(b) Wikipedia

Figure 2.3: Encyclopedia entries for the concept *computer*

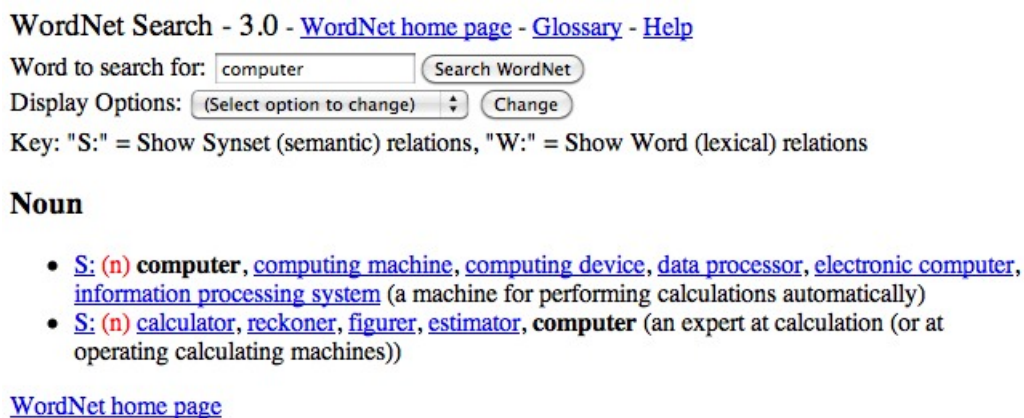


Figure 2.4: Lexical entries for the concept *computer*

mapped to concepts in WordNet—their corresponding word senses. For example, Figure 2.4 shows the WordNet concepts for the term *computer*: the most common sense—a particular type of machine—and a rarer sense—the person that uses a *computer*. Each concept has a definition and a list of synonyms associated with this sense.⁴

Beyond the information displayed in Figure 2.4, WordNet encodes the semantic relations among concepts, such as generic (hypernymy) and partitive relations (meronymy), and provides functions for mapping terms in free text to concepts therein (see Section 4.1). The resulting concept-based representation has been extensively utilized: in measuring semantic relatedness between texts (Mohler and Mihalcea, 2009; Tsatsaronis et al., 2009), information retrieval (Gonzalo et al., 1998; Voorhees, 1998), text categorization (Scott and Matwin, 1999; Gabrilovich and Markovitch, 2004) and clustering (Hotho et al., 2003; Recupero, 2007).

WordNet provides a term vocabulary and most methods use it to map terms in running text to WordNet concepts. For example, *computers* is first mapped to the WordNet term *computer* and then resolved to one of the two concepts in Figure 2.4 through sense disambiguation. Compared to encyclopedias, lexical resources usually cover more common words and expressions, some of which might not resolve to any factual concept in an encyclopedia. For example, about 19% of WordNet concepts are adjectives and adverbs (WordNet, 2011), such as *quick*

⁴Retrieved from WordNet's online search interface at <http://wordnetweb.princeton.edu/perl/webwn?s=computer>.

and *significantly*, which are usually not included in encyclopedias.

Less commonly explored resources include Roget's thesaurus, which encodes lexical knowledge, and web directories such as the Open Directory Project (Jarmasz, 2003; Gabrilovich and Markovitch, 2007), which mainly contain factual concepts. Yet they usually suffer from either a limited coverage or insufficient semantic information. For example, Roget's thesaurus has only about one thousand entries, which is tiny compared to WordNet's hundred thousand and Wikipedia's millions of concepts (see Chapter 4). Web directories have larger coverage (hundreds of thousands for the Open Directory Project), but do not encode the rich semantic relations between concepts as WordNet and Wikipedia do.

Domain independent vs. domain dependent resources

All the concept systems reviewed so far are domain independent: they can be applied to texts from different topic domains—at least in principle. Domain dependent resources such as the Medical Subject Headings (MeSH)⁵ and Agrovoc⁶ have also been studied and compared to their domain independent counterparts.

Both MeSH thesaurus and Agrovoc organize concepts based on hierarchical relations—superordinate concepts are more generic and broader than subordinate ones. Because of their dependence of the topic domain, research and applications of these systems are usually restricted to processing texts from that domain (Zhu et al., 2009; Bloehdorn and Hotho, 2004).

Bloehdorn and Hotho (2004) find that domain-specific systems tend to be more effective, in the context of text categorization, when they match the domain of the texts. Their experimental results find that the MeSH thesaurus is more effective than WordNet with the OHSUMed collection (see Section 3.1) because both focus on the medical domain.

⁵MeSH is United State National Library of Medicine's controlled vocabulary thesaurus, which is available for browsing at <http://www.nlm.nih.gov/mesh/MBrowser.html>.

⁶Agrovoc is a multilingual thesaurus covering the terminology of agriculture, food and related domains, created and maintained by the Food and Agriculture Organization of the United Nations, and can be browsed online at <http://aims.fao.org/website/Search-AGROVOC/sub>.

Categorization of concept systems

So far we have categorized concept systems by their type: encyclopedias and lexical resources; and by their generality: domain independent and domain specific. Based on the comprehensiveness of the semantic information encoded, they can be categorized as controlled vocabularies, glossaries, thesauri, and ontologies (McGuinness, 2003), with increasing comprehensiveness.

A controlled vocabulary is a manually defined list of terms, and is intended to ensure that a concept is always referred to using the same term. This eliminates ambiguity and variations, and is indispensable for tasks like cataloging and metadata assignment. A glossary is a vocabulary that includes a definition of each term. It is not necessarily a *controlled* vocabulary: it can be automatically gathered and maintained (Park et al., 2002).

Thesauri, including Library of Congress Subject Headings, MeSH and Agrovoc, additionally encode relations between terms such as synonyms, broader and narrower terms. Ontologies extend thesauri by introducing more strictly defined relations such as inheritance (e.g., *is-a* relations: *New Zealand* is a *country*) and properties (e.g., a *university* has a property *location*).

This thesis utilizes two concept systems: WordNet and Wikipedia. WordNet is usually considered as an ontology, due to its *is-a* hierarchies. Wikipedia also encodes such relations (in its category structure), but in a less rigorous way. For example, the Wikipedia article *English language* belongs to several categories, including the category *English language* and categories like *Languages of Australia* that specifies the languages spoken in a region, while in WordNet it belongs to just one: *West Germanic*.

Discussion

Based on the analysis at the beginning of this section, concepts can overcome the limitations of phrases and term clusters. For concepts to be effective, choosing the right concept system and an accurate way for mapping terms in free text to concepts is important. The different categorizations contribute to a better understanding of each system's characteristics, which is particularly valuable for making the right choice in applying them in practice. Both WordNet and Wikipedia are

generic, domain independent, have good coverage and provide rich information about how concepts are related and how strong the relations are, which makes them ideal resources to consult for measuring the thematic similarity between two texts.

2.1.4 Utilizing different features

Although concepts have many advantages over words and seem to be a suitable replacement, the question arises as to whether it is even more beneficial to combine words and concepts. For example, words potentially can complement concepts when the concept system consulted does not suit the input texts, in which case the effectiveness of the resulting concept-based representation might be impacted. This section surveys and discusses strategies that have been applied to utilize both words and concepts, in the context of text clustering.

Hybrid representation schemes

The most straightforward strategy is to combine the feature vectors: using words and concepts in the same representation. Hotho et al. (2003) develop three strategies for this: using concepts to expand, augment and replace words. They find in their text clustering experiments that using both features—words and WordNet concepts—is the most effective approach. However, this strategy drastically increases the dimensionality and the redundancy between features, which is often undesirable (see Section 4.5).

Multiple representations

Either words or concepts are sufficient to represent texts by themselves. The second strategy is to derive similarity scores using both representations individually and combine the scores. This requires choosing the right form for the combination, such as a linear weighted sum (Hammouda and Kamel, 2004; Hu et al., 2008; Song et al., 2009; Zhu et al., 2009), and determining values for the parameters.

The advantage of this—using a combination of similarities instead of a combination of features—is that it avoids increasing dimensionality and does not affect the accuracy of individual similarity values. However, estimating the weights can

be computationally expensive, because it usually requires an exhaustive search that repeatedly and iteratively tests every combination of weights in terms of the quality of the resulting clusters, and chooses the set that yields the best clusters. Because of this complexity, exhaustive search is limited in the number, range and resolution of the weights considered.

Ensemble methods

Instead of combining the similarity values, ensemble methods postpone integration until after clusters have been identified. They first perform clustering using each model separately, and then combine the resulting clusters based on co-associations between texts. For example, Fodeh et al. (2009) cluster documents individually with both nouns (words) and noun concepts from WordNet, yielding two co-association matrices where each row and column corresponds to a document and each cell's value indicates whether the corresponding row and column documents have been assigned to the same cluster or not.

Then the co-associations vote for whether two documents should be assigned to the same group, and finally standard techniques such as hierarchical agglomerative clustering are performed based on the votes to obtain the final text clusters. Ensemble methods require several clustering results to be generated, which is computationally expensive—especially when all the representations are high dimensional.

Discussion

These studies on combining words and concepts show limited success in text clustering. However, all these methods are based on heuristics, and are fully unsupervised. This thesis explores—from both the unsupervised and supervised perspectives (see Section 4.5 and Chapter 6 respectively)—whether it is necessary to combine different features; and if so, how.

2.2 Similarity measures

There are many measures for assessing the thematic similarity (or distance) between two texts (van Rijsbergen, 1979; Willett, 1988). The most commonly used one is the *cosine* measure (van Rijsbergen, 1979), which calculates the similarity between two texts as the cosine value of the angle between their feature vectors in the high dimensional space—using the the vector space model (Salton et al., 1975). Formally, let \vec{d}_A and \vec{d}_B be the feature vectors of texts d_A and d_B . Their similarity is calculated as:

$$\begin{aligned} \text{cosine}(d_A, d_B) &= \frac{\vec{d}_A \cdot \vec{d}_B}{|\vec{d}_A| \cdot |\vec{d}_B|} \\ &= \frac{\sum_{t \in V} w(t, d_A) \times w(t, d_B)}{\sqrt{\sum_{t \in V} w(t, d_A)^2} \sqrt{\sum_{t \in V} w(t, d_B)^2}}, \end{aligned}$$

where $w(t, d_A)$ is the weight of term t in d_A , and V denotes the vocabulary—all the words (or concepts) in the text collection. Evidence from various applications shows that the cosine rule is an effective measure of inter-document similarity (Willett, 1983; Rorvig, 1999; M. D. Lee et al., 2005).

Like many other measures, the cosine rule assumes that each feature is independent, which is tantamount to assuming that the dimensions of the vector space are orthogonal. This means that connections between features are not considered. Concepts and the semantic relations among them change this view, particularly when such relations become readily available and quantifiable.

Some studies attempt to take this into account by expanding a text’s representation with concepts that are closely related to ones already mentioned in the text (Bloehdorn and Hotho, 2004; Recupero, 2007), for example, using the generic and associative concept relations, so that when the same orthogonal measure is applied, texts with similar topics yet different vocabularies will eventually share some concepts. Texts that are already similar before the expansion are likely to be even more similar when these related concepts are added. However, this tends

to incur undesirable noise in the model by letting extraneous concepts slip into the representation. After all, the expansion is performed without considering the greater context: it has no information about which concepts, once added, will render the entire text collection more discriminative for clustering.

Other studies suggest performing such expansions with a particular context in mind—the pair of texts at hand whose similarity is requested (Hu et al., 2008). The expansion is then restricted to the concepts that are relevant to connecting these two texts, and concepts are chosen based on the content of both texts. This avoids adding extraneous concepts into the representation. Each text is expanded on the fly with respect to another text, which means that concept relations are accounted for every possible pair of texts and only relations that are relevant to clustering the current collection are considered. However, it is not clear how to decide which concepts are relevant to both texts, and how relevant they are—what their weights should be.

Both strategies enrich the similarity measure with concept relatedness. They provide a generic and comprehensible solution to the third and fourth problems of the bag-of-words model—inability to capture similarity beyond surface forms and insufficient robustness for handling new texts, which might contain previously unseen words or concepts. Although using term clusters and combinations (reviewed in Section 2.1.2) seems to have a similar ability to capture statistical relations between words, the approach is not generic because of its heavy dependence on the input data. Furthermore, the term clusters and combinations are difficult to comprehend.

2.3 Clustering techniques

The literature on clustering algorithms is massive (Willett, 1988; Jain et al., 1999), and a comprehensive survey would be a monumental task and out of the scope of this thesis. Considering that the clustering algorithms are not the focus of this research, this thesis uses several standard algorithms, which have been widely applied, especially in the context of concept-based text clustering. This section focuses on reviewing these methods.

Different clustering methods can be categorized in three ways (Hartigan, 1975;

Jain et al., 1999). The following list explains these categories and specializes them to apply to text:

- Agglomerative vs. divisive. The former begin by treating each text as a cluster and successively merge them until a stopping criterion is met (the bottom-up style); the latter begin by placing all texts in a single group and perform splitting until a stopping criterion is met (the top-down style).
- Hierarchical vs. partitional. This aspect relates to the structure of the clusters that are produced. The former algorithms form a hierarchy of clusters: clusters at lower levels are nested to upper level clusters. The latter produce a single flat partition.
- Hard vs. fuzzy. This aspect concerns cluster membership. The former methods allocate each text to a single cluster while the latter predict its degree of membership for multiple clusters. A fuzzy method can be converted to a hard one by assigning texts to the cluster that has the highest degree of membership.

Although clustering algorithms are not the focus of this thesis, they are important for a fair evaluation of the different representation models and similarity measures. In order to compare our concept-based text clustering methods with other people’s approaches, this thesis utilizes two popular clustering algorithms: the agglomerative hierarchical clustering algorithm and the partitional k -means algorithm. The rest of this section reviews them.

2.3.1 Hierarchical algorithms

Hierarchical clustering is usually done bottom-up, that is, in an agglomerative fashion. This operates in the following steps:

1. Treat each text as a cluster and compute the similarity between every pair of clusters.
2. Find the most similar pair of clusters, merge them into one and update the similarity of clusters that are involved this merge operation.

3. Terminate if the stopping criterion is met; otherwise go to step 2.

The stopping criterion used in this thesis is when the desired number of clusters is achieved.

There are a range of measures for calculating the similarity between two clusters, and this thesis uses four of them: single-link, complete-link, average-link and group-average-link. The single-link function measures the similarity between two clusters as the maximum similarity between its component objects (Sneath and Sokal, 1973), and the complete-link function takes the minimum similarity (King, 1967).

Average-link, which is also called UPGMA (Jain and Dubes, 1988), computes cluster similarity as the average of the pairwise similarities between the texts in each cluster. The group-average-link computes the quality of the merged cluster in terms of its average within-cluster similarity, and takes this as the similarity between two clusters (Manning et al., 2008).

Let $\Phi = \{\rho_1, \rho_2, \dots, \rho_k\}$ denote the set of clusters. The average-link function is formalized as:

$$\text{sim}(\rho_i, \rho_j) = \frac{1}{|\rho_i||\rho_j|} \sum_{d_A \in \rho_i} \sum_{d_B \in \rho_j} \text{sim}(d_A, d_B),$$

where $|\rho_i|$ is the size of ρ_i : the number of texts that ρ_i contains, and $\text{sim}(d_A, d_B)$ the similarity between documents d_A and d_B . The group-average-link calculates cluster similarity as

$$\text{sim}(\rho_i, \rho_j) = \frac{1}{(|\rho_i| + |\rho_j|)^2} \sum_{d_A \in \rho_i \cup \rho_j} \sum_{d_B \in \rho_i \cup \rho_j} \text{sim}(d_A, d_B),$$

that is, the average intra-cluster similarity if ρ_i and ρ_j were merged.

Hierarchical agglomerative clustering is a computationally expensive method, with the complexity about $O(N^2 \log(N))$ in time and $O(N^2)$ in space for clustering N documents (Jain et al., 1999). Nevertheless, its advantage is that clustering is performed in a deterministic way, whereas partitional methods are not deterministic, although they are usually more efficient. Furthermore, it produces a hierarchical structure of the texts. The inner structure of a cluster—which ob-

jects have been merged into this cluster and how—can be preserved during the clustering process, which is usually helpful for users to browse and comprehend the resulting clusters (Cutting et al., 1992).

2.3.2 Partitional algorithms

Partitional clustering methods find clusters by optimizing a certain objective function that defines the optimal solution (Hartigan, 1975). For example, the k -means algorithm minimizes the squared error in the resulting cluster structure, by assigning each point to its closest cluster in each iteration. Noting that exhaustive search through all possible partitions for the optimal solution is computationally prohibitive. It is common to approximate this by running the algorithms multiple times with different initialization, each time generating a different partition of the dataset, and then use the best clustering result.

Due to this approximation, partitional methods are usually efficient, with computational requirements ranging in the order of $O(N)$ to $O(N \log N)$ for clustering N documents (Willett, 1988). Therefore they are favoured for handling large data sets (Dhillon and Modha, 2001). The k -means algorithm is one of the most commonly used partitional clustering methods. It minimizes the squared error of cluster assignments (McQueen, 1967)—it finds a local minimum—and operates in the following steps:

1. Randomly choose k texts, each as a single-text cluster.
2. Assign each text in the collection to its closest cluster.
3. Recompute the cluster centroids based on the current cluster membership.
4. Terminate if the clustering has converged; otherwise go to step 2.

This iterative process converges when there is no change (or hardly any change) in cluster membership between two consecutive iterations, or when a pre-specified maximum number of iterations is reached. In our experiments, converge normally occurs within 20 iterations. The cluster centroid is the centre of the cluster, and is computed as the mean vector of all members of the cluster: the value of each dimension is the mean of that feature’s values in the cluster. Similarity between

a text and a cluster is calculated as the similarity between that text and the cluster’s centroid.

Other partitional algorithms include the expectation maximization algorithm for mixture models (Dempster et al., 1977; Witten et al., 2011) and spectral clustering methods (Ng et al., 2002). However, k -means and its variants such as bisec-kMeans (Steinbach et al., 2000) are the most popular partitional algorithms used in concept-based text clustering (Hotho et al., 2003; Hu et al., 2008; H.-T. Zheng et al., 2009). Thus this thesis uses the k -means algorithm as the representative of partitional clustering methods.

2.4 Evaluation measures

Evaluating the quality of a clustering result is an important yet difficult problem, simply because clustering is a subjective task. Even people have different views regarding how similar topics in different texts are (M. D. Lee et al., 2005) and how texts should be organized into groups (Macskassy et al., 1998), let alone computers. However, subjective evaluation is labour-intensive and expensive, especially for assessing clustering quality, which is a complicated task that involves several challenging judgements (see Section 1.2).

In practice, the difficulties of subjective evaluation are usually circumvented by evaluating against gold standards—existing categorization of texts, which are normally generated by human efforts or based on human judgements. Quality of clustering is measured in terms of its goodness of fit with respect to these categories, which can be quantified with mathematical measures. This section reviews several commonly used ones, each of which reflects a distinct perspective: *cluster purity*, *structural quality*, and *pairwise relations*.

We use $\Phi = \{\rho_1, \rho_2, \dots, \rho_k\}$ to denote the set of clusters and $\Omega = \{\omega_1, \omega_2, \dots, \omega_g\}$ to denote the set of categories in the collection—the gold standard. To compute these measures, each cluster is labelled with the category that is the most frequent one in that cluster. A text is correctly clustered if the cluster it is assigned to is labelled with the category it belongs to.

2.4.1 Cluster purity

There are two cluster purity measures: *purity* and *inverse purity*. *Purity* measures the percentage of texts that are accurately clustered. It is the weighted average of each cluster's purity, and is calculated by counting the number of correctly assigned texts and dividing by the total number of texts in the collection N :

$$\begin{aligned} Purity(\Phi, \Omega) &= \sum_{\rho_i} \frac{|\rho_i|}{N} \times \frac{\max_{\omega_j} |\rho_i \cap \omega_j|}{|\rho_i|} \\ &= \frac{1}{N} \sum_{\rho_i} \max_{\omega_j} |\rho_i \cap \omega_j|. \end{aligned}$$

Inverse purity measures the distribution of the categories in the clusters, by counting the number of texts that are assigned to the dominant cluster for each category:

$$\begin{aligned} InvPurity(\Phi, \Omega) &= \sum_{\omega_j} \frac{|\omega_j|}{N} \times \frac{\max_{\rho_i} |\rho_i \cap \omega_j|}{|\omega_j|} \\ &= \frac{1}{N} \sum_{\omega_j} \max_{\rho_i} |\rho_i \cap \omega_j|. \end{aligned}$$

These two measures resemble *precision* and *recall* in information retrieval. Purity can be viewed as the weighted precision of all clusters, and inverse purity as the weighted recall of all categories. Both are normalized between 0 and 1, and generally the higher the better. Both achieve the optimal value of 1 when all texts are correctly clustered. However, purity also achieves 1 if each text forms a cluster (i.e., $k = N$), and for inverse purity this happens if all texts are assigned to one group (i.e., $k = 1$). Therefore for a fair evaluation these two measures usually are used together.

2.4.2 Structural quality

The *normalized mutual information* (NMI) measure is independent of the number of clusters (Manning et al., 2008), and is calculated as:

$$NMI(\Phi, \Omega) = \frac{I(\Phi; \Omega)}{[H(\Phi) + H(\Omega)]/2},$$

where I is the mutual information between the set of clusters and the set of categories. Formally:

$$\begin{aligned} I(\Phi; \Omega) &= \sum_{\rho_i} \sum_{\omega_j} P(\rho_i \cap \omega_j) \log \frac{P(\rho_i \cap \omega_j)}{P(\rho_i)P(\omega_j)} \\ &= \sum_{\rho_i} \sum_{\omega_j} \frac{|\rho_i \cap \omega_j|}{N} \log \frac{N|\rho_i \cap \omega_j|}{|\rho_i||\omega_j|}, \end{aligned}$$

where $P(\rho_i)$, $P(\omega_j)$ and $P(\rho_i \cap \omega_j)$ are the probabilities of a text being in cluster ρ_i , category ω_j , and the intersection of ρ_i and ω_j respectively. H is the entropy, which is defined as:

$$\begin{aligned} H(\Phi) &= - \sum_{\rho_i} P(\rho_i) \log P(\rho_i) \\ &= - \sum_{\rho_i} \frac{|\rho_i|}{N} \log \frac{|\rho_i|}{N}, \end{aligned}$$

and the same for $H(\Omega)$.

The numerator—the mutual information between clusters and categories—measures the extent to which the information about the categories increases when the clusters are known, and a value of 0 indicates that the clustering is random with respect to the categories. However, it suffers from the same problem as purity: a clustering with N one-text clusters yields the maximum mutual information value of 1. Adding the denominator solves this problem, because entropy tends to increase with the number of clusters, and normalizes the measure between 0 and 1. Thus normalized mutual information can be used by itself to measure the overall structural quality of a clustering result in terms of its fitness

with respect to the categories.

2.4.3 Pairwise relations

FMeasure, borrowed from information retrieval (van Rijsbergen, 1979; Manning et al., 2008), measures the accuracy of decisions concerning pairwise relations, and is also known as *pairwise FMeasure*. *Precision* (P) is calculated by counting the number of correct decisions—texts belonging to the same categories being assigned to the same cluster—divided by the number of assignments, that is, the number of text pairs that share the same cluster membership. *Recall* (R) is the proportion of pairs sharing the same category membership that are assigned to the same cluster. This gives the following contingency table.

	Same cluster	Different clusters
Same category	TP (True positive)	FN (False negative)
Different categories	FP (False positive)	TN (True negative)

Precision and recall are then calculated as:

$$P = \frac{TP}{TP + FP} \qquad R = \frac{TP}{TP + FN},$$

and the FMeasure is

$$F_\beta = \frac{(\beta^2 + 1)PR}{\beta^2 P + R}.$$

This thesis sets $\beta = 1$ to weight precision and recall equally. $\beta < 1$ emphasizes precision over recall, that is, when grouping dissimilar texts into the same cluster is a more severe mistake than separating similar texts into different clusters. Conversely, $\beta > 1$ stresses recall over precision.

2.5 Summary

This chapter analyzes the components of text clustering and discusses the ones that are most relevant to this thesis—text representations, similarity measures, clustering methods and evaluation measures. We have not reviewed research

on using concepts for cluster descriptions, because this thesis does not involve presenting clusters to users. Yet concepts have great potential as descriptors of text clusters, especially considering that the relatedness between concepts can also be visualized, which can be valuable for information retrieval systems (Milne et al., 2007).

Our survey shows that employing concept systems to facilitate text clustering has drawn increasing attention in recent years. Most work focuses on feature generation, to create a representation model that is informative, compact and efficient. Yet a clear understanding of the requirements for concept systems to be effective for the task is still missing, and research on enriching document similarity measures with concept relatedness is also very limited. This thesis investigates these problems, and extends the power of knowledge to benefit clustering beyond feature generation.

3

Experimental method

As the previous chapter shows, there are many ways to perform clustering and to use concepts in the process. This chapter further defines the scope of the thesis by explaining the experimental datasets, the clustering methods and the evaluation methodology.

3.1 Experimental datasets

Our major concern is how effective concept-based text representation models and semantically enriched clustering techniques are in recovering the inherent thematic structure of a text collection. Therefore the evaluations use standard corpora whose thematic components are already labelled.

We created four datasets from three standard text collections, Reuters-21578, 20Newsgroups and OHSUMed. Each dataset has its distinct properties, and covers different domains, diverse topics and difficulty levels. The first two collections contain general topics, whereas the last one is specific to the medical domain. Table 3.1 shows summary statistics for the four datasets, and Appendix C provides more detailed information like category distribution and document length. Appendix A exemplifies their distinct characteristics by drawing one document from each dataset and showing the words and concepts identified from them.

The Reuters-21578 collection¹ consists of short news articles that appeared on the Reuters newswire in 1987. It has 11,367 documents, manually labelled with

¹Available at <http://kdd.ics.uci.edu/databases/reuters21578/reuters21578.html>.

Dataset	Number of Categories	Number of Documents	Average size of categories
SmallReuters	30	1658	55.3
Med100	23	2256	98.1
NewsSim3	3	2938	979.3
NewsDiff3	3	2780	926.7

Table 3.1: Statistics of the four experimental datasets

one or more of 82 categories. 9494 documents are uniquely labelled, i.e., assigned to only one category. Following Hu et al. (2008)’s work, filtering categories with less than 15 or more than 200 documents resulted in a subset with 30 categories comprising 1658 documents, which is called the SmallReuters dataset. It has the largest number of categories yet its documents are the shortest (see Appendix C).

The 20Newsgroups dataset² is a collection of 19,997 documents gathered in 1995 from 20 Usenet newsgroups, with an approximately equal number of documents in each group. Documents are newsgroup posts, and are categorized by the newsgroup they appear in. Each document consists of a subject line followed by a message body. Some of the newsgroups discuss similar topics, such as *comp.windows.x* and *comp.os.ms-windows.misc*. In order to understand whether topic separation impacts the performance of our methods, we created two subsets. One has closely related topics—the *comp.windows.x*, *comp.graphics* and *comp.os.ms-windows.misc* categories—and is called NewsSim3; the other, called NewsDiff3, contains three very different topics: *sci.space*, *alt.atheism* and *rec.sport.baseball*. Intuitively, NewsSim3 is more difficult to cluster, considering the subtle distinction between the topics it contains.

The OHSUMed collection³ is domain specific, consisting of 348,566 records of medical papers from the MEDLINE database published between 1987 and 1991, about two-thirds of which (233,445) have both a title and an abstract. Each document is a concatenation of its title and abstract (if any). The collection is not categorized, but every document has a list of medical subject headings (MeSH) assigned by human indexers. One commonly used version (Moschitti

²Available at <http://people.csail.mit.edu/jrennie/20Newsgroups/>

³The original dataset is available at <ftp://medir.ohsu.edu/pub/ohsumed>.

and Basili, 2004)⁴ categorizes the 50,216 documents published in 1991 that have an abstract into one or more of the 23 *cardiovascular diseases* categories in the MeSH thesaurus. A document belongs to a category if it is labelled with at least one MeSH term from that category, giving a total of 34,389 documents related to *cardiovascular diseases*. Filtering out those appearing in multiple categories reduces this to 18,302 documents, as in Hu et al. (2008)’s work. Restricting the maximum size of each category to 100 documents with random sampling produced the Med100 subset (every category, except one, contains more than 100 documents, see Appendix C). It is the only domain-specific dataset, which allows us to investigate the extent to which general concept systems such as WordNet and Wikipedia can provide useful information for analyzing texts from a particular topic domain.

3.2 Clustering methods

Both the hierarchical agglomerative clustering and k -means algorithms are used to create text clusters, so as to compare with related work (see Section 4.4). For the former, the four functions for computing the similarity between two clusters—single-link, complete-link, average-link and group-average-link (see Section 2.3.1)—are tested, and the stopping criterion is when the specified number of clusters is achieved. For the latter, the cluster centroid is the mean vector of all texts belonging to that cluster (see Section 2.3.2), and the iterative process converges when there is no change in cluster membership between two consecutive iterations, or when a pre-specified maximum number of iterations is reached. In our experiments, convergence normally occurs within 20 iterations.

For k -means, clustering results are always reported based on five independent runs, each using different random documents as the initial seed clusters. For hierarchical clustering algorithms, which are deterministic, only one run is needed for each test.

The cosine measure for document similarity (see Section 2.2) is used for both clustering algorithms. Each document is represented by a weighted vector, using the vector space model. Each dimension corresponds to a word or concept, and the

⁴Available at <http://disi.unitn.it/moschitti/corpora.htm>

associated weight is the *term frequency* \times *inverse document frequency* ($tf \times idf$) weight of that word or concept:

$$tf \times idf(t, d) = tf(t, d) \times \log \frac{N}{df(t)},$$

where $tf(t, d)$ is the number of occurrences of term t in document d , $df(t)$ is the number of documents that mention term t , and N is the total number of documents in the collection. For hybrid representations (see Section 4.5), $tf \times idf$ weighting is performed after merging the features.

3.3 Evaluation methodology

A central claim of this thesis is that concepts are more informative than words as text descriptors for clustering texts by their topics. Hence the bag-of-words model is used as a baseline in all evaluations throughout the thesis.

Cluster quality is assessed by the four evaluation measures from Section 2.4: purity, inverse purity, normalized mutual information and FMeasure, and against the existing category structure in each dataset. To compute these measures, each cluster is labelled with the category that is the most frequent one in that cluster. A text is correctly clustered if the cluster it is assigned to is labelled with the category it belongs to.

Statistical significance is established using paired t -tests with confidence level $p = 0.05$ throughout this thesis. For k -means, the significance test is performed based on results from different runs. For hierarchical clustering, results are obtained by varying the number of clusters in increments of 5, from 5 to 60, producing 12 results for performing the test. In cases that involve comparing k -means with hierarchical clustering algorithms—there is only one such case (in Section 4.4.5)—the sample comprises clustering results obtained with varied number of clusters; and for the k -means algorithm, its average performance over five runs for each number of clusters is used.

4

Concept-based document representation

Chapter 1 identifies two problems for concept-based text clustering. This chapter explores solutions to the first task: how to represent a text by the concepts it mentions instead of by words, and investigates the following hypothesis:

representing text by concepts provides a more effective basis for text clustering than the traditional bag-of-words model.

Effectiveness is measured by the quality of the resulting clusters: the better the clusters, the more effective the representation.

The purpose of this chapter is to establish the foundation of this thesis by exploring ways to employ WordNet and Wikipedia for identifying concepts from running text, and systematically evaluating their effectiveness in clustering. We start with WordNet as the easier case. Sections 4.1 and 4.2 analyze WordNet and Wikipedia as the source of concepts and introduce methods for mapping terms in running text to them. Section 4.3 describes experimental design, which involves two aspects: comparing representation models across different clustering algorithms, to clarify whether some are consistently more effective than others; and comparing clustering algorithms across different models, to identify the most effective clustering techniques for concept-based text clustering. Section 4.4 presents and discusses results from these evaluations. Considering that Wikipedia and WordNet are two distinct resources, Section 4.5 investigates whether combining information from both is better than clustering with each resource individually.

4.1 Identifying WordNet concepts from text

WordNet is a lexical database of the English language. It contains common English words, and concepts take the form of synonym sets (*synset* in WordNet terminology): a group of interchangeable words and phrases that have the same meaning. For example, the synset for *a machine for performing calculations automatically* contains six synonyms, including *computer*, *computing device* and *data processor*. Synsets are the basic building blocks of WordNet, and the current version of WordNet (3.0) has about 118,000 of them (WordNet, 2011).

Terms in WordNet are associated with the synsets they appear in. A term can belong to one or more synsets, depending on how many distinct meanings it possesses. For example, the term *computer* belongs to two synsets: the one above, and another representing a much rarer meaning—*an expert at calculation or at operating calculating machines*. Terms with only one meaning are called *monosemous*, while ones with multiple meanings are called *polysemous*. The current version of WordNet has about 155,000 terms, and about 20% of them are polysemous (WordNet, 2011).

Indexing a text with WordNet concepts can be broken down into two steps: first identify candidate concepts by mapping terms in the text to terms in WordNet; then disambiguate ambiguous terms and identify their intended meanings—determining the correct concepts (i.e., synsets). Section 4.1.1 discusses the considerations involved in the first step, and Section 4.1.2 analyzes sense disambiguation methods for WordNet.

4.1.1 Identifying candidate concepts

Mapping terms in free text to WordNet terms introduces two issues: handling morphological variations in these terms and identifying their lexical categories. WordNet provides built-in functions for unifying morphological variations that map inflectional variations, including pluralities and tenses, to their corresponding base forms. In contrast, derivational variations are kept as distinct terms from their base forms, because derivational changes usually alter a term’s meaning. For example, *happy* and *happiness* are recognized as distinct WordNet terms.

The lexical category of a term is commonly known as its part of speech (denoted by POS). WordNet organizes concepts by their parts of speech, associating each concept with one of four parts of speech—*noun*, *verb*, *adjective* and *adverb*. Meanings of a polysemous term are also grouped by their corresponding POS. For example, the term *tax* can be either a noun or a verb, the former being unambiguous with the single meaning *charge against a citizen's person or property or activity for the support of government*, and the latter with four meanings, including *levying a tax on* and *using something to the limit*.

It is arguable whether POS tagging is necessary for connecting texts to WordNet concepts. Both the candidate selection and the subsequent sense disambiguation tasks can be accomplished without it. However, without POS information, accuracy of both tasks can be impacted and efficiency will be sacrificed.

For example the word *rose* corresponds to two WordNet terms—*rose* as a noun or an adjective and *rise* as a verb—and 21 concepts corresponding to the three, one and seventeen meanings for the noun, adjective and verb respectively. Without knowing its POS, all 21 possible meanings must be assessed by the sense disambiguation algorithm. This incurs considerable computational overhead, because context-based disambiguation process—as the next section shows—involves pairwise concept comparisons, whose number grows with the number of items that need to be compared. In contrast, knowing the POS restricts the search of the intended sense to a much smaller space and avoids unnecessary comparisons. In this case, disambiguation can be avoided altogether if *rose* is known to be an adjective.

4.1.2 Sense disambiguation

Although polysemous terms only count 20% of all WordNet terms, this does not reduce the importance of the sense disambiguation task at all. Common expressions tend to be polysemous: the more frequently used a term is, the more likely it is to be polysemous (Fellbaum, 1998). Polysemous terms need to be disambiguated and resolved to their intended meanings. This section describes two such methods: the most-common-sense rule for disambiguation and disambiguation based on context.

As the name implies, the most-common-sense rule always chooses the most common meaning of an ambiguous term. The intuition is that the most common meaning is also the most likely meaning, regardless of the term’s surrounding context. In WordNet, concepts (i.e., meanings) associated with an ambiguous term are ranked in descending order of their *commonness*, which is derived from word frequencies obtained from a large standard English corpus, the British National Corpus. Thus the most common rule always takes the first concept in the ranked list. Although simple, this heuristic has been shown to be very effective (Hotho et al., 2003; Varelas et al., 2005).

The second method—context-based disambiguation—is also commonly used in the literature (Hirst and St-Onge, 1997; Medelyan et al., 2008). It assesses how each possible meaning of an ambiguous term fits its surrounding context, and chooses the best fit as the intended meaning. The unambiguous terms co-occurring in the context provide useful information about the context. For example, if the word *pluto* co-occurs with terms like *Disney*, which is unambiguous, it is more likely to mean the cartoon character than the god in Greek mythology, while the opposite is true if the context contains terms like *Poseidon*, which is also unambiguous. Therefore the more closely a candidate meaning relates to these unambiguous terms—i.e. context concepts—the better it fits the context thus the more likely it is the intended one.

To indicate that *pluto* the cartoon character is more related to *Disney* than *Poseidon* and vice versa for *pluto* the Greek god requires a measure of concept relatedness. The literature describes extensive work on semantic relatedness measures for WordNet concepts (Resnik, 1995; Leacock and Chodorow, 1997; Jiang and Conrath, 1997; Lin, 1998). We use the path length based measure of Leacock and Chodorow (1997) (denoted by LCH henceforth), because it has been shown to be more accurate and more consistent with human judgement of concept relatedness than other measures (Varelas et al., 2005; Strube and Ponzetto, 2006).

The LCH measure utilizes WordNet’s concept taxonomies. WordNet organizes concepts according to the hierarchical relations among them—the generic *is-a* and the partitive *part-of* relations—and forms taxonomies, a small part of which is illustrated in Figure 4.1. Higher concepts are more generic, and are hypernyms

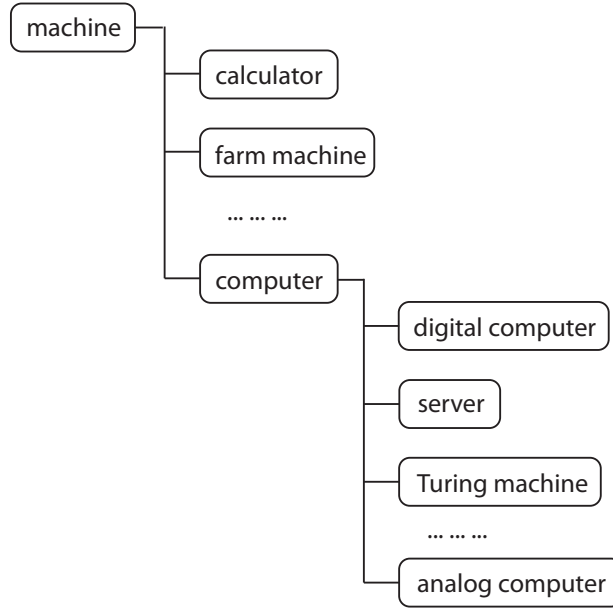


Figure 4.1: Fragment of WordNet’s concept taxonomy

of their descendants. The LCH measure is based on the edge counting method of Rada et al. (1989), which defines semantic concept distance as the number of nodes in the taxonomy along the shortest path between two concepts. Leacock and Chodorow refine it by introducing the depth of the taxonomy. Formally, given two concepts A and B , the relatedness between them is computed as

$$LCH(A, B) = -\log \frac{length(A, B)}{2D},$$

where $length(A, B)$ is the number of nodes along the shortest path between them and D is the maximum depth of the taxonomy.

To disambiguate an ambiguous term, each possible meaning is assessed by measuring its relatedness with the unambiguous context concepts in the document and the one with the highest overall relatedness is chosen as the intended meaning. When there are no unambiguous terms in a text—no context concepts—the disambiguation process reverts to the most-common-sense rule.

4.2 Identifying Wikipedia concepts from text

A WordNet synset is both a conceptual unit expressing a unique meaning and an assembled group of synonyms associated with that meaning. These two aspects are realized individually in Wikipedia. First, the basic conceptual unit is a Wikipedia article: a page dedicated to a specific concept that provides a detailed and well structured explanation of it. For example, the Wikipedia article *computer* succinctly describes the history of computing, the basic components of a computer, and a variety of other related topics.

Another type of Wikipedia page—the redirect page—groups synonyms together. These provide alternative names for the target article, including synonyms, acronyms and common variant spellings. For example, the article *computer* has 30 redirects, including synonyms like *computer system* and *computing device*, morphological variations like *computers*, and even common spelling mistakes like *computor* and *camputer*. Wikipedia contains about 2.7 million articles, 45% of which have at least one redirect, with about 1.2 redirects per article on average.

The two-step procedure from the previous section—first identifying candidate concepts and then disambiguating the one representing the intended sense—also applies to Wikipedia. Various methods have been explored to tackle each step. This section first describes the process of selecting Wikipedia articles for a particular text in Section 4.2.1. Sections 4.2.2 and 4.2.3 discuss the sense disambiguation algorithms for Wikipedia concepts.

4.2.1 Identifying candidate concepts

We have discussed several methods for mapping terms in free text to Wikipedia articles in Section 2.1.3, which are briefly reviewed here. Earlier methods match terms against the title of the article (Strube and Ponzetto, 2006) and articles with a positive match are associated with the text. The consequence of restricting to article titles is that only surface matches are counted and variations in expressions are ignored.

Gabrilovich and Markovitch (2007) select Wikipedia articles based on the overlap between their full text and the content of the text. This is less restricted than

the strict string match method, but more expensive computationally, because it involves full text level analysis.

Mihalcea and Csomai (2007) and Milne and Witten (2008b) exploit another resource, the anchor text in Wikipedia pages, as an intermediary for connecting words and phrases in free text to Wikipedia articles. Anchor texts are short phrases that articles use when referring to other articles. They provide a large number of alternative expressions for the article being pointed to. For example, *CPU cache* is referred to as *cache memory* in the article *computer*, and as *cache* in the article *binary search tree*, which are only two of the 97 expressions used in Wikipedia when mentioning this concept. Thus, mentions of a Wikipedia concept in free text can be recognized by matching with the anchor texts that have been used in Wikipedia when referring to the concept.

The anchor-text-based approach for linking Wikipedia concepts to text has several advantages over strict title matching and the text overlap method. It is more flexible than the former and more efficient than the latter, because full text parsing and matching is avoided. Furthermore, it avoids indexing text with millions of Wikipedia concepts and the truncation required for the latter. We use Milne and Witten’s method of using Wikipedia’s anchor text vocabulary to connect a text to related Wikipedia articles, and review it here.

Wikipedia’s anchor text vocabulary is huge: there are about 4.5 million distinct phrases. The consequence is that most texts contain many phrases that can be matched with entries in the anchor text vocabulary. However, not every match is equally useful, and keeping all matches does not necessarily contribute to a more effective representation of the text. Particularly in text clustering, discriminative capacity is much more important than exhaustiveness.

Therefore, anchor phrases are weighted by their likelihood of being good text descriptors. This can be derived from Wikipedia, based on the intuition that a common expression of a concept will appear frequently as one of its anchors. Each phrase p in the anchor text vocabulary can be weighted by taking the number of Wikipedia articles in which it appears as a link (denoted L) and dividing it by the number of articles in which that text appears (denoted as T), whether linked

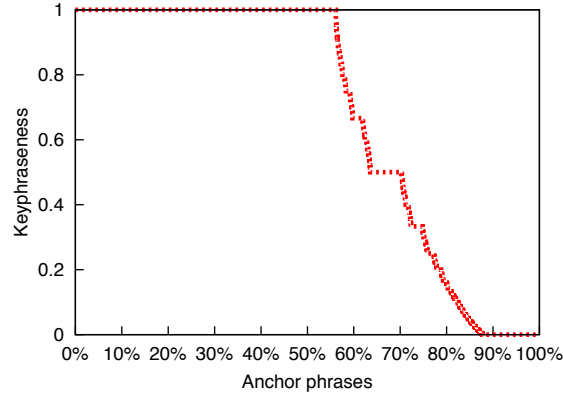


Figure 4.2: Distribution of the keyphraseness of Wikipedia anchor phrases

or not, as follows:

$$\text{keyphraseness}(p) = \frac{L}{T}.$$

This is called the *keyphraseness* of an anchor phrase (Mihalcea and Csomai, 2007; Medelyan et al., 2008). Phrases with high keyphraseness are more likely to be valid descriptors. For example, although a common word such as *the* might occur frequently as an anchor to the concept *Article (grammar)*, its keyphraseness will be close to zero due to its ubiquitous occurrence in Wikipedia.

Figure 4.2 plots the distribution of the keyphraseness of anchor text in Wikipedia against the 4.5 million distinct phrases, sorted in descend order of keyphraseness. The distribution’s right tail shows that about 10% of the phrases have a keyphraseness close to or equals zero. These phrases are either rare expressions, or common expressions that can be considered as stopwords such as *the*. They are more likely to introduce noise into the clustering process and thus should not be matched with. Therefore anchor phrases are filtered by their keyphraseness, and those with a keyphraseness below a specified threshold are discarded. In this thesis this threshold is set to 0.03 unless otherwise specified, which was selected though preliminary experiments of parameter tuning.

The articles that the valid matching anchor phrases point to form the set of candidates for representing a text. The next problem is to determine articles that correspond to the intended meaning of each phrase—the sense disambiguation

problem. The next two sections discuss the algorithms for this task.

4.2.2 Sense disambiguation: unsupervised vs. supervised

Just as terms can have multiple meanings, the same text can be used to connect different Wikipedia articles in different contexts. About 9.6% of the 4.5 million anchor text in Wikipedia are ambiguous: each has been used to refer to more than one article. The small proportion might make the disambiguation problem seem trivial, but based on occurrences they account for 57% of all anchors in Wikipedia. Thus the problem of sense disambiguation still exists for anchor text.

For WordNet, sense disambiguation is conducted in an unsupervised way. However, when labelled data is available, supervised methods usually are more effective and accurate than unsupervised ones. Both Mihalcea and Csomai (2007) and Milne and Witten (2008b) use supervised machine learning techniques to perform disambiguation with Wikipedia concepts and confirm that they outperform unsupervised heuristics.

The main reason for preferring unsupervised methods over supervised ones is that supervised methods require labelled training data, which is usually difficult and expensive to obtain. However, the enormous number of anchors in Wikipedia provides a copious amount of labelled data, using which supervised methods can learn to disambiguate more accurately than unsupervised approaches. The target of every inter-article link in Wikipedia specifies the intended sense of the anchor, thereby creating a positive example given the current context, and all other possible senses for the anchor—other articles that it has been used to link to—provide negative examples.

For example, the Wikipedia article *binary search tree*, which points to the article *CPU cache* using *cache*, provides a positive example for the *CPU cache* sense of the phrase *cache*. *Cache* has 10 possible senses (i.e., destination articles) in Wikipedia, such as the more general concept of *computer cache* that includes both CPU and disk caches, and the *Cache county in the U.S. state of Utah*. The other nine possible targets of *cache* form negative examples with respect to this context.

Mihalcea and Csomai (2007) and Milne and Witten (2008b) design different

features to characterize the association between an anchor phrase, its surrounding context and its target article. Each example from Wikipedia is represented by these features and a classifier such as a naive Bayes model or a decision tree is trained with them. When it comes to disambiguating a term in free text, the same features are calculated for each sense of the term, and the trained classifier predicts the best one—the most likely target article, i.e., concept.

Features used by Mihalcea and Csomai include the ambiguous word itself, its part of speech, its directly neighbouring words and their parts of speech, and a feature indicating its global context. Milne and Witten avoid natural language processing by using three features: the prior probability of a sense being the intended one for the given phrase, the average relatedness of the sense to the phrase’s surrounding context, and the overall quality of the context. Here, context is represented by the set of unambiguous anchor phrases in it: phrases that only point to a single article throughout the entire Wikipedia.

Although using fewer features, Milne and Witten’s method achieves competitive performance with Mihalcea and Csomai’s approach and it is more efficient because it avoids natural language processing. Therefore, it is used in this thesis for disambiguating Wikipedia concepts, and reviewed in more detail in the next section.

4.2.3 Milne and Witten’s sense disambiguation algorithm

The prior probability of a sense is defined as the number of times it is used as the destination of an anchor divided by the total number of occurrences of this anchor in Wikipedia. For example, out of the 1078 occurrences of *pluto* as an anchor, it refers to the dwarf planet 742 times and to the Disney cartoon character 126 times, resulting in a prior probability of 68.8% and 11.7% for each sense respectively. This resembles the *commonness* of WordNet concepts: more common senses are expected to have higher prior probability.

The other two features—average relatedness to the context and the context’s quality—require calculating the relatedness between Wikipedia concepts. Research on semantic relatedness measures for Wikipedia concepts is extensive and ongoing (Strube and Ponzetto, 2006; Gabrilovich and Markovitch, 2007; Milne

and Witten, 2008a). We leave the detailed discussion of these measures to the next chapter, and here describe only Milne and Witten (2008a)’s measure WLM, which is used in this thesis.

The relatedness measure

WLM measures the relatedness between two concepts based on their hyperlink structure: incoming links made to the articles and outgoing links extending out from them. Because each type has a distinct distribution in Wikipedia, they are modelled separately.

Given two articles A and B , let A_{out} and B_{out} denote the sets of hyperlinks found within them, and A_{in} and B_{in} denote the sets of hyperlinks that are made to them. The first component computes relatedness based on A_{out} and B_{out} , using the cosine measure:

$$WLM_{out}(A, B) = \frac{\sum_{l \in A_{out} \cup B_{out}} w(l, A) \times w(l, B)}{\sqrt{\sum_{l \in A_{out} \cup B_{out}} w(l, A)^2} \times \sqrt{\sum_{l \in A_{out} \cup B_{out}} w(l, B)^2}}.$$

Here $w(l, A)$ is the weight of a link l with respect to article A , which is 0 if $l \notin A_{out}$ and $\log(\frac{|W|}{|T|})$ otherwise, where $|W|$ is the total number of articles in Wikipedia and $|T|$ is the total number of articles that link to the target of l . This resembles the inverse document frequency weighting (see Section 3.2).

Incoming links are modelled after the *normalized Google distance* (Cilibrasi and Vitányi, 2007), formally

$$WLM_{in}(A, B) = 1 - \frac{\max(\log |A_{in}|, \log |B_{in}|) - \log |A_{in} \cap B_{in}|}{\log |W| - \min(\log |A_{in}|, \log |B_{in}|)},$$

where $A_{in} \cap B_{in}$ denotes the set of hyperlinks that link to both A and B . WLM computes the overall relatedness between A and B as the average of these two components.

This measure is used to quantify the relatedness between each candidate sense and the context. Furthermore, each context concept is weighted by the average of

its keyphraseness and its overall relatedness to the other context concepts. The third feature describes the quality of the context. The context quality feature is calculated as the sum of each context concept’s weight. This describes the overall cohesiveness of the context.

The disambiguation process

The disambiguation process starts by identifying the unambiguous anchor phrases in a text, which forms the context for disambiguation. A new example consisting of the above three features—prior probability, relatedness to the context and the context’s quality—is then created for each candidate sense of an ambiguous term. The sense classifier takes in each example and predicts the probability of the corresponding sense being the intended one given the feature values. The sense with the highest probability is selected.

The original purpose of the methods in Mihalcea and Csomai (2007) and Milne and Witten (2008b) is to automatically detect phrases in free text that are worth linking to a Wikipedia article, so as to help readers further understand the current text. This is known as the *wikification* task. The overall process of wikification differs from creating a concept-based text representation, due to the different nature of the tasks. For instance, wikification usually requires filtering concepts so that only *interesting* ones will be linked, which is not necessary when creating a text representation. Yet the idea of utilizing Wikipedia’s anchor text as an indexing vocabulary for mapping document terms to Wikipedia articles, and the idea of learning sense disambiguation from Wikipedia, are both applicable here.

4.3 Experimental design

Any quantitative evaluation of a text representation scheme must be coupled with a particular target application. This thesis focuses on text clustering, and so the different representation schemes are evaluated in terms of their effectiveness in creating coherent text clusters. A clustering outcome with higher cluster quality indicates a more effective representation, other things being equal.

Algorithms for clustering are not the focus of this thesis, yet different algo-

rithms have different behaviours that might impact the performance of a representation scheme. Most research on using concepts in text clustering adopts the k -means algorithm (Hotho et al., 2003; Hu et al., 2008). It is less clear whether the results hold for other clustering algorithms and how they compare to each other. Consequently, we test another extensively used clustering method, hierarchical agglomerative clustering with four common linkage functions: single-link, average-link, complete-link and group-average-link (see Section 2.3.1).

The four datasets described in Section 3.1 are used. All documents are converted to lower case before converting to words or mapping to concepts. In the bag-of-words model, words are alphabetic sequences stemmed with Porter’s stemmer (Porter, 1980). Stopwords are removed using the stopwords list in the Weka software, which contains 526 stopwords,¹ as are words appearing less than five times in a dataset, which gives the best baseline results for clustering using the bag-of-words representation on most datasets.

The most recent version of WordNet 3.0 is used, which has 117,659 synsets and 155,287 distinct terms (WordNet, 2011). Documents are first segmented into sentences, and words are tagged with their parts of speech using the maximum-entropy-based tagger from the OpenNLP package.² Instead of using a stemmer, WordNet’s morphology functions are used to handle morphological variations, because it has been shown to be more effective than using Porter’s stemmer for clustering (Hotho et al., 2003). Two sense disambiguation techniques are tested: the most-common-sense baseline and context-based disambiguation.

The snapshot of Wikipedia used in this thesis contains 2.7 million articles or concepts. Hyperlinks and anchor texts that are associated with each article are summarized using the WikipediaMiner toolkit (Milne and Witten, in press). It has an indexing vocabulary of about 4.5 million anchor terms after lower casing. Restricting the minimum likelihood of a term being a valid expression of the target concept to 0.03 reduces this to about four million. Terms are disambiguated using the supervised disambiguation algorithm described in Section 4.2.3, and the best model from Milne and Witten (2008b)’s paper, which uses bagging (Breiman,

¹This list is based on that in Rainbow (available at <http://www.cs.cmu.edu/~mccallum/bow/rainbow/>).

²Apache OpenNLP (Version 1.5), available at <http://incubator.apache.org/opennlp/>.

1996) of C4.5 decision trees (Quinlan, 1993) and is trained with 100 randomly sampled Wikipedia articles.

Documents are represented using the vector space model (VSM), with $tf \times idf$ weighting (see Section 3.2). Similarity between two documents is measured by the cosine measure (see Section 2.2). Both k -means and hierarchical agglomerative clustering require a pre-specified number of clusters, which we set to the number of existing categories in each dataset, unless otherwise specified, which gives the best performance in our informal investigation. As discussed in Section 2.3.2, the k -means algorithm generates different clusters when initialized differently. The results reported in this thesis are averaged over five independent runs, and each run is initialized with a different set of randomly chosen seed documents.

4.4 Experimental results

This section presents the experimental results. We first discuss the advantages and disadvantages of each representation model with an example document. Then we compare the dimensionality of the models, and analyze their relative effectiveness in text clustering. Finally, we investigate the comparative performance of different clustering algorithms and identify which ones are more effective for concept-based text clustering.

4.4.1 An example of representations

This section draws an example document from the Reuters collection to illustrate the differences between representation schemes. Table 4.1 compares words and concepts that are frequently mentioned in Reuters document #15264, which discusses ongoing attempts by Teck Cominco—a Canadian mining company—to begin a joint copper-mining venture in Highland Valley, British Columbia. WordNet concepts are sets of synonyms, and the subscripts of each WordNet term denote its part of speech—with 1 meaning nouns, 2 verbs, 3 adjectives and 4 adverbs—and commonness rank of the sense (see Appendix A). For example, the first synset in Table 4.1 refers to the most common sense of *negotiation* as a noun (denoted by `negotiation11`): *a discussion intended to produce an agreement*, which

4.4. EXPERIMENTAL RESULTS

Document	TECK STILL IN TALKS ON B.C. COPPER VENTURE Teck Corp said it was continuing talks about joining a joint copper venture at Highland Valley, British Columbia, held by affiliates Cominco Ltd CLT and Lornex Mining Corp, but did not know when negotiations would be completed. Teck vice-president of administration John Guminski said in reply to a query that the talks had been “ongoing for a long time.” He declined to speculate on the outcome. Cominco, 29.5 pct owned by a consortium led by Teck, is optimistic that the talks will soon be concluded, spokesman Don Townson told Reuters. “I think all partners are hopeful that the situation will be resolved,” Cominco’s Townson said. “We’re optimistic that they will be concluded shortly,” he added. Townson declined to specify when the talks might end. Cominco and Teck’s 22 pct-owned Lornex agreed in January 1986 to form the joint venture, merging their Highland Valley copper operations. Cominco and Lornex share equally in control and management of the Highland Valley operations, while Cominco has a 55 pct share of production and Lornex receives 45 pct. For the six months following July 1, 1986, when the venture officially started production, Highland Valley had total ore milled of 22.6 mln short tons, grading an average of 0.41 pct copper, Townson said. Cominco’s share of production was 43,000 short tons of copper contained in concentrate, 1,200 short tons of Molybdenum in concentrate, 340,000 ounces of silver and 800 ounces of gold, he said. A consortium, 50 pct owned by Teck and 25 pct each by MIM (Canada) Inc and Metallgesellschaft Canada Ltd, acquired its Cominco stake last year from Canadian Pacific Ltd CP.				
Features (total number)	Most frequent 20 features				
Words (77)	cominco (8)	pct (7)	teck (6)	copper (5)	talk (5)
	vallei (4)	ventur (4)	product (3)	share (3)	short (3)
	ton (3)	canada (2)	concentr (2)	conclud (2)	consortium (2)
	corp (2)	declin (2)	joint (2)	oper (2)	silver (1)
WordNet concepts (83)	{negotiation ₁₁ , dialogue ₁₄ , talks ₁₆ } (6) {copper ₁₁ , Cu ₁₁ , atomic_number_29 ₁₁ } (5) {percentage ₁₁ , percent ₁₁ , per_centum ₁₁ , pct ₁₁ } (5) {valley ₁₁ , vale ₁₁ } (4) {venture ₁₁ } (4) {share ₁₁ , portion ₁₄ , part ₁₈ , percentage ₁₂ } (3) {production ₁₁ } (3) {short ₃₁ } (3) {short_ton ₁₁ , ton ₁₁ , net_ton ₁₁ } (3) {Canada ₁₁ } (2) {joint ₃₁ } (2) {optimistic ₃₁ } (2) {upland ₃₁ , highland ₃₁ } (2) {worsen ₂₁ , decline ₂₁ } (2) {operations ₁₂ , trading_operations ₁₁ } (2) {own ₂₁ , have ₂₄ , possess ₁₂ } (2) {consortium ₁₁ , pool ₁₄ , syndicate ₁₂ } (2) {reason ₂₁ , reason_out ₂₁ , conclude ₂₁ } (2) {ounce ₁₁ , troy_ounce ₁₁ , apothecaries’_ounce ₁₁ } (2) {dressed_ore ₁₁ , concentrate ₁₁ } (2)				
Wikipedia concepts (27)	Teck Cominco (9)	Teck (6)	Copper (5)		
	Canada (3)	Short ton (3)	Product (business) (2)		
	British Columbia (2)	Ounce (2)	Consortium (2)		
	Mining (1)	Joint venture (1)	Ore (1)		
	Pacific Ocean (1)	Gold (1)	Silver (1)		
	Metallgesellschaft (1)	United Kingdom (1)	Molybdenum (1)		
	Negotiation (1)	Partnership (1)			

Table 4.1: Example of features in different representations

Method	Concepts identified
Gabrilovich and Markovitch	Teck; John Townson; Cominco Arena; Allegheny Lacrosse Officials Association; Scottish Highlands; Productivity; Tumbler Ridge, British Columbia; Highland High School; Economy of Manchukuo; Silver; Gold (color); Copper (color);
Hu et al.	Tech Cominco; British Columbia; Mining; Molybdenum; Joint Venture; Copper

Table 4.2: Wikipedia concepts identified by other approaches

is also the fourth and sixth most common sense of *dialogue* and *talks* (denoted by dialogue_{14} and talks_{16} respectively).³

All the representation schemes are able to pick up on the different minerals and units—*copper*, *silver*, *ounce*—and the concept-based ones explicitly relate them to synonyms such as *Cu* and *oz*. WordNet fails to note specific named entities such as *Teck Cominco*, but identifies terms such as *complete* that do not resolve to Wikipedia articles. It is quite clear from Table 4.1 that Wikipedia concepts are much more succinct than both words and WordNet concepts from the human point of view. This is a helpful characteristic for generating descriptions of clusters. Although the issue of describing clusters is not investigated in this thesis, it has been shown that users prefer a more informative representation than a simple list of the most frequently occurring keywords in the cluster (Harper et al., 1999). Wikipedia concepts, as shown in Table 4.1, are appropriate for this task.

Table 4.2 summarizes concepts identified from this document by two other Wikipedia-based approaches.⁴ This presents an intuitive comparison of our method (see Table 4.1), implemented in the Katoa toolkit, with related work. The first is the full-text-based method of Gabrielovich and Markovitch (2007), described in Section 4.2, which gathers Wikipedia concepts by measuring term overlap with the given text. This unfortunately allows unrelated concepts such as *Scottish Highlands* and *Economy of Manchukuo* to creep into the representation. Also,

³The meaning of a WordNet synset or a Wikipedia concept can be retrieved by using WordNet’s online search interface at <http://wordnetweb.princeton.edu/perl/webwn/> and by searching for the concept’s name in Wikipedia at <http://en.wikipedia.org>.

⁴Results for these two approaches are taken from Hu et al. (2008), which are part of the identified concepts.

this method performs disambiguation only indirectly, which introduces more irrelevant concepts such as *Copper (color)*.

Both Hu et al. (2008)’s system and Katoa produce the tightest representation of a document, because they only contain the Wikipedia concepts that it discusses. Nevertheless, both are able to expand out from these concepts to consider their related ones, and thus connect texts with similar themes regardless of their textual overlap—or lack of it. For example, when computing the similarity between two documents, Hu et al.’s system considers broader topics mined from the categories to which each article belongs and associated topics mined from the links extending out from each article. Thus *Teck Cominco* can be augmented with *Mining companies in Canada* and *Con Mine* (Hu et al., 2008). Katoa’s representation models can also be expanded (as Section 5.4 will show): a concept such as *Teck Cominco* could be expanded on demand with a huge pool of possibilities, such as different mining companies (*Codelco*, *De Beers*, and about a hundred others), tools (*Drilling rig*, *Excavator*, etc.) and locations (the *Pebble Mine* in Alaska, for example). The flexibility allows them to compete with models already expanded with concept relations.

4.4.2 Comparison of dimensionalities

Table 4.3 compares the dimensionality of each model. Filtering words that occur less than five times in the collection significantly reduces the bag-of-words model’s dimensionality. For WordNet and Wikipedia, the difference between the number of matching terms and concepts indicates the extents to which multiple expressions are used to refer to the same concept. The largest gap occurs on the NewsSim3 dataset, where about 50% of the anchor phrases turn out to be synonymous with another phrase. The concept-based models have more features than the filtered bag-of-words model, but we will show their value in subsequent sections.

4.4.3 Evaluating against category structure

This evaluation assesses clustering against the categories in each dataset by setting the number of clusters equal to that of the categories, and uses the k -means clustering algorithm. Overall, the results in Table 4.4 show that both concept-based

Dataset	Words		Wikipedia		WordNet	
	All	Filtered	Anchor phrases	Concepts	WordNet terms	Concepts
SmallReuters	7651	2794	7233	5478	7395	6619
Med100	13,018	5271	15,107	9217	10,388	9515
NewsSim3	38,561	7902	18,332	9042	10,331	9419
NewsDiff3	20,091	7363	23,623	13,294	15,023	13,442

Table 4.3: Dimensionality of different representations

representations successfully and consistently improve upon the baseline—the bag-of-words model—across different evaluation measures (with one exception which will be justified later), and in many cases the improvements are statistically significant (with paired t -test and $p = 0.05$, see Section 3.3). Results for using all words in the baseline are consistently worse than those using filtered words, and thus not shown. This indicates that concepts are more informative for conveying the topics of a document, regardless of how distinctly the topics are separated and whether they are domain specific or not. Performance with hierarchical agglomerative clustering is presented in the next section.

For WordNet, both disambiguation methods discussed in Section 4.1.2 were evaluated and the most-common-sense rule turned out to be sufficient, which is consistent with Hotho et al. (2003)’s findings. In fact, disambiguation based on context often suffers from unnecessary distinctions between WordNet senses. Appendix D gives detailed results. Therefore the results reported hereafter were obtained with the most-common-sense rule.

On the NewsSim3 and NewsDiff3 datasets, both WordNet and Wikipedia concepts achieve considerable improvements. Interestingly, yet not surprisingly, the best model differs in each case: Wikipedia concepts on NewsSim3 and WordNet concepts on NewsDiff3. This can be explained by their distinct characteristics. Wikipedia concepts are thematically dense descriptors—they provide topic-related information—while WordNet concepts are lexical features, which are more fine grained and some do not correspond to any concrete topics. For example, adverb and adjective concepts like *significantly* and *beautiful* rarely provide topic-related information. When the distinction between the topics of different texts is vague, as for the NewsSim3 dataset, adding non-topic concepts can further blur the boundaries, which is not helpful for clustering.

Dataset	Scheme	Purity		InvPurity		NMI		FMeasure	
		Score	Impr.	Score	Impr.	Score	Impr.	Score	Impr.
SmallReuters	Words	0.668	–	0.646	–	0.687	–	0.483	–
	WordNet	0.675	1%	0.665*	2.9%	0.697*	1.5%	0.501	3.7%
	Wikipedia	0.696	4.2%	0.678*	5%	0.704*	2.5%	0.545*	12.8%
Med100	Words	0.247	–	0.264	–	0.209	–	0.110	–
	WordNet	0.258*	4.5%	0.277	4.9%	0.211	1%	0.124*	12.7%
	Wikipedia	0.312*	26.3%	0.335*	26.9%	0.284*	35.9%	0.170*	54.5%
NewsSim3	Words	0.357	–	0.871	–	0.008	–	0.469	–
	WordNet	0.389	9%	0.931	6.9%	0.056	600%	0.498	6.2%
	Wikipedia	0.614*	72%	0.711	–18.4%	0.248*	3000%	0.520	10.9%
NewsDiff3	Words	0.566	–	0.569	–	0.149	–	0.433	–
	WordNet	0.904*	59.7%	0.933*	64%	0.767*	415%	0.864*	99.5%
	Wikipedia	0.807*	42.6%	0.877*	54.1%	0.579*	289%	0.746*	72.3%

*: statistically significant improvements

Table 4.4: Performance of different representations with the k -means clustering algorithm

To investigate this effect, Figure 4.3 compares the discriminative power of WordNet and Wikipedia concepts in the NewsSim3 and NewsDiff3 datasets, in terms of their log-transformed information gain values. Information gain is a widely used measure of feature salience in machine learning (Witten et al., 2011). It calculates the number of bits of information obtained for category prediction by knowing the value of a term in a document. Formally, it is defined as:

$$\text{InfoGain}(\Omega; a) = H(\Omega) - H(\Omega|a),$$

where H denotes the entropy of a variable, Ω denotes the set of categories—the gold standard—and a is a particular attribute such as a word or concept. Attribute values are numeric, thus they are discretized first using the method based on the minimum description length principle (Fayyad and Irani, 1993; Kononenko, 1995).

Figure 4.3(a) shows similar distributions for most WordNet and Wikipedia concepts on NewsSim3, yet more Wikipedia concepts have greater information gain values—more discriminative power. In contrast, when the distinction between topics (i.e., categories) is clear, as in the NewsDiff3 dataset, which contains

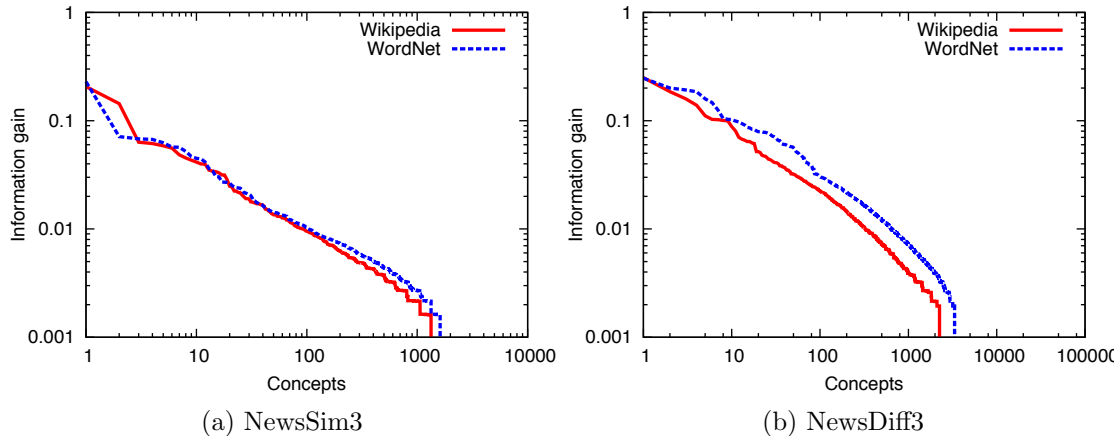


Figure 4.3: Discriminative power of concepts in the NewsSim3 and NewsDiff3 datasets

three very different topics *baseball*, *religion* and *scientific space activities*, WordNet’s non-topic features seem better able to further distinguish these topics and contribute to better text clusters, as Figure 4.3(b) shows.

It is worth noting that although here we use the categories of texts to evaluate feature quality, this information is always covered—not used by the feature generation and clustering processes.

Only one result in Table 4.4 is worse than the baseline: the inverse purity of clusters obtained with the Wikipedia concept-based model on the NewsSim3 dataset, which degrades from the baseline performance 0.871 to 0.711. However, the decrease does not necessarily imply worse clustering. As explained in Section 2.4, inverse purity by itself only measures the extent to which texts from the same category are assigned to the same cluster, and does not consider whether texts from different categories are grouped together. In particular, the maximum inverse purity of 1 can be easily achieved by putting all texts into one cluster. A fair judgement must combine both inverse purity and purity. Increases in the other three measures on this dataset show that Wikipedia concepts do indeed separate its documents more into different groups rather than place them in a single group, but in such a way that makes the overall cluster structure fit the existing text categorization better.

The smallest improvements occur on the SmallReuters dataset. Hu et al.

Concept systems	Methods	Purity	InvPurity
Wikipedia	Hu et al. (2008)	0.655	0.598
	Gabrilovich and Markovitch (2007)	0.605	0.548
	Katoa	0.696	0.678
WordNet	Hotho et al. (2003)	0.607	0.548
	Katoa	0.675	0.665

Table 4.5: Performance of other approaches on the SmallReuters dataset with the k -means clustering algorithm

(2008) also report performance of three related methods in terms of cluster purities on this dataset: for their method and Gabrilovich and Markovitch (2007)’s method, which both utilize Wikipedia, and Hotho et al. (2003)’s WordNet-based approach. These methods represent the state-of-the-art in exploiting Wikipedia and WordNet for generating document representation models. Table 4.5 compares our method Katoa with them, and shows that it competes favourably with all of them, for both concept systems and in both aspects of cluster quality: cluster purity and inverse purity. For cluster purity, 6.3% and 15% increases are obtained over Hu’s and Gabrilovich’s methods respectively, and an 11.2% improvement on Hotho’s method when using WordNet. Greater improvements are observed for the inverse purity measure: 13.4%, 23.7% and 21.4% respectively. Hu et al. further optimize their method by using half of the dataset as training data to tune the various parameters involved in their system, achieving a purity of 0.697 and an inverse purity of 0.636 on the remaining half. Katoa obtains accuracy at a similar level but without any training and tuning.

Hu et al.’s method is Katoa’s closest competitor. One important constituent of their method is their extension of the overlap-based document similarity measure—the cosine measure—to include semantic connections among concepts, so that documents with different yet related concepts can also be related (as discussed in Section 4.4.1). In contrast, we have not yet enriched the similarity measure with such information—it will be discussed in the next chapter—and only taken surface similarity into account. With surface similarity only, Hu et al. achieve 0.603 and 0.544 in purity and inverse purity, which gives Katoa a 15.4% and 24.6% advantage over this baseline. Using anchor text as index vocabulary and the su-

pervised machine-learning-based method for disambiguation both contribute to Katoa’s success. In contrast, Hu et al. use string matching to relate document terms to Wikipedia articles and an unsupervised disambiguation method that was originally designed for named entities.

Hotho et al.’s method, which consults WordNet, also utilizes concept relations. If a concept and its hypernym both appear in the same document, the latter is stressed by augmenting its weight with the descendant concept. For example, if a document mentions both *laptop* and *computer*, *computer* will be emphasized by adding *laptop*’s weight to it. Furthermore, they only consider nouns, and ignores verb, adverb and adjective concepts. In contrast, Katoa uses all concepts and outperforms their representation model, despite the fact that it does not consider concept generalizations (we rectify this in Chapter 5).

4.4.4 Evaluating with different numbers of clusters

The previous evaluation was conducted under a specific condition: using k -means with the number of clusters equal to the existing number of categories. This section investigates whether Katoa’s behaviour stays the same under different conditions: with different clustering algorithms and when looking for different cluster structures.

We tested the two clustering algorithms— k -means and hierarchical agglomerative clustering—and varied the number of clusters from 5 to 60 in increments of 5. Figures 4.4 to 4.8 plot the overall cluster quality against the number of clusters; for NewsDiff3 and NewsSim3 datasets, plotting starts with 3 clusters—the number of categories in these datasets. Cluster quality is quantified with the normalized mutual information of the resulting clusters (NMI, see Section 2.4.2), because unlike purity and inverse purity, NMI by itself indicates the goodness of fit between the clusters and the specified category structure. FMeasure is also not used because it measures the extent to which pairwise document relations are obeyed in the clusters, which does not as directly reflect the structural similarity between clusters and categories as NMI.

Figure 4.4 shows that for the k -means algorithm the concept-based models are consistently more effective than the baseline bag-of-words model across all

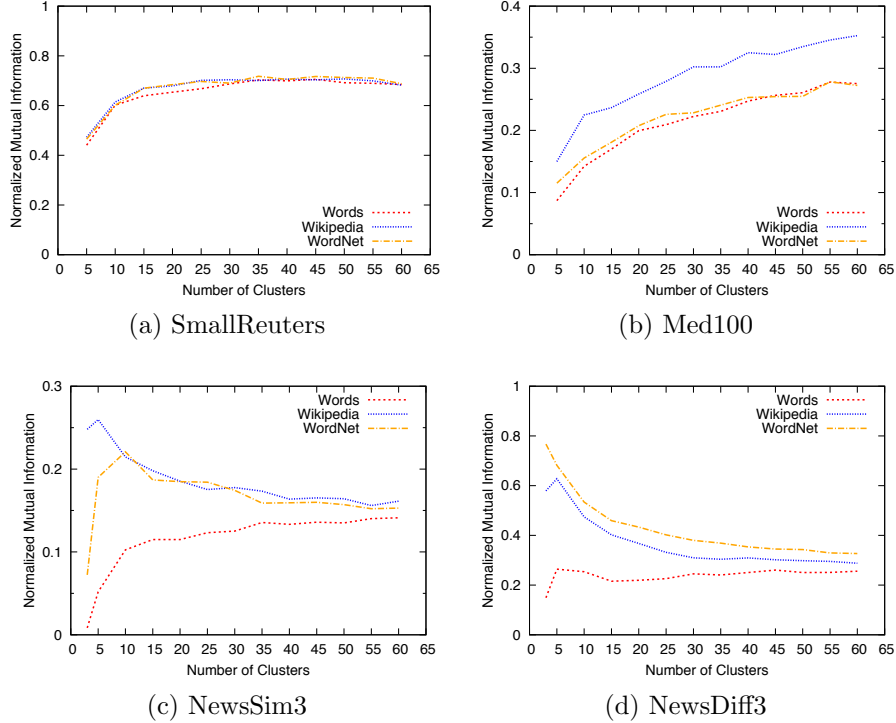


Figure 4.4: Performance of the k -means clustering algorithm across datasets

datasets. A trend exists across these figures that Wikipedia concepts are in general more discriminative than WordNet concepts, except on the NewsDiff3 dataset (see Figure 4.4(d)) where the additional lexical features from WordNet turn out to be advantageous. This suggests that Wikipedia concepts are better choices in the absence of prior knowledge of a given text collection.

Figure 4.4(b) illustrates the impact of topic domains: Wikipedia, considering its coverage of world knowledge, is clearly favoured for domain-specific texts. WordNet, as a lexical database, covers very limited topic domains: many medical terms have corresponding articles in Wikipedia, but are absent from WordNet. This explains the fact that WordNet concepts only achieve slightly better performance than words, while Wikipedia concepts are much more effective on this dataset.

In contrast to k -means, performance with the hierarchical clustering methods varies substantially with different linkage functions, as Figures 4.5 to 4.8 show. For group-average-link and average-link (Figures 4.5 and 4.6), concepts

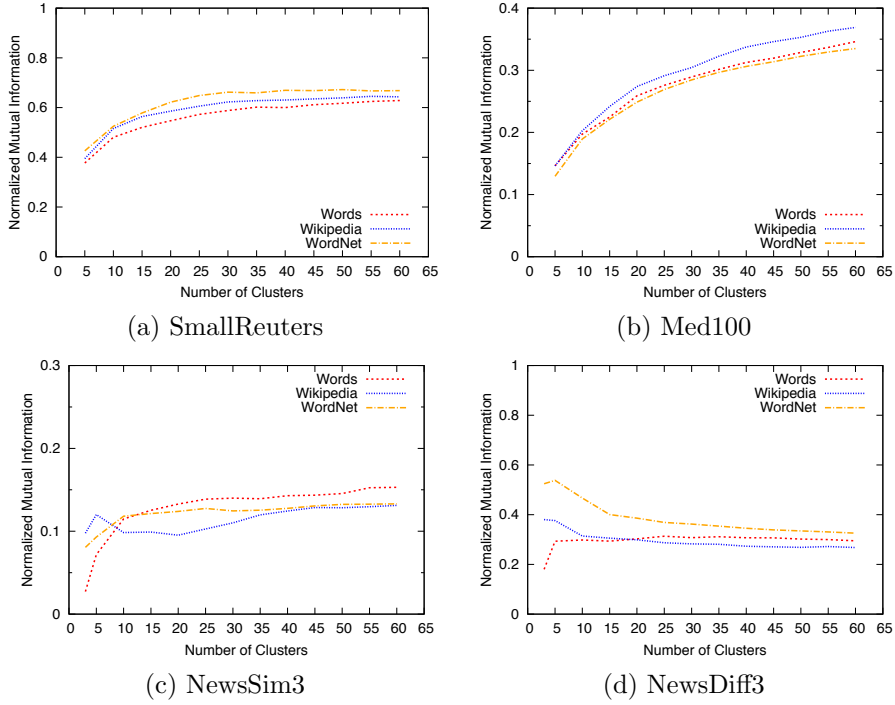


Figure 4.5: Performance of hierarchical agglomerative clustering with group-average-link across datasets

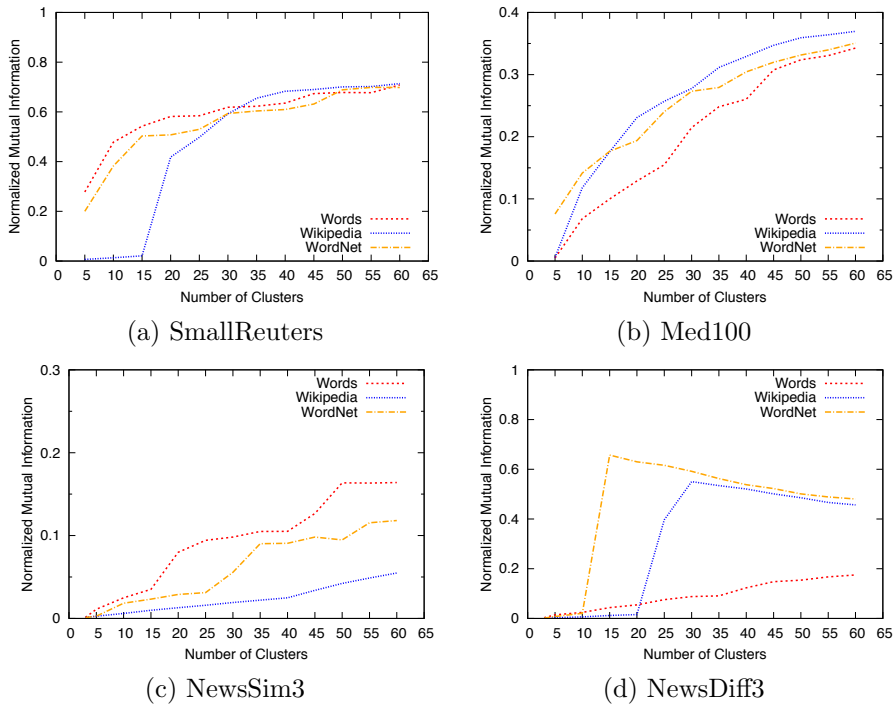


Figure 4.6: Performance of hierarchical agglomerative clustering with average-link across datasets

4.4. EXPERIMENTAL RESULTS

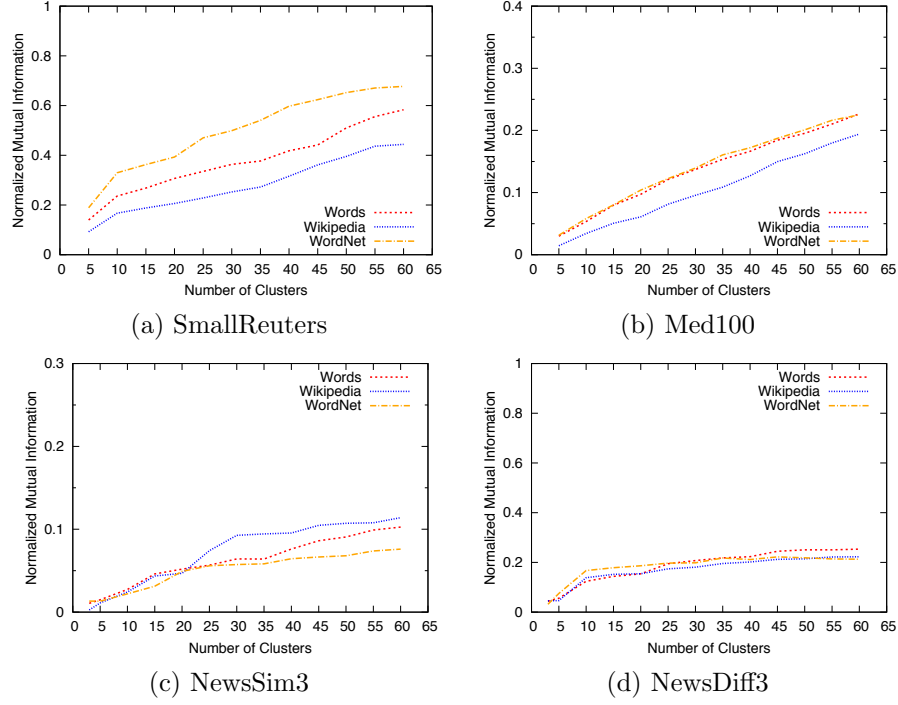


Figure 4.7: Performance of hierarchical agglomerative clustering with complete-link across datasets

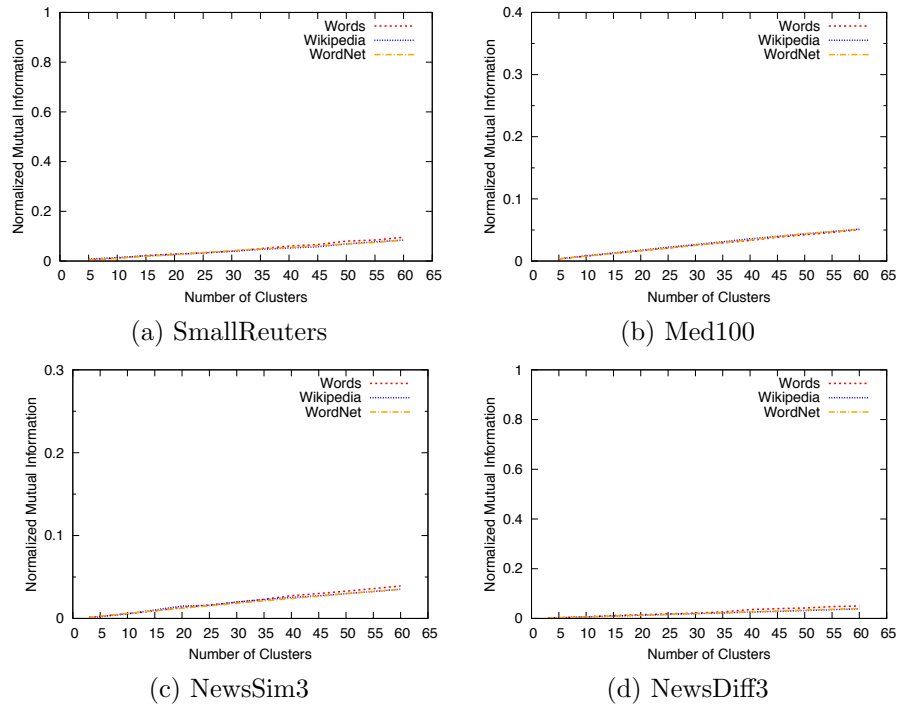


Figure 4.8: Performance of hierarchical agglomerative clustering with single-link across datasets

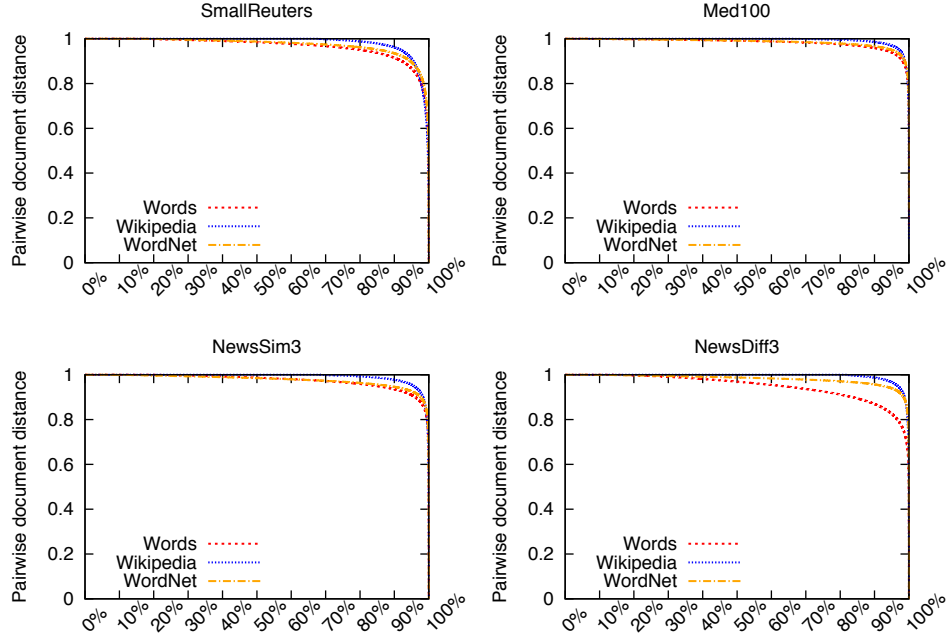


Figure 4.9: Distribution of pairwise document distances in each dataset

usually outperform words with an appropriate number of clusters (e.g. see Figures 4.5(c) and 4.5(d)). Especially with the latter function, results on SmallReuters and NewsDiff3 (see Figures 4.6(a) and 4.6(d)) show that bigger clusters tend to be merged into one when the specified number of clusters is small, resulting in an extremely unbalanced cluster structure and a low NMI value. This highlights the importance of this parameter. Estimating the number of clusters in a dataset is a problem in its own right, and has been extensively researched, such as *X*-means (Pelleg and Moore, 2000) and Tibshirani et al. (2001)’s gap-statistic-based method. These off-the-shelf methods can be applied immediately with the clustering algorithms, yet evaluating their accuracy is of less interest to this thesis, therefore we do not discuss them in detail.

Figures 4.7 and 4.8 show the results of complete-link and single-link. With complete-link, the Wikipedia-based model falls behind the bag-of-words baseline on three datasets: see Figures 4.7(a), (b) and (d). This is because complete-link takes the minimum similarity between components of two clusters as the similarity between the clusters. In other words, it assumes that components in each cluster are very close to each other. However, Wikipedia concepts, which are

less ubiquitous than words and WordNet concepts, tend to decrease the similarity between texts, leading to assimilated similarity between clusters (e.g. close to zero for the majority of clusters in the later stages of the clustering process) and increased variations within clusters, which confuses the complete-link method and results in less satisfactory performance.

To investigate this, Figure 4.9 compares the distribution of pairwise document distances calculated with the three models. The top and bottom lines always correspond to the Wikipedia-based and bag-of-words models respectively. With a closer look at the hierarchical clustering process, we find that decrease in intra-cluster similarity results in highly unbalanced clusters: most documents are aggregated into a few dominant clusters while a few are scattered in several outlier clusters. In contrast, as document distances decrease (i.e., similarities increase), as in the bag-of-words model, the assumption of complete-link becomes more justified and empirically more documents are assigned to the smaller clusters, contributing to a more balanced cluster structure. Results on the NewsSim3 dataset shows similar effects: Wikipedia concepts only outperform the baseline when there are sufficient clusters to ensure that documents in the same cluster are similar to each other.

The single-link approach is the least effective clustering method and the differences between representation models are trivial. Documents tend to converge in one cluster because of its chaining effect (Jain et al., 1999): the tendency to produce elongated and unbalanced clusters.

These results also show that the WordNet concepts and words have similar behaviour compared to Wikipedia concepts, as Figures 4.4(b), 4.5(c) and 4.6(a) show. This is not surprising because the former retain a greater proportion of a document’s surface information, whereas the latter are more succinct.

Overall, results of this evaluation demonstrate the effectiveness of Katoa’s representation models under varying conditions. Although there are occasions where they fall behind the bag-of-words baseline, we show in the next section that with more effective clustering algorithms they always exceed it.

Dataset	SmallReuters			Med100			NewsSim3			NewsDiff3			Sum
	Wins	Losses	Diff	Wins	Losses	Diff	Wins	Losses	Diff	Wins	Losses	Diff	
<i>k</i> -means	12	0	12	6	4	2	11	1	10	9	1	8	32
Group average	7	3	4	12	0	12	10	2	8	8	2	6	30
Average	6	4	2	7	3	4	3	7	-4	5	3	2	4
Complete	3	9	-6	3	9	-6	4	7	-3	4	8	-4	-19
Single	0	12	-12	0	12	-12	0	11	-11	0	12	-12	-47

Table 4.6: Relative performance of different clustering algorithms across datasets

4.4.5 Evaluating the clustering algorithms

This section focuses on the comparative performance of different clustering algorithms: knowing which ones are consistently more effective helps one make better choices in real world applications. We compare the five clustering algorithms with each other—*k*-means and hierarchical agglomerative clustering with four linkage functions—and count how often one algorithm obtains a statistically significant improvement over another.

This evaluation uses the three representation models—the bag-of-words model and the two concept-based models—and varies the number of clusters from 5 to 60, resulting in 12 results on each dataset. For *k*-means clustering, five runs are performed for each specified number of clusters, each with a different set of randomly sampled documents as initial seed clusters; and the average performance of these runs is taken. For hierarchical algorithms, which are deterministic, one run is conducted for each specified number of clusters.

Each algorithm is compared against the other four, with the three representation models, resulting in $3 \times 4 = 12$ comparisons in total. Table 4.6 summarizes the number of wins (i.e., significantly better result) and losses (i.e., significantly worse result). The difference indicates an algorithm’s effectiveness, and is summed in the last column of Table 4.6.

In general, *k*-means and hierarchical agglomerative clustering with group-average-link are the best contenders. Single-link and complete-link both suffer from equating cluster similarity with the similarity between a single pair of documents. Average-link and group-average-link avoid this by taking every document pair into account (see Section 2.3.1).

4.5 Combining different representations

We have seen the distinct characteristics of the concept systems and their impact on clustering: WordNet concepts capture more lexical features, while Wikipedia concepts provide more topic-related information. This section explores whether combining different types of features can further benefit the task.

The most straightforward approach is to concatenate features from each individual representation (including words), creating the most comprehensive feature set. Taking the document in Table 4.1 as an example, such combination will include named entities like *Teck Comino* that do not exist in WordNet, as well as terms like *complete* that do not have a corresponding concept in Wikipedia.

However, this representation counts many concepts at least twice, as a word and as a concept, resulting in undesirable redundancy. An alternative is to take the redundancy between features into account by discarding those that are captured in another model. Thus, terms that have been mapped to WordNet or Wikipedia concepts are discarded, while other terms are kept.

These two strategies are applied to the Wikipedia-based model, because WordNet concepts are pervasive: it is unlikely that meaningful concepts will be found in text fragments that do not match any WordNet term. Four hybrid representations are generated: two by *adding* WordNet concepts and words respectively (denoted by AddWordNet and AddWord), and two by *supplementing* with WordNet concepts or words in the text that cannot be mapped to any Wikipedia concept (denoted by ReplaceWordNet and ReplaceWord).

The goal of combining different features is to further improve the concept-only models. Table 4.7 shows that only on two datasets this goal is achieved, by the AddWordNet and AddWord models on SmallReuters and the ReplaceWord model on NewsDiff3, which are noted in bold. The experimental setup is identical to that in Section 4.4.3—using k -means and the exact number of clusters—and the better baseline results are noted in bold italics.

These results reveal an interesting trade-off between the increase in the additional information and the redundancy brought by adding features. Models created using the *add* strategy are most comprehensive, yet they generally outperform only the inferior baseline, not the other one. AddWordNet outperforms

Dataset	Representation	Purity	InvPurity	NMI	FMeasure
SmallReuters	WordNet	0.675	0.665	0.697	0.501
	Wikipedia	0.696	0.678	0.704	0.545
	ReplaceWord	0.662	0.641	0.680	0.483
	ReplaceWordNet	0.663	0.646	0.682	0.492
	AddWord	0.711	0.714*	0.735*	0.572
	AddWordNet	0.725*	0.723*	0.737*	0.578*
Med100	WordNet	0.258	0.277	0.211	0.124
	Wikipedia	0.312	0.335	0.284	0.170
	ReplaceWord	0.302	0.301	0.258	0.151
	ReplaceWordNet	0.293	0.324	0.247	0.149
	AddWord	0.287	0.301	0.246	0.145
	AddWordNet	0.312	0.311	0.273	0.159
NewsSim3	WordNet	0.389	0.931	0.056	0.498
	Wikipedia	0.614	0.711	0.248	0.520
	ReplaceWord	0.387	0.958	0.063	0.512
	ReplaceWordNet	0.423	0.901	0.104	0.498
	AddWord	0.392	0.947	0.062	0.506
	AddWordNet	0.481	0.853	0.178	0.521
NewsDiff3	WordNet	0.904	0.933	0.767	0.864
	Wikipedia	0.807	0.877	0.579	0.746
	ReplaceWord	0.922*	0.951*	0.798*	0.884
	ReplaceWordNet	0.758	0.901	0.602	0.733
	AddWord	0.562	0.603	0.155	0.437
	AddWordNet	0.836	0.931	0.714	0.822

*: statistically significant improvements

Table 4.7: Performance of different hybrid representations with the k -means clustering algorithm

WordNet and loses to Wikipedia on Med100 and NewsSim3, while the opposite is observed on NewsDiff3. Only on SmallReuters they outperform both baselines, which is probably due to the characteristics of this dataset, based on similar results reported by Hotho et al. (2003). Regardless of redundancy, WordNet concepts are usually better supplements than words.

The *replace* strategy is less effective, and it only improves upon the best baseline for NewsDiff3. A possible reason is that text that cannot be mapped to Wikipedia concepts is not helpful for discriminating the topics of the documents anyway, which make the WordNet concepts stemmed from it useless in the hybrid models. The only success of this strategy is on NewsDiff3: when topics are

well separated, adding non-topic features helps to further clarify the distinctions. The *replace* strategy's unsatisfactory performance again suggests that Wikipedia concepts are better thematic descriptors for texts than the other feature sets.

4.6 Summary

The goal of this chapter is to investigate the fundamental problem of concept-based text clustering: how to represent topics in texts with concepts. We introduced Katoa's methods for mapping texts to concepts that explore WordNet and Wikipedia, and systematically evaluated them in text clustering.

The empirical results show that Katoa's concept-based representation models are consistently more effective than the traditional bag-of-words model, when combined with an appropriate clustering method. This provides strong support for the hypothesis set out at the beginning of this chapter: representing texts by concepts provides a more effective basis for text clustering than the traditional bag-of-words model.

Several other issues are investigated: the characteristics of each concept system, the consistency of a clustering algorithm's behaviour with different concept-based representation models, and the effectiveness of strategies for combining different types of features. By investigating all these aspects, this chapter establishes the footing for subsequent research.

This chapter makes several contributions. First, it identifies state-of-the-art methods for mining concepts and their relations from WordNet and Wikipedia, and utilizes them to create a concept-based text representation that competes favourably with the best methods in the literature. Second, although it has been shown previously for WordNet and Wikipedia individually that they can improve clustering, this is the first systematic comparison of the two concept systems on a variety of text collections.

5

Semantically enriched clustering

In the previous chapter we demonstrated the advantages of using concepts as text descriptors, but one glaring shortcoming is that the clustering process treats concepts as *orthogonal* features: it only uses binary information about whether one feature is the same as another or not, regardless whether the features are close. The rich semantic connections among concepts are unfortunately ignored during the clustering process, thus we refer to it as the *plain* concept-based clustering method in this chapter. Noticing this omission, this chapter investigates the following hypothesis:

utilizing semantic concept relatedness can help to further enhance concept-based text clustering.

Integrating semantic relations into text clustering is an attractive yet challenging idea. It is challenging because the traditional plain clustering method has been shown to be quite effective, and there is even evidence that considering concept relatedness can adversely impact clustering (Passos and Wainer, 2009). It is not clear whether expanding Katoa to consider concept relatedness by consulting WordNet and Wikipedia can actually contribute to more effective text clustering, and if so how. Furthermore, it is also interesting to see whether the same methods for utilizing concept relatedness can be effective with both concept systems.

There are various ways to enrich clustering with concept relatedness. This chapter explores three such methods. The first section below explains the motivation of each one by considering an application scenario. In the previous chapter

we explained the concept relatedness measures for WordNet and Wikipedia without discussing them in the context of other measures; Section 5.2 provides this discussion and surveys their alternatives. These three methods take two distinct perspectives for utilizing concept relatedness: the first two (explained in Section 5.3) use it to identify and emphasize thematically more representative concepts; the other (explained in Section 5.4) targets the orthogonality between features and extends the computation of document similarity to include concept relatedness. These methods are then evaluated and discussed in Section 5.6. Section 5.7 revisits the hypothesis and discusses the strengths and weaknesses of our approaches.

5.1 Semantic concept relatedness and clustering

To illustrate why semantic concept relatedness can be useful for relating documents, Figure 5.1 shows three short documents that can be regarded as sharing the same theme—*smoking and health*.¹ Terms that successfully resolve to a Wikipedia article are noted in bold and listed beside the documents with the concepts (i.e., Wikipedia articles) they map to. Figure 5.2 shows the corresponding concept-document matrix.

All documents mention a variety of concepts concerning *health*: some, such as *infection* and *health* are general, while others, such as *weight loss* (from the term *losing weight*) and *ischemia*: inadequate blood supply, are specific. These concepts are semantically related to each other, which invalidates the *bag-of-features* model’s underlying assumption of independence.

The repeated occurrences of concepts about the same topic—*health* in this case—instantly reveals the document’s theme, or at least one aspect of it. This suggests that concepts that are closely related to the theme, such as *angina*, might be more representative and useful compared to less related ones like *United Kingdom*. Such effects are not reflected in either the representation models or the clustering processes described in the previous chapter. In fact, as Figure 5.2

¹Although the last document does not literally mention *smoking*, it is still appropriate to be considered as relevant because it talks about a health situation that is commonly caused by *smoking*.

5.1. SEMANTIC CONCEPT RELATEDNESS AND CLUSTERING

D1 By giving up smoking , losing weight , and becoming more active people can reduce their risk of cardiovascular disease two to three-fold, which largely outweighs the risks of taking the medications .	smoking → Tobacco smoking losing weight → Weight loss cardiovascular disease → Cardiovascular disease medications → Pharmaceutical drug
D2 Smoking and passive smoking can cause many health problems such as respiratory infections , asthma and lung cancer .	smoking → Tobacco smoking passive smoking → Passive smoking health → Health respiratory → Respiratory system infections → Infection asthma → Asthma lung cancer → Lung cancer
D3 In the UK , there are 2 million people affected by angina : the most common symptom of coronary heart disease . Angina pectoris , commonly known as angina , is severe chest pain due to ischemia (a lack of blood , hence a lack of oxygen supply) of the heart muscle .	the UK → United Kingdom angina → Angina pectoris coronary heart disease → Coronary heart disease angina pectoris → Angina pectoris chest pain → Chest pain ischemia → Ischemia blood → Blood oxygen → Oxygen heart muscle → Cardiac muscle

Figure 5.1: Example documents on *smoking and health*

shows, most concepts will be treated equally because they all occur once in these documents (if the inverse document frequency weighting is not considered). This motivates us to consider measuring and utilizing a concept’s thematic representativeness, as Section 5.3 will show.

The three documents in Figure 5.1 share similar topics. However, the plain clustering method will not notice this because only one concept occurs in more than one document: *tobacco smoking*, which is noted in bold in Figure 5.2. A more effective approach would take into account the fact that concepts like *angina pectoris* and *coronary heart disease* are related to *tobacco smoking*. Then, documents on similar topics but expressed with different yet related concepts could be connected. This motivates the other enriched clustering method of Katoa that counts concept relatedness in computing document similarities—extending document similarity beyond surface overlap (see Section 5.4).

	D1	D2	D3
1: Angina pectoris	0	0	3
2: Asthma	0	1	0
3: Blood	0	0	1
4: Cardiac muscle	0	0	1
5: Cardiovascular disease	1	0	0
6: Chest pain	0	0	1
7: Coronary heart disease	0	0	1
8: Health	0	1	0
9: Infection	0	1	0
10: Ischemia	0	0	1
11: Lung cancer	0	1	0
12: Oxygen	0	0	1
13: Passive smoking	0	1	0
14: Pharmaceutical drug	1	0	0
15: Respiratory system	0	1	0
16: Tobacco smoking	1	1	0
17: United Kingdom	0	0	1
18: Weight loss	1	0	0

Figure 5.2: Concept-document matrix of the example documents in Figure 5.1

5.2 Measures for semantic concept relatedness

To incorporate concept relatedness into clustering, one first needs a measure that quantifies the strength of the connection between two concepts. We use a numeric score between 0 and 1 to indicate how closely the concepts relate to each other, 1 meaning synonymous and 0 meaning completely unrelated. Developing such a measure is a research problem in its own right, and has been extensively investigated for both WordNet and Wikipedia. We have already mentioned two: the path length based measure of Leacock and Chodorow (1997) (LCH) for WordNet and the hyperlink structure based measure of Milne and Witten (2008a) (WLM) for Wikipedia. Here we discuss their alternatives, and explain why we chose these ones.

There are three types of relatedness measures for WordNet. First are path-based measures (Rada et al., 1989; Wu and Palmer, 1994; Leacock and Chodorow, 1997; Hirst and St-Onge, 1997), which use the shortest path in the taxonomies between two concepts. The second type, which includes Resnik (1995)’s information content measure and its variants (Lin, 1998; Jiang and Conrath, 1997), combines

distributional statistics from a corpus with the structural taxonomy. The third type measures relatedness based on text overlap between concepts (Lesk, 1986; Banerjee and Pedersen, 2003), where text normally refers to the brief gloss associated with each concept in WordNet.

For Wikipedia, there are three widely acknowledged measures: WikiRelate (Strube and Ponzetto, 2006), the Wikipedia Link based Measure (WLM) (Milne and Witten, 2008a) used in Katoa, and the Explicit Semantic Analysis (ESA) measure (Gabrilovich and Markovitch, 2007). WikiRelate is like Leacock and Chodorow’s LCH measure but replaces WordNet’s taxonomies by Wikipedia’s category structure. WLM, described in Section 4.2.3, computes relatedness between Wikipedia articles based on their associated hyperlink structures. ESA first represents each concept by a weighted vector of Wikipedia articles identified based on full text overlap, and computes relatedness using the cosine rule on the vectors.

Accuracy and efficiency are the two factors to consider when choosing a concept relatedness measure. Accuracy can be assessed by comparing against human judgement: the more consistent its predictions are with human rated semantic relatedness between concepts, the more accurate a measure is. Efficiency is important because the number of pairwise concept relations is the square of the number of concepts, which ranges from about 5,500 to 13,300 for the four experimental datasets of Section 3.1 (see Section 4.4.2). The two relatedness measures used in this thesis—LCH and WLM—have both been shown to be quite accurate and efficient compared to their competitors (Strube and Ponzetto, 2006; Milne and Witten, 2008a). Thus, they are used in Katoa.

5.3 Concept reweighting based on centrality

How to identify concepts that are more representative of a text’s theme and utilize them to enhance clustering? We introduce *context centrality* as a way of assessing how representative a concept is of a given context. Concepts with higher centrality are thematically more relevant and thus more representative. Sections 5.3.2 and 5.3.3 investigate two different ways of utilizing context centrality, which are then compared and discussed in Section 5.3.4.

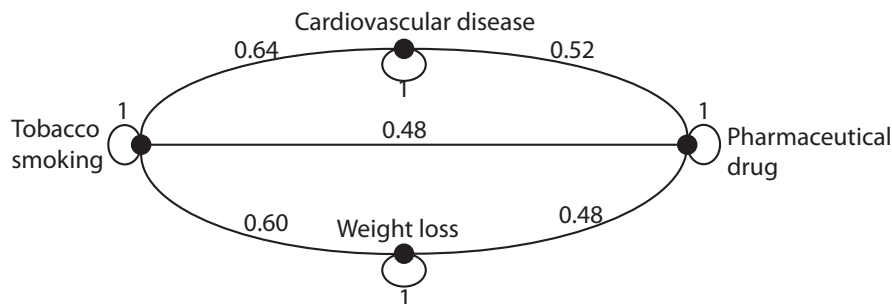


Figure 5.3: Concept graph of the first example document in Figure 5.1

5.3.1 Context centrality

Not every concept is equally informative for conveying the topics of a text: usually some are closer to the theme than others. Consider again document D3 in Figure 5.1, which concerns a certain kind of heart problem in the UK. The name in medical terminology (*angina*) and the problem of *coronary heart disease* that it is associated with are closer to the document's theme, because most concepts mentioned in this document are about similar topics. Thus such concepts should probably be emphasized because they are more representative.

Concepts and their connections can be represented by a weighted undirected graph whose vertices are concepts and whose edges connect pairs of concepts, weighted by their relatedness. Figure 5.3 shows an example: the concept graph created from document D1 in Figure 5.1. When two concepts have zero relatedness, we nevertheless create an edge with zero weight. We also create an edge from each vertex to itself, with a weight of 1.

In graph theory, the *centrality* of a vertex quantifies how central it is, given the graph it occurs in (Freeman, 1978). For weighted undirected graphs, such as the concept graphs here, one way to measure centrality is to calculate a vertex's average edge weight to every vertex in the graph, including itself (Freeman, 1978). A vertex with high centrality is considered more likely to be the *centre* of the graph.

A concept graph represents the thematic *context*, comprising every concept in the text. Vertex centrality now translates into the extent to which a concept is the thematic centre of the context, and higher values indicate that it is closer to the central theme. Thus vertex centrality can be used as a measure of *context*

centrality, to measure the representativeness of a concept with respect to a given context.

Formally, for a set C of concepts, denote the resulting weighted graph by G_C and a concept c 's weight in C by $w(c, C)$. The context centrality of a concept c with respect to the context C is defined as

$$CC(c, C) = \frac{\sum_{c_j \in C} rel(c, c_j) \times w(c_j, C)}{\sum_{c_j \in C} w(c_j, C)},$$

where $rel(c, c_j)$ is the relatedness between c and c_j (including the relatedness of c to itself if $c \in C$). Context centrality is normalized between 0 and 1 and a higher centrality indicates that the feature is more representative of the context.

The concept weight $w(c, C)$ can be based on either a concept's presence and absence—a binary measure—or its number of occurrences—a weighted measure. For the former, $w(c, C)$ is 1 if a concept occurs in the context and 0 otherwise, and the denominator equals to the total number of concepts in C . The latter weighting scheme will emphasize connections to concepts that are frequently mentioned in a context. For example, *angina pectoris* is mentioned three times in document D3, so any connection with it will be counted three times.

At this point, concept weights could be either binary or its occurrence frequency, and *idf* weighting has not been applied yet at this stage. In the following sections, we will adapt these weights to emphasize thematically representative concepts. Two types of context are considered: the text that mentions a concept (*local centrality* in the next section) and another text (*relative centrality* in Section 5.3.3).

5.3.2 Local centrality

The most straightforward way to utilize context centrality is to reweight each concept by its centrality with its surrounding context, which usually refers to the text that mentions the concept. This is called *local centrality* to emphasize that the surrounding context is used as the reference.

Concepts	Coronary					Cardiac muscle	Occurrences	Binary Weighted			
	United Kingdom	Angina pectoris	heart disease	Chest pain	Ischemia				Blood Oxygen		
United Kingdom	1.0	0.14	0.11	0.08	0.01	0.07	0.04	0	1	0.181	0.173
Angina pectoris	0.14	1.0	0.73	0.7	0.7	0.44	0.34	0.74	3	0.599	0.679
Coronary heart disease	0.11	0.73	1.0	0.63	0.65	0.46	0.4	0.66	1	0.58	0.61
Chest pain	0.08	0.7	0.63	1.0	0.57	0.49	0.47	0.66	1	0.575	0.6
Ischemia	0.01	0.7	0.65	0.57	1.0	0.56	0.46	0.76	1	0.588	0.61
Blood	0.07	0.44	0.46	0.49	0.56	1.0	0.58	0.54	1	0.518	0.502
Oxygen	0.04	0.34	0.4	0.47	0.46	0.58	1.0	0.42	1	0.464	0.439
Cardiac muscle	0	0.74	0.66	0.66	0.76	0.54	0.42	1.0	1	0.598	0.626

Table 5.1: Example of local context centrality

D2 concepts		Passive			Lung			Occurrences	Binary Weighted
D1 concepts		smoking	Smoking	Health	Respiratory	Infection	Asthma		
Tobacco smoking	0.69	1.0	0.45	0.53	0.55	0.55	0.63	1	0.629
Cardiovascular disease	0.54	0.64	0.5	0.52	0.56	0.52	0.52	1	0.543

(a) with D2

D3 concepts		Coronary					Cardiac		Occurrences	Binary Weighted	
D1 concepts		United Kingdom (1)	Angina pectoris (3)	heart disease (1)	Chest pain (1)	Ischemia (1)	Blood Oxygen (1)				
Tobacco smoking	0.18	0.58	0.58	0.63	0.56	0.46	0.39	0.50	1	0.485	0.504
Cardiovascular disease	0.14	0.65	0.71	0.56	0.61	0.55	0.40	0.59	1	0.526	0.551

(b) with D3

Table 5.2: Example of relative context centrality

Recall that a text is represented by a vector of concepts, and each concept has an associated weight. We adapt this weight by multiplying it with the concept's local centrality, thus emphasizing thematically representative ones that have high centrality values, and reducing the impact of less relevant ones, by multiplying their weights with low centrality values. Given a document d_A with concepts C_A , the new weight of a concept c mentioned in d_A becomes

$$w_r(c, d_A) = w(c, C_A) \times CC(c, C_A).$$

The subscript r denotes that this is the adapted weight.

Table 5.1 shows an example of the binary and weighted local context centrality calculated for concepts in document D3 of Figure 5.1. The lowest and highest centrality values in each scheme are noted in bold. Except for *angina pectoris*, this document mentions all concepts only once, yet they receive quite different weights after taking their relatedness with each other into account. Concepts belonging to the medical domain become more prominent, the stray concept *United Kingdom* has the lowest centrality in both schemes and is diminished.

Reweightings with context centrality allows the same concept to be treated differently in different contexts. For example, when computing the similarity between D3 and another document, *United Kingdom*'s contribution will be reduced from 1 to about 0.16. However, if mentioned in a document about holding the 2012 Olympic Games in London, *United Kingdom* would likely be strengthened because it is probably more related to the context.

5.3.3 Relative centrality

The definition of context centrality applies to any concept and text, regardless of whether it occurs in that text or not. Given two texts, the centrality of each concept in one text with respect to the other can be measured. This is called *relative centrality* to indicate that it is the other text that is used as the reference and the reweighting is applied to both texts. For example, considering *tobacco smoking* and *health*, the former is likely to be more relevant if the other text also discusses *smoking*, while the latter is more likely to be relevant if the other focuses on health-related topics.

Reweighting by relative centrality is performed *during* the clustering process, before measuring the similarity between two texts. The intuition is that concepts that are more coherent with the other text are more relevant for relating the two texts and should therefore be stressed.

Computing relative centrality is the same as computing local centrality, the only difference being that the context in question is not its surrounding context but a different one in which the concept might or might not occur. Formally, given two documents d_A and d_B , let C_A and C_B denote the set of concepts associated with each document. Before computing the similarity between d_A and d_B , each concept c from C_A is weighted as:

$$w_r(c, C_A; C_B) = w(c, C_A) \times CC(c, C_B).$$

Concepts in C_B are weighted in the same way with respect to C_A , by exchanging C_A and C_B in the above formula. The left part of the above formula defines that the adapted weight w_r depends on both documents: a document will be weighted differently when compared to different documents.

As an example of relative centrality, Table 5.2 computes the centrality of *tobacco smoking* and *cardiovascular disease* from document D1 with respect to documents D2 and D3, and shows that *tobacco smoking* is more coherent with D2 while *cardiovascular disease* is more coherent with D3. This makes sense because D2 mentions both *smoking* and *health*, while D3 only mentions the *health* aspect. This example clearly shows the diversity in inter-document connections, and relative centrality’s capability in identifying features that are relevant to them. Thus emphasizing concepts with high relative centrality should help to strengthen such connections and help clustering.

5.3.4 Discussion

Local and relative centrality extends the plain clustering method to take into account the diversity in the contexts, and the resulting variations in a concept’s representativeness. Figure 5.4 shows the distribution of each concept’s (binary) local and relative centrality for all the Wikipedia concepts in the SmallReuters dataset, in terms of the standard deviation of each concept’s centrality values

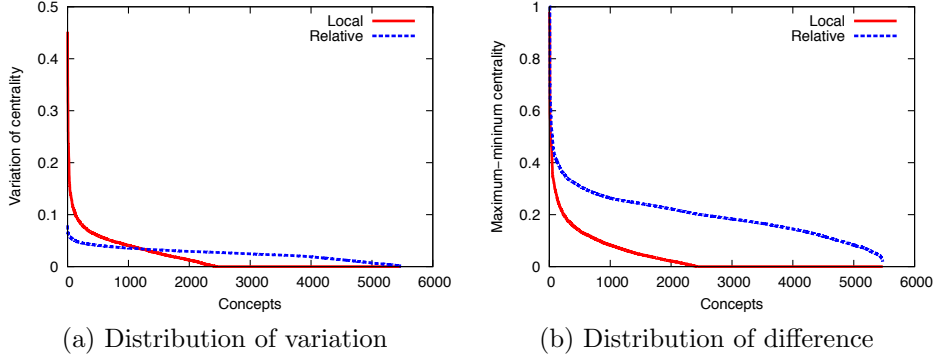


Figure 5.4: Distribution of concept’s local and relative context centrality on the SmallReuters dataset

(Figure 5.4(a)), and the size of its value range—gap between its maximum and minimum values (Figure 5.4(b)).

For a given concept, the number of values (i.e., sample size) for local centrality equals its document frequency—the number of documents that mention it; and for relative centrality it equals the collection’s size—the total number of documents. Each plot is ranked individually, i.e., the same x value usually corresponds to a different concept for each plot. Similar distributions exist for WordNet concepts and the other three datasets.

In most cases, a concept’s local centrality varies with each single document that mentions it. For example, *export* occurs in 193 documents and has 192 distinct local centrality values, ranging from 0.1 to 0.4 with the average being 0.2, which yields a standard deviation of 0.05. The relative centrality of a concept always changes with the other document that is used as the reference, unless it is not related to any other concept in the entire dataset. In fact, Figure 5.4 shows that none of the concepts has only one relative centrality value across the dataset: difference is always greater than zero. For example, the relative centrality of *export* varies between 0 and 0.42 with an average of 0.13 and a standard deviation of 0.05.

Figure 5.4 shows that concepts indeed tend to appear in diverse contexts and their importance tends to change when comparing to different contexts, a fact that the traditional plain representation models and clustering processes fail to capture. The distinct distributions also suggest that different behaviours are to

be expected for reweighting by local and relative centrality when evaluating them in text clustering.

5.4 Beyond surface similarity

Cosine similarity and many other inter-document similarity measures assume that features are orthogonal: only surface overlaps—cooccurring words or concepts—are counted, while semantic connections between features are ignored. For example, Table 5.2(b) clearly shows that *cardiovascular disease* is related to document D3: with a context centrality value of about 0.53. However, since it is not mentioned in D3, it will contribute nothing to the similarity between D2 and D3.

This section explores methods for enriching document similarity measures with relatedness between concepts, so as to enable standard clustering algorithms to connect texts on related topics that nevertheless do not mention the same concepts. We discuss existing methods in the next section and present Katoa’s method in Section 5.4.2.

5.4.1 Existing methods

In general, methods for enriching document similarity with semantic relations can be categorized into two types.

The first type expands a text’s representation to new concepts based on those already mentioned in that text. Such extension is usually made to more generic concepts such as hypernyms of existing ones (Bloehdorn and Hotho, 2004; Recupero, 2007) or concepts that are closely related to them (Hu et al., 2008). For example, document D1 mentions *cardiovascular disease* and D3 mentions *coronary heart disease*, both of which belong to the same Wikipedia category: *cardiovascular diseases*. By expanding to this category, we can bridge the different surface concepts and relate the two documents. Hu et al.’s system, which is discussed in the previous chapter, automatically expands representation with closely related concepts, which could be *cardiovascular system*, *smoking* and *heart disease* for both D1 and D3. Both methods need to restrict the expansion somehow, and only concepts within a certain range are considered: for example a certain depth

in the hierarchy for the former and a heuristically selected list of related concepts for the latter.

The second type focuses on the relative comparisons, and only considers concepts that are relevant to connecting a pair of documents. This is to avoid a shortcoming of the first method: expansion is performed without any context. For example, based on the concept *smoking*, which is literally mentioned in document D1, the above method can expand to its hypernyms like *addiction* and *habits*, and its closely related concepts like *tobacco*, *cigarette*, and *nicotine*. However, most of the expanded concepts are irrelevant for connecting document D1 with D3, which discusses a *coronary heart disease*: it is unlikely that document D3 will be augmented with any of these concepts. In contrast, the most relevant concepts for connecting these two documents are those mentioned in one but not the other. For example, none of the concepts in document D1—*cardiovascular disease*, *pharmaceutical drug*, *tobacco smoking* and *weight loss*—is mentioned in document D3, and neither do the nine concepts in D3. Only taking these thirteen concepts into account will solve the orthogonality problem: D3 will be enriched with the four concepts from D1 and vice versa for D1. Concept relatedness—connection between features—is considered in determining the weights of the enriched concepts.

5.4.2 Katoa’s semantically enriched document similarity

We consider the second approach, for two reasons. First, it does not require any pre-specified parameters such as those for the first method: the maximum depth to search for general concepts or a list of related concepts for each concept. Second, it does not require the concept system to have a hierarchical structure of generic relations. Both contribute to the method’s generality. Although WordNet and Wikipedia both provide a generic-relation-based hierarchy, the method will be more generic if it does not presume so.

Before computing the similarity between two documents, each is first enriched with concepts from the other that are missing in the current one, and the document being enriched is referred to as the *target document*. Concepts that occur in both are not considered in the enriching process. Next, a weight is determined

for each enriched concept, based on two factors: its most related concept in the target document and its relatedness with the whole document.

Formally, assume we are given two documents d_A and d_B with the sets of concepts C_A and C_B . We first enrich C_A with concepts from d_B that are not mentioned in d_A . For each such concept c_e ($c_e \in C_B$ and $c_e \notin C_A$), the first component—its strongest connection with C_A —is denoted by c_e^A , that is, $c_e^A = \max_{c \in C_A} rel(c_e, c)$, and the second component—its relatedness with the whole document—is its centrality with C_A : $CC(c_e, d_A)$. Thus the enriched concept c_e 's weight in d_A is

$$w_e(c_e, d_A) = w(c_e^A, d_A) \times rel(c_e, c_e^A) \times CC(c_e, d_A),$$

where $w(c_e^A, d_A)$ is c_e 's most related concept c_e^A 's weight in d_A , and $rel(c_e, c_e^A)$ is their relatedness. Then document d_B is enriched in the same way with concepts from d_A that are missing in d_B .

For example, documents D2 and D3 will be augmented with a non-zero weight for *cardiovascular disease* when they are compared with document D1, which mentions the concept. *Cardiovascular disease*'s most related concepts are *tobacco smoking* for document D2 with a relatedness value of 0.64, and *coronary heart disease* for D3 with a relatedness value of 0.71 (see Table 5.2). Thus its weight for the former is $1 \times 0.64 \times 0.543 = 0.348$ and $1 \times 0.71 \times 0.526 = 0.373$ for the latter. Here 1 is the original weight of its most related concept, while 0.543 and 0.526 are its binary relative context centrality with documents D2 and D3 respectively.

The two components of an enriched concept's weight—its strongest semantic connection and context centrality with the document—are both necessary. The former represents the most likely strength of the connection between these two documents regarding the enriched concept, while the latter adjusts it by considering how important the connection is for the target document. We will show that both contribute to the enriched measure's effectiveness in clustering in Section 5.6.3.

Figure 5.5 compares the similarity matrix of the three documents in Figure 5.1 calculated without (Figure 5.5(a)) and with the enriching process (Figure 5.5(b)). The plain method uses the cosine rule, which only counts surface overlaps, there-

	D1	D2	D3		D1	D2	D3
D1	1.0	0.124	0	D1	1.0	0.688	0.521
D2	0.124	1.0	0	D2	0.688	1.0	0.436
D3	0	0	1.0	D3	0.521	0.436	1.0

(a) plain
(b) enriched

Figure 5.5: Plain and semantically enriched document similarity

fore documents have low similarities. In contrast, they become more similar after taking the semantic connections among concepts into account, and none of them is completely different to another—with a zero similarity—any more. This reflects the fact that they all discuss the same topic *smoking and health* after all, but from different perspectives. Furthermore, the relative similarity remains the same: those that have some surface overlaps, such as documents D1 and D2, still receive a higher similarity than those that do not, like documents D2 and D3.

5.5 Experimental design

The following experiments will investigate whether integrating concept relatedness can further enhance concept-based text clustering. The baseline to compare with is the plain method presented in the previous chapter—the two concept-based representation models with standard clustering processes (i.e., results in Section 4.4). The purpose is to test whether the three enriched clustering methods presented in the previous sections can further improve the plain representation models and clustering methods, and how.

We use the four datasets introduced in Section 3.1 and the two clustering algorithms: *k*-means and the hierarchical agglomerative clustering with group-average-link, because they are significantly more effective than the others (see Section 4.4.5). *Term frequency* \times *inverse document frequency* (*tf* \times *idf*) weighting is performed after the reweighting and enriching processes, i.e., upon the adapted weights (from term frequencies). Clusters are compared against the existing category structures, with the exact number of clusters. The overall cluster quality is quantified with the normalized mutual information measure.

As discussed, context centrality can be computed as either a *binary* measure:

considering a concept’s presence and absence, or a *weighted* measure: considering its number of occurrences in a text. This affects all of Katoa’s methods for employing concept relatedness—reweighting by local context centrality (denoted by Local), reweighting by relative context centrality (denoted by Relative) and applying the enriched similarity measure (denoted by Enriched). This results in two versions of each method, which are then compared with the plain method for the two representation models, on the four experimental datasets and using the two clustering methods.

5.6 Experimental results

We first provide an overview of each method’s effectiveness by summarizing the number of times they successfully outperform the baseline plain method. Then we drill down for each clustering algorithm and each dataset, to provide a better understanding of the situations in which they are effective.

Recall that the enriched measure comprises two components. We evaluate each component individually in Section 5.6.3. Section 5.6.4 compares the binary and weighted versions of each method, investigating whether it is necessary to take the number of occurrences of a concept into account when calculating context centrality.

5.6.1 Overall effectiveness

Table 5.3 shows how often each method is able to obtain a cluster structure that is statistically significantly better and worse than the baseline plain method. Each cell in Table 5.3 involves eight comparisons: four datasets times two clustering algorithms, and in total 96 comparisons: eight comparisons times three methods times two versions (binary and weighted) times two representation models.

Statistical test is performed with paired t -test and $p = 0.05$ (see Section 3.2). For k -means, statistical significance on each dataset is established on clustering results of five runs of k -means, each with different random sets of documents as initial seed clusters. For hierarchical clustering, which is deterministic, significance test uses results obtained by varying the number of clusters from 5 to 60. A win

	Local		Relative		Enriched		Sum
	Binary	Weighted	Binary	Weighted	Binary	Weighted	
WordNet	3–0	1–2	2–3	1–2	0–6	0–7	7–20
Wikipedia	8–0	2–3	1–4	1–5	4–1	6–0	22–13
Sum	11–0	3–5	3–7	2–7	4–7	6–7	

Table 5.3: Overall performance of semantically enriched clustering methods

for the enriched method equals a loss for the baseline plain method. Table 5.3 shows both the wins and the losses for the enriched methods (i.e., wins–losses).

Results in Table 5.3 clearly identify two trends: reweighting with local context centrality is the most effective method among the three, and all methods are more effective with Wikipedia concepts. The success of local centrality indicates that the plain concept-based representations are indeed quite comprehensive in capturing topic-related information, which provides a strong basis for the reweighted model to be even more discriminative.

Reweighting by relative context centrality is surprisingly the least effective, with only five improvements in total. This is quite disappointing, especially consider that relative centrality directly depicts the thematic connections between two texts. One possible reason is that although relevant concepts are emphasized, they are not counted if not mentioned in both texts, due to the orthogonality between features. Meanwhile, weights of the overlapping concepts are nevertheless reduced, by multiplying by their centrality, which is a score between zero and one. Take *tobacco smoking* for example, which is mentioned in both documents D1 and D2. Before reweighting, it contributes 1 to their similarity, which is reduced to 0.629 after reweighting (see Table 5.2).

The decrease in the weights of the overlapping concepts will be compensated if the resemblance between distinct concepts is also taken into account. We will show in Section 5.6.3 that relative context centrality is indeed effective when this requirement is met.

Results in Table 5.3 advocate Wikipedia as a more effective concept system for concept-based text clustering: the three methods improve the baseline twice more often with Wikipedia concepts than with WordNet concepts. Furthermore, recall that in the previous chapter we found that the Wikipedia based representation

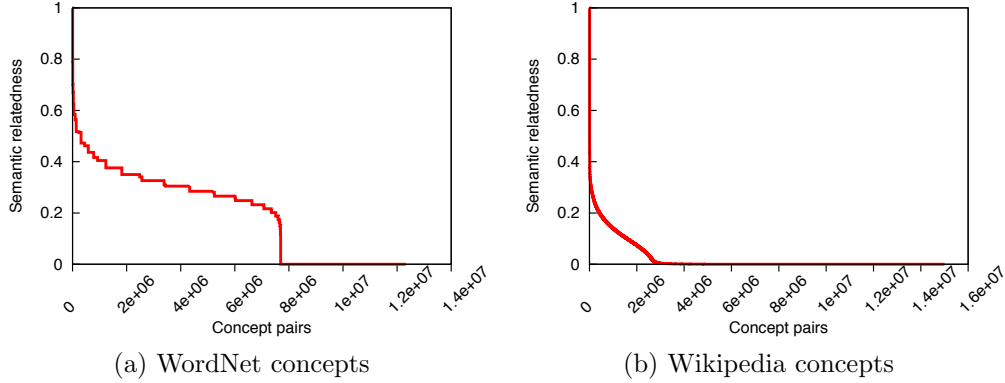


Figure 5.6: Distribution of concept relatedness on the SmallReuters dataset

model outperformed its WordNet counterpart on three of the four datasets, which further confirms the preference towards using Wikipedia.

The gap between Wikipedia and WordNet can be explained from two aspects. First, a considerable proportion of WordNet concepts are not topic-related, yet they are mentioned repeatedly throughout a document. For example, repeated occurrences of common expressions like *mention* bias the calculation of context centrality. Topic-related concepts will be underweighted because of their low relatedness to these concepts, which adversely impacts clustering.

The second reason concerns the accuracy of the concept relatedness measure. When assessing against human judgement on concept relatedness, WLM is twice more accurate than LCH (Strube and Ponzetto, 2006; Milne and Witten, 2008a). Empirically, for example, WLM predicts that *transportation* is not related to *mention* at all, while LCH predicts that they are related: with a relatedness of 0.31. Less accurate relatedness also biases the calculation of context centrality, which affects the overall effectiveness of the three methods with WordNet concepts.

Figure 5.6 compares the distributions of concept relatedness on the SmallReuters dataset. Figure 5.6(b) shows that the majority of Wikipedia concepts are regarded as unrelated with WLM: concept pairs with a relatedness less than 0.1 account for about two thirds of all pairs. In contrast, with LCH about half of all pairs have a non-zero relatedness, and over 90% of them vary between 0.2 and 0.6, which is less realistic. This is because LCH is based on path length, and only when such a path does not exist the relatedness is zero. However, concepts

	Dataset	Baseline	Local		Relative		Enriched	
			Binary	Weighted	Binary	Weighted	Binary	Weighted
WordNet	SmallReuters	0.697	0.704*	0.702	0.704	0.694	0.620 \circ	0.621 \circ
	Med100	0.211	0.193	0.195	0.188 \circ	0.189	0.189 \circ	0.168 \circ
	NewsSim3	0.056	0.071	0.055	0.080	0.034	0.010	0.015
	NewsDiff3	0.767	0.817*	0.774*	0.818*	0.771	0.44 \circ	0.29 \circ
Wikipedia	SmallReuters	0.704	0.724*	0.713*	0.690	0.702	0.686 \circ	0.696
	Med100	0.284	0.311*	0.305	0.260 \circ	0.249 \circ	0.323*	0.325*
	NewsSim3	0.248	0.261*	0.245	0.239	0.234	0.136	0.132
	NewsDiff3	0.579	0.594*	0.578	0.213 \circ	0.231 \circ	0.655*	0.682*

*, \circ : statistically significant improvements and degradations

Table 5.4: Performance of semantically enriched clustering with the k -means clustering algorithm

can always reach each other as long as they belong to the same taxonomy (i.e., have the same part of speech), resulting in a large number of non-zero relatedness scores.

5.6.2 Effectiveness with different clustering algorithms

Tables 5.4 and 5.5 show the detailed performance of the enriched clustering methods with k -means and hierarchical agglomerative clustering respectively. The best result on each dataset is noted in bold.

These results show that a method’s effectiveness is highly correlated with the clustering algorithm in use. For example, the enriched document similarity (the weighted version) consistently improves the baseline with hierarchical agglomerative clustering on all datasets when using Wikipedia-based representation model, while its successes with k -means are mixed. In contrast, reweighting by binary local context centrality turns out to be more effective with k -means.

This is probably because k -means represents a cluster with its centroid, which is the mean vector of all the component documents of that cluster. A cluster centroid thus has much more concepts than a normal document would have. For example, an average document in the SmallReuters dataset has 15 concepts, while an average cluster centroid has about 420 concepts. Each document, when compared with a cluster centroid, will be enriched with all concepts in that cluster

			Local		Relative		Enriched		
Dataset			Baseline	Binary	Weighted	Binary	Weighted	Binary	Weighted
WordNet	SmallReuters	0.662	0.63	0.604	○	0.623	0.617	0.592	○ 0.579
	Med100	0.269	0.245	0.224	○	0.246	○ 0.234	0.195	○ 0.177
	NewsSim3	0.081	0.075	0.075		0.065	○ 0.059	0.035	0.008
	NewsDiff3	0.524	0.636	*	0.505	0.621	*	0.573	*
Wikipedia	SmallReuters	0.623	0.641	*	0.604	○	0.618	0.604	○
	Med100	0.291	0.309	*	0.271	○	0.267	○ 0.272	○
	NewsSim3	0.098	0.102	*	0.078	○	0.128	*	0.1
	NewsDiff3	0.38	0.394	*	0.384	*	0.235	○ 0.231	○

*, \circ : statistically significant improvements and degradations

Table 5.5: Performance of semantically enriched clustering with hierarchical agglomerative clustering with group-average link

unless the enriched concept is unrelated to any concept mentioned in the document, resulting in a drastic increase in dimensionality. Besides, centrality with respect to a cluster centroid is likely to be excessively low, due to the large number of concepts. Both factors potentially incur the inferior performance of the enriched document similarity measure with k -means.

These results suggest that one needs to consider both the clustering algorithm and the concept system, for an effective utilization of concept relatedness in clustering. We summarize these findings as follows:

- With Wikipedia:
 - Adapting concept weight with (binary) local context centrality always improves clustering, regardless of the clustering algorithm.
 - The (weighted) enriched similarity measure is effective, especially when the hierarchical agglomerative clustering with group-average-link is used.
- With WordNet:
 - The plain method—the unweighted concept-based representation and the cosine similarity—is usually a safer choice.
 - Reweighting by either local or relative centrality (binary versions) is likely to be effective when used with the k -means clustering algorithm.

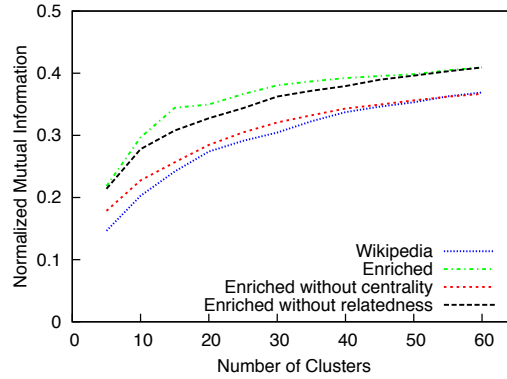


Figure 5.7: Performance of the enriched similarity measure’s components on the Med100 dataset

5.6.3 Effectiveness of the enriched document similarity

This evaluation investigates the effectiveness of the enriched document similarity measure’s two components: a concept’s strongest connection and its centrality with the text being enriched. This involves three schemes: the standard measure, the measure without considering context centrality, and the measure without considering the most related concept. The baseline to compare with is the plain clustering method.

Figure 5.7 shows the performance of the weighted versions of these schemes on the Med100 dataset, with Wikipedia concepts and using hierarchical agglomerative clustering. Similar results exist for the other datasets. It shows that both components are necessary, and context centrality is especially important: cluster quality is substantially improved when it is taken into account.

This is because the relatedness between concepts needs to be adjusted based on their surrounding contexts. For example, *tobacco smoking* and *cigarette* are considered as closely related in general. However, if *cigarette* is mentioned in a text on *cigarette trading*, which has little to do with *smoking*, the general strength of this connection is weakened in this particular context. Relative centrality comes in naturally and captures exactly this variation of strength. In particular, if a concept is completely unrelated to the enriched text—with zero relatedness to every concept mentioned in that text—it will not be augmented at all due to a zero centrality.

5.6.4 Binary and weighted schemes

Results in the previous sections show mixed performance of the binary and weighted versions of context centrality. To investigate their comparative effectiveness, Table 5.6 summarizes how often one version achieves a clustering that is statistically significantly better than the other. The purpose is to demonstrate one scheme’s relative performance with respect to the other, therefore, they are *not* compared to the baseline like in Table 5.3.

Each cell involves eight comparisons: four datasets times two clustering algorithms; thus the upper bound is eight for each cell of Table 5.6. Statistical significance test is performed in the same way as described in Section 5.6.1: for k -means, it uses clustering results of five runs of k -means; and for the deterministic hierarchical clustering, it uses results obtained by varying the number of clusters from 5 to 60. A win for one scheme equals a loss for the other, and Table 5.6 only shows the wins.

The trend is consistent across datasets and for both Wikipedia and WordNet: the binary version is more effective, especially with WordNet concepts. WordNet’s lexical characteristic again provides the explanation: lexical features that are unrelated to topics can bias the calculation of context centrality, and including their frequencies enlarges this bias.

There exists a notable exception though: the enriched similarity measure performs better when the occurrence frequencies are counted and Wikipedia is used. This indicates that a concept’s occurring frequency is indeed desirable for an effective measure of document similarity, when the representation model and the concept relatedness measure are accurate. These empirical findings provide valuable guidance for using Katoa’s methods in practice.

	Local		Relative		Enriched	
	Binary	Weighted	Binary	Weighted	Binary	Weighted
WordNet	5	0	6	0	4	0
Wikipedia	5	0	4	2	0	5

Table 5.6: Relative performance of the binary and the weighted schemes of the semantically enriched clustering methods

5.7 Summary

This chapter investigates whether utilizing semantic concept relatedness in clustering can further enhance the plain concept-based text clustering, and introduces the three enriched clustering methods in Katoa. Empirical experimental results show that two of them are effective—reweighting concepts by their centrality with the surrounding context, and extending the document similarity measure beyond surface overlap. They also show that both factors—concept system and clustering algorithm—impact the effectiveness of the enriched methods.

In general, with Wikipedia concepts and hierarchical clustering methods, two of the three methods consistently improve upon the baseline: reweighting with local centrality and using the enriched similarity measure. These methods operate on the basis of a normal document, which influences their effectiveness with the k -means algorithm, due to its mean vector representation of cluster centroids. Despite that, the two methods still improve the baseline across all datasets, even when using k -means, especially the former.

From the efficiency perspective, reweighting by local centrality is the most efficient method among the three. It only requires scanning each text in a given collection once, and every cluster centroid generated in the process of k -means clustering. Reweighting by relative centrality and the enriched document similarity are more expensive, and have similar time complexity, which mainly comes from computing the centrality of a concept with each text in the collection, or each cluster centroid if k -means is used.

This results in computing $|D| \times |C|$ centrality values for hierarchical clustering, where $|D|$ and $|C|$ denote the total number of texts and concepts in a collection, and $k \times iter \times |C|$ for k -means clustering, where k is the number of clusters and $iter$ is the number of iterations before convergence, which is always less than 20 in our experiments.

Take the largest experimental dataset—NewsDiff3—for example. It contains 2780 documents and about 13,000 concepts, resulting in about 36 million and 780,000 computations for hierarchical clustering and k -means respectively. This takes about one hour and one minute respectively, given ten computations per millisecond. For hierarchical clustering, the relative context centrality values can

be cached so that they will not be computed repeatedly during the clustering process.

Recall that we discussed several competing methods in the previous chapter, which already utilize concept relatedness in their clustering process (see Section 4.4.3). Although Katoa’s plain representation models and clustering method already outperform these methods, integrating concept relatedness obtains even greater advantages.

6

Learning document similarity

Accurate clustering requires a precise definition of closeness: how close objects—texts in this case—are to each other. Choosing the right similarity measure is no less important than choosing a good representation (Hartigan, 1975). For example, we showed in the previous chapter that taking concept relatedness into account contributes to a more accurate measure for assessing the similarity between two texts based on their topics.

Similarity measures for texts are usually designed based on empirically observed psychological properties (M. D. Lee et al., 2005) or on mathematical analysis of the texts (Baeza-Yates and Ribeiro-Neto, 1999). In general, what these measures try to achieve is a mapping from the relations between a pair of texts to a numeric score, which usually varies between 0, meaning they have completely different topics, and 1, meaning the same topics. Instead of handcrafting a measure using expert knowledge, such mappings can be automatically *learned* from a group of texts whose similarity is already known. Machine learning provides off-the-shelf tools for automatically constructing this kind of mapping: no expert knowledge is required.

There are many aspects to consider when describing the relation between two texts. The proportion of cooccurring surface forms is the most common, and usually the only feature considered. The concept-based representation models and methods for integrating concept relatedness describe the relations between texts from different angles, and thus provide new perspectives. For example, the distribution of local and relative context centrality in each text indicates how

coherent two texts are with respect to each other, and thus can be considered as one perspective on the thematic similarity between texts.

The diversity of information channels highlights another advantage of using machine learning: it can automatically learn the best way to combine these channels. We do not need to decide how the distinct aspects should be combined, but can let the algorithm *learn* this from training examples—pairs of texts whose similarities are already known.

It is expected that learned measures can be more effective than traditional ones, which usually consider only a single aspect of thematic similarity between texts. However, evaluating the quality of a similarity measure is a subjective and challenging task. Generally there are two kinds of evaluation. One is to assess directly against human judgement: a measure is considered good if its predictions are consistent with human rated scores. The other is to evaluate the measure in a target application: an accurate measure results in superior performance of the application. This leads to the two hypotheses that this chapter investigates, each corresponding to one type of evaluation:

with machine learning, the learned measure can predict thematic similarity between texts as consistently as human judgement

and

with machine learning, the learned measure is more effective than the cosine measure in text clustering.

Here the cosine measure is chosen as a representative handcrafted measure because it is one of the most widely applied measures, and usually at least as effective as other common measures (Willett, 1983; M. D. Lee et al., 2005; Strehl et al., 2000).

Learning the similarity measure requires training data—pairs of texts whose similarities are already known. Such data is rare, and Section 6.1 introduces the dataset used in this research. Section 6.2 discusses possible features for describing thematic similarity between texts, while Section 6.3 explains the machine learning algorithms that Katoa uses. Two evaluations are conducted: the learned similarity measure is compared with human judgement in Section 6.4, and used for clustering in Section 6.5. Section 6.6 compares our method with related work; Section 6.7 summarizes the chapter.

6.1 Manually assigned document similarity

Training data is crucial, but like all manually labelled data, it is difficult and expensive to obtain. There is little currently available data on manually rated thematic similarity between texts. We know of only one dataset with a substantial number of human raters, collected by M. D. Lee, Pincombe, and Welsh in 2005 (M. D. Lee et al., 2005; Pincombe, 2004), referred to hereafter as the HE50 dataset.

HE50 consists of fifty short news documents from August 2002, selected from a group of articles taken from the Australian Broadcasting Corporation’s news mail service, which provides text e-mails of headline stories. These documents were chosen so that some topics have identifiable *sub-topics*; for example, the articles on Australian politics contain a grouping on an ongoing argument amongst the Democrats (Pincombe, 2004). Different sub-topics are likely to use different yet semantically related expressions, thus this particular design captures how humans relate documents that are literally different.

These documents are quite short—between 51 and 126 words each. In total, they contain 1583 distinct words after converting to lower case. Assessments of word distribution showed that they are normal English documents (Pincombe, 2004). The documents were paired in all possible ways, generating 1225 pairs, excluding 50 self pairs.

The judges were 83 students from the University of Adelaide, Australia. Document pairs were presented in random order, and the order of documents within each pair was randomized as well. The students rated the pairs on a five-point scale: 1 indicating highly unrelated and 5 indicating highly related. Each pair received eight to twelve valid human judgements. Judgements were averaged, giving a collection of 1225 relatedness scores.

Consistency is assessed in terms of Pearson’s linear correlation coefficient. M. D. Lee et al. show that the human raters’ judgements are quite consistent throughout the task and with each other: on average, a human rater has 0.6 correlation with others. This dataset has become the benchmark for evaluating document similarity measures, and several recent results (Gabrilovich and Markovitch, 2005; Yeh et al., 2009) are reviewed in Section 6.6. The consistency

between human raters has served as the baseline for assessing automated measures, and the aim is to make automated measures as consistent with humans as humans are among themselves.

6.2 Features

In principle, any aspect that describes the thematic connections between two texts can become a feature in a machine learning setting. This section explores them, considering which ones a pair of texts has that might differ from those present in another pair. Not all features explored here can distinguish one pair of texts from another. Our aim is to identify distinctive ones, so that the learning algorithm can utilize them to model human judgement.

Before further discussion, we need to clarify the terminology used in this chapter. We use *feature type* to denote a category of *features* that describe the same aspect of thematic similarity between texts. Each feature type consists of several *features*, reflecting the various perspectives it encompasses. Each feature is expressed with one or two *attributes* in the vector that represents a pair of texts, each attribute corresponding to one dimension. In contrast, in the previous chapters we use *feature* instead of *attribute* to denote the dimensions.

Table 6.1 lists four feature types and eighteen features used for learning. The first and the most straightforward type measures similarity between two texts with different representations and different similarity measures, and is discussed in Section 6.2.1. As discussed precedingly, the local and relative context centralities of concepts reflect the coherence of each text and the relation between two texts respectively, which can be used for learning similarity, and Section 6.2.2 explains the features derived from them. Whereas the centrality measures are based on *one-to-many* relations between concepts: how related one concept is with respect to a group of concepts, Section 6.2.3 investigates *one-to-one* concept relations by examining the strongest connection between texts. Section 6.2.4 considers the fact that topics are usually expressed by groups of closely related concepts and analyzes features that describe these concept groups.

Feature Type	ID	Feature	Number of attributes
Overall similarity	F1	CosineWords	1
	F2	CosineConcepts	1
	F3	EnrichedConcepts	1
Context centrality	F4	MaxLocal	2
	F5	MinLocal	2
	F6	AvgLocal	2
	F7	SDLocal	2
	F8	MaxRelative	2
	F9	MinRelative	2
	F10	AvgRelative	1
	F11	SDRelative	2
Strongest connection	F12	MaxRel	1
	F13	MaxNOrel	1
Concept groups	F14	AvgGroupSize	2
	F15	MaxGroupRel	1
	F16	MinGroupRel	1
	F17	AvgGroupRel	1
	F18	SDGroupRel	1
Total	F1–F18		26

Table 6.1: Features used for learning document similarity

6.2.1 Overall similarity

Similarity based on cooccurrence is one of the most commonly used features—and usually the only one. Because of this, we take it as a starting point. The cosine similarity measure (see Section 4.3) has been widely applied in many fields. Therefore the first features are similarities measured by the cosine rule with different representation models. Similarity computed with the enriched measure (see Section 5.4) produces another feature that measures similarity beyond the surface forms of texts.

This generates three features: abbreviated *CosineWords*, *CosineConcepts* and *EnrichedConcepts* respectively (i.e., features F1–F3 in Table 6.1). These three features constitute the first feature type, called *overall similarity*, and each feature corresponds to one attribute.

6.2.2 Context centrality

Context centrality consists of local and relative centrality. In the previous chapter they are computed and utilized based on single concepts: by reweighing a concept with its centrality. For the machine learning algorithm to be generic, the centrality of each concept needs to be generalized, so as to avoid undesirable dependency on a specific vocabulary. Thus we derive several features that describe the overall distribution of concepts' local and relative centrality, given a pair of texts. For example, if two texts share similar topics, a considerable proportion of the concepts are likely to have high relative centrality values, resulting in a high average value and a low standard deviation.

Four features are derived to describe the distribution of centrality values: minimum, maximum, average and standard deviation. The first two are trivial to obtain. The use of *binary* or *weighted* context centrality affects the values of the last two—average and standard deviation (see Table 6.4). The average centrality is the (binary or weighted) mean of the centrality values of all concepts, and the standard deviation is also based on these (binary or weighted) values.

These four features apply to both measures, resulting in eight features respectively: *MaxLocal*, *MinLocal*, *AvgLocal* and *SDLocal* for local centrality (i.e., features F4–F7 in Table 6.1); *MaxRelative*, *MinRelative*, *AvgRelative*, and *SDRelative* for relative centrality (i.e., F8–F11). Each feature corresponds to two attributes, because its value differs for each text. For example, the maximum local and relative centrality differs for the two documents in Figure 6.1. The average relative centrality (i.e., *AvgRelative*) is an exception: only one attribute is needed, because it is symmetric, and is obtained by averaging over all the concepts in both documents (see the definition in Section 5.3.3).

Figure 6.1 shows two documents from the HE50 dataset, the concepts identified from them, and their ranks by (binary) local and relative context centrality. Three concepts from document *B* stand out in the comparison: *sustainability*, the *United Nations* and *earth*, which are noted in bold. These are ranked ninth, sixth and eighth by local centrality, but become the top three concepts when ranked by relative centrality, which indicates that they represent the major connections between these two documents. Figure 6.1 clearly shows again that local centrality

describes characteristics of the document itself while relative centrality describes inter-document connections.

- A The real level of world inequality and environmental degradation may be far worse than official estimates, according to a leaked document prepared for the world’s richest countries and seen by the Guardian. It includes new estimates that the world lost almost 10% of its forests in the past 10 years; that carbon dioxide emissions leading to global warming are expected to rise by 33% in rich countries and 100% in the rest of the world in the next 18 years; and that more than 30% more fresh water will be needed by 2020.

Concept	Occurrences	Local centrality (and its rank)		Relative centrality (and its rank)	
Global warming	1	0.502	(1)	0.298	(1)
Natural environment	1	0.492	(2)	0.242	(2)
Carbon	1	0.353	(3)	0.111	(5)
Fresh water	1	0.331	(4)	0.108	(6)
The Guardian	1	0.289	(5)	0.186	(3)
Inequality	1	0.289	(6)	0.128	(4)

- B Pope John Paul II urged delegates at a major U.N. summit on sustainable growth on Sunday to pursue development that protects the environment and social justice. In comments to tourists and the faithful at his summer residence southeast of Rome, the pope said God had put humans on Earth to be his administrators of the land, "to cultivate it and take care of it." "In a world ever more interdependent, peace, justice and the safekeeping of creation cannot but be the fruit of a joint commitment of all in pursuing the common good," John Paul said.

Concept	Occurrences	Local centrality (and its rank)		Relative centrality (and its rank)	
Pope	2	0.427	(1)	0.067	(9)
Pope John Paul II	1	0.374	(2)	0.081	(8)
Social justice	1	0.37	(3)	0.244	(4)
God	1	0.369	(4)	0.171	(6)
Rome	1	0.337	(5)	0.039	(10)
United Nations	1	0.33	(6)	0.267	(2)
Common Good	1	0.297	(7)	0.127	(7)
Earth	1	0.278	(8)	0.255	(3)
Sustainability	1	0.268	(9)	0.411	(1)
Fruit	1	0.148	(10)	0.187	(5)

Figure 6.1: Local and relative context centrality of concepts in two sample documents from the HE50 dataset

6.2.3 Strongest connection

The centrality measures assess relations between one concept and a set of concepts. For example, maximum centrality measures the strongest *overall* relatedness to a group of concepts. Besides these one-to-many relations, one-to-one relations also provide useful information about how similar two texts are.

We define two such features: the maximum relatedness between single concepts in two texts, abbreviated *MaxRel*, and the maximum relatedness between concepts that appear in one text but not the other, i.e., between non-overlapping concepts, abbreviated *MaxNORel*—F12 and F13 in Table 6.1 respectively. For texts that have at least one concept in common, *MaxRel* is 1; otherwise *MaxRel* equals *MaxNORel*.

For example, the strongest connection between document A and B in Figure 6.1 is between *natural environment* and *sustainability*, whose relatedness value is 0.63. Because the two documents have no concept in common, *MaxRel* and *MaxNORel* both have value 0.63 for this example.

6.2.4 Concept groups

Concepts mentioned in the same text are not only related but can form their own structures: highly related concepts are often used together when describing a topic that they all relate to. Suppose a document explains *oil spill*. It might mention some alternative names of refined *oil* (such as *petroleum*, *gasoline*, *diesel*), some oil companies (such as *Shell* and *BP*), and oil's influence on species like *seabirds*, and *marine mammals*. In this example, these concepts can be organized into three groups: one with *oil spill*, *oil* and its alternative names; one with the oil companies; and the other with *seabirds* and *marine mammals*. Each group represents a more subtle and detailed aspect of this document's topic. Documents that share similarity in either aspect are somewhat similar to the current one, and those that mention all aspects are even more alike.

Concepts can be clustered based on their relatedness to each other, so that closely related ones are organized into the same group while those with low relatedness are separated into different groups. Each concept group, such as those shown above, reflects a topic or a subtopic; thus they can be used to model

thematic connections. This provides features that describe inter-document relations at the *topic* level, which is intermediate between the document and concept levels examined previously.

Specifically, concepts are clustered to form *cliques*—complete subgraphs—in order to make the topics (or subtopics) as coherent as possible. Again, concepts are modelled by a weighted undirected graph, with concepts as vertices. Unlike the graph used for modelling context centrality described in the previous chapter, where every concept is connected with the others in the graph, here only those whose relatedness exceeds a certain threshold are connected. The maximal cliques of this graph give the concept groups we seek. Every pair of concepts assigned to the same group exceeds this threshold, and no other concept can be added to any of these groups.

For example, Figure 6.2 shows the groups with at least two concepts identified from the two example documents with a relatedness threshold of 0.5. Only one group is found in document *A*, and it is only highly related to one of the three groups in document *B*—the third group consisting of *sustainability* and *social justice*. *Pope* appears in two of the three groups in document *B*, because *Rome* and *Pope John Paul II* are insufficiently related for the two groups to be merged.

Document	Concept groups
A	{Global warming, Natural environment}
B	{Pope, Rome}
	{Pope, Pope John Paul II}
	{Sustainability, Social justice}

Figure 6.2: Concept groups in the example documents in Figure 6.1

Several features can be derived from these concept groups. The *averaged group size* of each document is abbreviated *AvgGroupSize* (i.e. feature F14 in Table 6.1). The relatedness between concept groups results in four features that describe the distribution: its maximum, minimum, average, and standard deviation, abbreviated *MaxGroupRel*, *MinGroupRel*, *AvgGroupRel* and *SDGroupRel* respectively (i.e. F15–F18).

There are two ways to treat concepts that cannot be assigned to any groups:

they can either form a singleton—a group by itself—or be ignored. This results in two schemes called *full* and *strict* respectively. For example, document A has one group in the *strict* scheme and five in the *full* scheme: four singletons and one with two concepts.

Generally, given documents d_A and d_B , let the concept groups identified from them be $G_A = \{\varsigma_1, \dots, \varsigma_p\}$ and $G_B = \{\varsigma_1, \dots, \varsigma_q\}$, where ς represents a concept group and p and q are the total number of concept groups for d_A and d_B respectively. The relatedness between ς_h from d_A and ς_l from d_B is calculated as:

$$rel(\varsigma_h, \varsigma_l) = \frac{\sum_{c_i \in \varsigma_h} \sum_{c_j \in \varsigma_l} w(c_i, d_A) \times w(c_j, d_B) \times rel(c_i, c_j)}{|\varsigma_h| \times |\varsigma_l|},$$

where $|\varsigma|$ refers to the size of group ς and is calculated as $\sum_{c \in \varsigma} w(c, d)$. Here $w(c, d)$ is concept c 's weight in document d that produces ς .

The relatedness between two concept groups is also affected by whether the occurrence frequencies of concepts are taken into account or not, i.e., whether the *binary* or the *weighted* scheme is used. Thus $w(c, d)$ equals 1 if c is mentioned in d and 0 otherwise in the binary scheme, and it equals the number of occurrences of c in d in the weighted scheme.

The average group relatedness is the mean of every possible pair of groups weighted by each group's size, formally:

$$grouprel(d_A, d_B) = \frac{\sum_{\varsigma_h \in G_A} \sum_{\varsigma_l \in G_B} rel(\varsigma_h, \varsigma_l) \times |\varsigma_h| \times |\varsigma_l|}{\sum_{\varsigma_h \in G_A} |\varsigma_h| \times \sum_{\varsigma_l \in G_B} |\varsigma_l|}.$$

If no concept group is found for a document (in the *strict* scheme), the average group relatedness is set to -1 , to differentiate this from the case where none of the groups are related, that is, when $grouprel(d_A, d_B)$ equals 0.

6.3 Learning algorithms

Now we investigate machine learning techniques and algorithms for learning document similarity. The learning problem is to find a mapping from a set of features, represented by a vector of attributes with numeric values, to a score in $[0, 1]$. *Regression* is a technique for learning a mapping that predicts a numeric quantity, and the learning outcome is called a *regression model*. In this case, the input to the regression model is a vector with numeric values representing the relation between a pair of texts and the output is a prediction in $[0, 1]$, 1 meaning that the two texts have the same topics and 0 meaning that they have different topics.

Documents from the HE50 dataset are used as training data to train the regression model. Each of the 1225 document pairs is represented by a weighted vector, where each dimension of the vector corresponds to one of the attributes in Table 6.1. The last dimension corresponds to the *label* of the given pair: the average normalized similarity score assigned by human raters. The regression algorithm thus uses the training data to learn the dependencies between the attributes and human judgement.

Both linear and nonlinear regression algorithms can be used for this task. The former assume linear relations between the attributes and the label; whereas the latter assume a nonlinear relation. One linear and three nonlinear regression algorithms (with non-linear kernel functions) are tested: linear regression; support vector machines (SVM) for regression (abbreviated *SVMreg*) (Shevade et al., 2000); *LibSVM*, which uses the *libsvm* (Chang and Lin, 2001) tools to build SVM classifiers; and the Gaussian process for regression (denoted *GaussianProcess*) (MacKay, 1998). All these methods are commonly used techniques and are implemented in Weka (Witten et al., 2011), which is used throughout this thesis. Appendix E describes the configuration for each algorithm.

There are two options when converting a pair of texts into a training example: whether or not to consider each concept's occurrence frequency, which results in the *binary* and *weighted* schemes; and whether or not to create singleton concept groups, the *full* or *strict* schemes. These result in $2 \times 2 = 4$ combinations: *binary strict*, *binary full*, *weighted strict* and *weighted full*. Furthermore, each scheme is applied to the two concept systems (WordNet and Wikipedia) separately, which

Algorithm	Wins	Losses	Wins–Losses
LinearRegression	16	4	12
SVMreg	20	0	20
LibSVM	0	22	–22
GaussianProcesses	6	16	–10

Table 6.2: Relative performance of different regression algorithms

produces eight versions of the training dataset.

These eight datasets were used to test the regression algorithms. Each algorithm is trained and tested with five runs of 10-fold cross-validation on each version: in each run the algorithm is trained on 90% of the data and tested on the remaining 10%. Performance is measured as Pearson’s linear correlation coefficient between the algorithm’s predictions and the gold standard similarities on the 10% testing data. Given two samples X and Y with n values, let \bar{X} and \bar{Y} be their means. Then Pearson’s linear correlation coefficient is defined as:

$$r = \frac{\sum_{i=1,\dots,n} (X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1,\dots,n} (X_i - \bar{X})^2} \sqrt{\sum_{i=1,\dots,n} (Y_i - \bar{Y})^2}},$$

where \bar{X} and \bar{Y} denote the average value of X and Y respectively.

Each algorithm is compared with the other three over eight datasets, resulting in $3 \times 8 = 24$ comparisons in total. A win is counted if one algorithm achieves a statistically significant improvement over another, with a loss counted for the other algorithm. Statistical significance test is performed using the paired t -test with a confidence level $p = 0.05$. Table 6.2 lists the number of wins and losses for the four algorithms, and the difference between the two (Wins–Losses) suggests whether one algorithm should be preferred. Table 6.2 clearly identifies *SVMreg* (with the RBF kernel, see Appendix E) as the most effective regression algorithm, and so it is used for all subsequent experiments.

6.4 Evaluation against human judgement

As mentioned at the beginning of this chapter, there are two ways to evaluate document similarity measures: against human judgement and in a target application. This section reports results of the first evaluation—how consistent the learned measure is with human judgement. Section 6.4.1 discusses the baselines to compare our method with, and Section 6.4.2 describes the experiment setup. Section 6.4.3 investigates the predictive value of each feature individually. Section 6.4.4 selects the informative ones and shows their combined performance.

6.4.1 Baselines

When collecting the HE50 dataset, M. D. Lee et al. (2005) find an average correlation of 0.6 between human raters. This becomes the first baseline in this evaluation, and also the baseline for testing whether or not the first hypothesis stands.

M. D. Lee et al. also test several document similarity measures using the bag-of-words representation. They find only trivial differences between different similarity measures, and the cosine measure yields a correlation of 0.42 with manually assigned similarities. Their best result is achieved using latent semantic analysis (see Section 2.1.2) on a larger collection of 364 documents also from Australian Broadcasting Corporation news. Document vectors are transformed to the new feature space and the cosine measure is used with the new vectors. This technique is as consistent with an average human rater as the human raters are themselves. None of the similarity measures that uses the standard bag-of-words

Baselines	Pearson's Correlation
Inter-rater (M. D. Lee et al., 2005)	0.6
Bag of words (M. D. Lee et al., 2005)	0.42
Latent Semantic Indexing (M. D. Lee et al., 2005)	0.6
ESA (Gabrilovich and Markovitch, 2005)	0.72
ESA-G (Yeh et al., 2009)	0.77

Table 6.3: Performance (consistency with human judgement) of other approaches on the HE50 dataset

representation they test approaches this level.

Two research groups have reported results on the HE50 dataset, summarized in Table 6.3: explicit semantic analysis (ESA) of Gabrilovich and Markovitch (2005), and its improvement ESA-G by Yeh et al. (2009) (see Section 6.6 for more details about these methods). Both use the cosine measure with a Wikipedia concept-based representation model, and both yield a greater correlation than the average between human raters, which means that their methods are more consistent with the average human rater than human raters are among themselves. The result of ESA-G is the best on this dataset so far. These two methods and the inter-rater consistency comprise the three baselines for assessing Katoa’s learned similarity measure.

6.4.2 Experimental design

Each document in the HE50 dataset produced three representations: the traditional bag-of-words model and two concept-based models with WordNet and Wikipedia respectively. Documents were first converted to lower case and stop-words were removed as in M. D. Lee et al. (2005), using the stopword list in Weka, leaving 1378 words in the bag-of-words model, with 38.2 words for each document on average. Words were then stemmed with the Porter’s stemmer (Porter, 1980), further reducing the total number to 1187, with the average document containing 37.1 words.

The concept-based representations were created in the same way as in Chapter 4. In total 492 distinct Wikipedia concepts were identified from this dataset, which is only half of the number of words. Documents are represented by 13.1 Wikipedia concepts on average, from a minimum of five to a maximum of 25 concepts. With WordNet, the number of concepts per document varied between 26 and 63, with 1201 concepts in total and 39.2 concepts per document on average.

Again, all results reported were averaged over five independent runs of 10-fold cross-validation. In each run, the regression algorithm is trained on 90% of the document pairs (1102 examples) and tested on the remaining 10% (123 examples). Performance is measured on the hold-out 10% testing data with the Pearson’s linear correlation coefficient, to indicate the predictive capability of the

learned model on new data.

It is worth noting that the HE50 dataset is very small, which makes it easy to overfit the algorithm to this particular dataset. 10-fold cross-validation is a more appropriate evaluation than the one-off comparison used in previous research (M. D. Lee et al., 2005; Gabrilovich and Markovitch, 2005; Yeh et al., 2009; Stone et al., 2008). It tests on hold-out data, which has not been seen by the algorithm. Averaging over five runs further reduces the dependency of the learned model on the randomization used for 10-fold cross-validation.

6.4.3 Effectiveness of individual features

The value of a feature is measured in terms of how the regression model learned from this single feature alone performs. Specifically, a document pair is converted into a new example that contains only this feature along with the class label. The regression algorithm is then trained and tested with the converted examples, and its performance indicates the predictability of the feature on its own.

Except for the first feature—cosine similarity of the bag-of-words representation, all the other features involve concepts and utilize the relatedness between concepts. In the previous chapter we found that Wikipedia concepts and the WLM measure for concept relatedness are usually more effective than their WordNet counterparts. Comparing Table 6.4 and 6.5 shows the same trend for learning the document similarity measure. Therefore this section focuses on discussing results obtained using Wikipedia as the external concept system.

Overall similarity

The *overall similarity* section of Table 6.4 contains three features. The first two use the cosine similarity measure and the third feature uses the semantically enriched measure. The distinction between *binary* and *weighted* schemes only applies to the enriched measure when computing context centrality, thus only one result is shown for the first two features (F1 and F2). Words and concepts are weighted by their $tf \times idf$ weights. For the enriched measure (F3), term weighting is applied after the enriching process.

Two noticeable trends echo the findings in the previous two chapters. The

Feature Type	ID	Feature (and its number of attributes)	Pearson's Correlation			
			Binary		Weighted	
Overall similarity	F1	CosineWords (1)			0.57	
	F2	CosineConcepts (1)			0.603	
	F3	EnrichedConcepts (1)	0.717			0.710
Context centrality	F4	MaxLocal (2)	−0.039		−0.001	
	F5	MinLocal (2)	−0.043		0.038	
	F6	AvgLocal (2)	0.374		0.045	
	F7	SDLocal (2)	0.022		0.004	
		LocalCentralityCombined (8)	0.155		0.174	
	F8	MaxRelative (2)	0.691		0.685	
	F9	MinRelative (2)	0.703		0.707	
	F10	AvgRelative (1)	0.327		0.320	
	F11	SDRelative (2)	0.679		0.657	
		RelativeCentralityCombined (7)	0.725		0.711	
Strongest connection	F12	MaxRel (1)			0.62	
	F13	MaxNORel (1)			0.643	
		MaxRelatednessCombined (2)			0.688	
Concept groups			Strict	Full	Strict	Full
	F14	AvgGroupSize (2)	0.176	0.137	0.176	0.137
	F15	MaxGroupRel (1)	0.655	0.481	0.655	0.489
	F16	MinGroupRel (1)	0.002	0.001	0.001	0.001
	F17	AvgGroupRel (1)	0.664	0.608	0.674	0.665
	F18	SDGroupRel (1)	0.474	0.618	0.451	0.624
		GroupRelatednessCombined (4)	0.7	0.689	0.703	0.718
Combined		F1–F18 (26)	0.809	0.799	0.808	0.8

Table 6.4: Predictive value of features generated with Wikipedia concepts

gap between F1 and F2 indicates that the concept-based representation is more discriminative than the bag-of-words model. Comparing F2 and F3 identifies the enriched measure as more effective than the overlap-based cosine measure. This difference is remarkable because the dimensionality of the concept-based representation is much lower than that of the bag-of-words model in this case: the number of distinct concepts is less than half of that of words. Furthermore, both improvements—F2 over F1 and F3 over F2—are statistically significant in both the binary and the weighted schemes.

However, Table 6.5 shows contrary results with WordNet. This is because

6.4. EVALUATION AGAINST HUMAN JUDGEMENT

Feature Type	ID	Feature (and its number of attributes)	Pearson's Correlation			
			Binary		Weighted	
Overall similarity	F1	CosineWords (1)			0.57	
	F19	CosineConcepts (1)			0.517	
	F20	EnrichedConcepts (1)	0.342		0.314	
Context centrality		LocalCentralityCombined (8)	0.136		0.109	
		RelativeCentralityCombined (7)	0.177		0.177	
Strongest connection		MaxRelatednessCombined (2)			0.047	
Concept groups			Strict	Full	Strict	Full
		AvgClusterSize (2)	0.017	0.033	0.017	0.033
		GroupRelatednessCombined (4)	0.234	0.283	0.132	0.259
Combined		F1, F4-F18, F19, F20 (26)	0.589	0.589	0.584	0.588

Table 6.5: Predictive value of features generated with WordNet concepts

the documents are short and contain a considerable proportion of proper names, most of which do not exist in WordNet. Thus the concept-based model is not as informative as it is with the four experimental datasets in Section 3.1.

Context centrality

The *context centrality* section of Table 6.4 shows that relative centrality is much more informative than local centrality, reflected by the drastic difference in the performance of both individual features and their combinations (abbreviated *LocalCentralityCombined* and *RelativeCentralityCombined* respectively). This finally shows the capability of relative centrality as a measure that directly assesses the connections between texts. It indicates that when modelled properly, relative centrality can be effectively translated to document similarity. In particular, the minimum relative centrality by itself is almost as good as the state-of-the-art methods in Table 6.3.

Local centrality, in contrast, focuses on characteristics of each individual document. Intuitively, it is only helpful when document pairs are similar in other respects, in which case the most coherent pair—with higher local centrality values—is likely to be even more similar than the others, because its component documents are more coherent.

It is worth clarifying that these results do not conflict with the findings in the previous chapter. Here centrality is generalized with four features to describe its overall distribution, and a supervised learning algorithm is used to determine the appropriate mapping to transform them into document similarity. In contrast, local centrality is only helpful when used to highlight representative concepts—the reweighting method described in the previous chapter—and its overall distribution is less relevant to determining the similarity between two documents.

Among the four features of local centrality (F4 to F7), the average value is the most effective. In contrast, every relative centrality feature is quite informative, and as a whole (denoted as *RelativeCentralityCombined*) they achieve a higher correlation than any other single feature type. The binary scheme is slightly better than the weighted scheme, but the difference is not statistically significant.

Strongest connection

The *strongest connection* section shows the predictability of the strongest one-to-one relation between concepts in different texts. Both features strongly predict the inter-document similarity. The distinction between the binary and the weighted representations does not influence their values. We do not consider the weakest concept connection—minimum concept relatedness—because it is zero in most cases and hardly correlates at all with human judgement.

Concept groups

The *strict* and *full* schemes—determined by whether stray concepts that cannot be assigned to any groups are treated as singletons—affect all features in the *concept groups* section, thus their results are shown separately. All results here are obtained using a relatedness threshold of 0.5 for creating the concept cliques.

The first feature—the averaged size of concept groups in each document (F14)—alone yields a correlation of 0.176 and 0.137 in the *strict* and *full* schemes respectively. This feature does not involve a concept’s number of occurrences in a text, so the same result is displayed under the weighted scheme.

The minimum relatedness between concept groups (F16) contributes little in every case. The reason for this is similar to why the weakest connection between

texts is not considered: even texts with very similar topics can mention some unrelated concepts, giving this feature a value close to zero in most cases.

The average size of concept groups is not effective either, especially when compared with the other three features (F15 *MaxGroupRel*, F17 *AvgGroupRel* and F18 *SDGroupRel*). As with the local centrality features, this is probably because it describes characteristics of the text itself, while the others directly target relations between texts.

6.4.4 Effectiveness of combinations of features

The last row of Table 6.4 shows the effectiveness of the combination of all 18 features (F1-F18) discussed so far. However, not every feature is equally useful. The previous discussion shows that five of them are ineffective: the three local centrality features (F4, F5 and F7), the average concept group size (F14) and the minimum relatedness between concept groups (F16). Excluding these features reduces the combination from 18 features and 26 attributes to 13 features and 18 attributes (excluding the class attribute). Table 6.6 compares the performance of the learned model trained before and after removing these five features.

Discarding these uninformative features is actually advantageous, although the improvements are not statistically significant. Table 6.6 also shows that stray concepts are better treated as outliers instead of singleton clusters: the *strict* schemes outperform the *full* schemes and the improvements in both cases are statistically significant. In contrast, the differences between the two *strict* schemes—*binary strict* and *weighted strict*—are not significant.

So far we have not compared the learned similarity measure with previous work on this dataset. All previous research evaluates similarity measures in an unsupervised way: similarity scores are compared directly with human judgement;

	Binary		Weighted	
	Strict	Full	Strict	Full
F1-F18 (26)	0.809	0.799	0.808	0.8
F1-F3, F6, F8-F13, F15, F17, F18 (18)	0.811	0.8	0.809	0.8

Table 6.6: Performance (consistency with human judgement) of the learned measure on the HE50 dataset

and the entire dataset is used for computing correlation. In contrast, here we use five runs of 10-fold cross-validation, and compute correlations only on the held-out test data.

Comparing Table 6.6 with Table 6.3, the learned similarity measure achieves much greater consistency with human judgement than the measures tested by M. D. Lee et al. (2005). It also outperforms the ESA and the ESA-G methods, which represent the state of the art on this dataset. It would be interesting to determine whether these differences are statistically significant. Unfortunately, we cannot do this because results on individual document pairs are not available for the ESA and the ESA-G methods. However, we *can* compare statistical significance with the cosine measure. This will show whether the improvement of the learned similarity over cosine similarity is merely due to chance.

To investigate this, we used the same held-out 10% test data to evaluate the cosine measure: how correlated it is with labelled similarities and how does it compare with the correlation yielded by the learned measures. For the latter, the inter-document similarity is the prediction of the trained regression model, whereas for the former it is calculated by the cosine rule—no machine learning is involved in this case. The cosine measure achieves an average correlation score of 0.56 using the bag-of-words representation and 0.59 using the Wikipedia-based representation over the same five runs of 10-fold cross-validation as those that produced the results in Table 6.6. Each of the eight schemes listed in Table 6.6 provides a statistically significant improvement over the cosine measure, with both representation models.

Results of this evaluation provide strong support for the first hypothesis: with machine learning, the learned measure can predict thematic similarity between texts as consistently as human judgement. In fact, experimental results show that the learned measure is more consistent with an average human rater than they are among themselves. In addition, it is also more consistent than the widely applied cosine measure and the best methods in literature (ESA and ESA-G).

6.5 Evaluation with text clustering

Evaluation in a target application is often suggested as an important addition to evaluation against human judgement (Budanitsky and Hirst, 2001). A similarity measure’s quality is assessed by the extent to which the target application’s performance is improved due to the use of this measure. Text clustering is the target application in this thesis, and this section reports how the learned similarity measure influences clustering.

This evaluation is a crucial addition, because the HE50 dataset is tiny: it only contains 50 documents. It is desirable to test the learned similarity measure on more data, particularly previously unseen data. This evaluation tests it by using it to predict similarities for texts from different sources and different domains that are unseen for the learned measure.

All the features described in the previous sections are generic—they are not specific to any particular dataset but describe the generic characteristics of two texts having similar or different topics. This allows the regression model to be evaluated on completely different test data. We use documents from the four experiment datasets discussed in Section 3.1. This forms a more justified and challenging assessment of how well the measure generalizes to new data. Section 6.5.1 describes the experiment and Section 6.5.2 presents and discusses the results.

6.5.1 Experimental design

We compare the learned measure with three baselines. Two come from Chapter 4: the plain clustering methods with bag-of-words and Wikipedia concept-based representations. The semantically enriched clustering presented in Chapter 5 forms the other baseline. Specifically, we use binary local centrality, because it consistently improves over the above two baselines on all datasets.

The regression model is trained differently in this evaluation. Whereas only 90% of the examples are used previously to build the regression model by using 10-fold cross-validation, here all examples are used to train the model: all 1225 document pairs from the HE50 dataset. The aim is to learn the model as ac-

Clustering algorithm	Datasets	Words Cosine	Concepts Cosine	Rewighted local	Learned measure
<i>k</i> -means	SmallReuters	0.687	0.704	0.724	0.631 \circ
	Med100	0.209	0.284	0.311	0.261 \circ
	NewsSim3	0.008	0.248	0.261	0.143 \circ
	NewsDiff3	0.149	0.579	0.594	0.556 \circ
Hierarchical agglomerative clustering	SmallReuters	0.588	0.623	0.641	0.696*
	Med100	0.276	0.291	0.309	0.365*
	NewsSim3	0.027	0.098	0.102	0.167*
	NewsDiff3	0.18	0.38	0.394	0.613*

*, \circ : statistically significant improvements and degradations

Table 6.7: Performance (normalized mutual information) of the learned measure in clustering the four experimental datasets

curately as possible from the training data. This is safe because here we test the learned measure on previously unseen data. The regression model is built with the *binary strict* scheme and the selected set of features—13 features and 18 attributes.

Again, we test two clustering algorithms: *k*-means and hierarchical agglomerative clustering with group-average-link, and report results in terms of the normalized mutual information (NMI) of the generated clusters.

6.5.2 Experimental results

Table 6.7 compares the learned measure with the three baselines with both clustering algorithms, and shows that in general the learned measure is effective with the hierarchical clustering algorithm but not with *k*-means. The representation of cluster centroids in *k*-means—the mean vector of a cluster’s components—again impacts the learned similarity measure’s performance with this algorithm. The centroid differs from a normal document: for example, it will have a non-zero value for every word or concept mentioned in any document in that cluster. The inferior performance indicates that the learned model might not be applicable to cluster centroids: after all, they are not *real* documents.

In contrast, results with the hierarchical algorithm show that the learned similarity measure is very effective: it outperforms all baselines on every dataset. This is particularly remarkable in three aspects. First, the training dataset is tiny: it

only contains fifty documents yet the learned measure can be effectively applied to much larger datasets. Second, documents in the training dataset are significantly shorter than those in the four experimental datasets—37 words compared to more than a hundred on average—yet the effectiveness of the learned measure is not affected. Third, documents in the training dataset come from different source and cover different topics compared to those clustered, which demonstrates that the learned measure is both generic and robust.

The contrast between the two clustering algorithms inspires the question: what if k -means use an alternative representation of clusters that is based on the individual component text? A group of texts could be represented by its members, and its similarity to another text or a group of texts could be measured by the average similarity with all member texts. This follows the average-link criterion for hierarchical clustering, which takes the average similarity between each cluster’s components as the similarity of two clusters. Yet clusters are built differently: hierarchical clustering builds clusters bottom-up, in each step merging the most similar clusters until the desired number of clusters is met; k -means takes randomly picked seed texts as clusters and iteratively updates cluster membership until the process converges.

With this representation of text clusters, the learned measure obtains drastic improvements using k -means over the baselines: the NMI values of the resulting clusters are 0.792, 0.348, 0.298 and 0.724 on SmallReuters, Med100, NewsSim3 and NewsDiff3 respectively. The improvements over all baselines are statistically significant. This indicates that the alternative representation of clusters is a better fit to the learned measure.

Overall, the results of this evaluation provide strong support for the second hypothesis that the learned measure is more effective than the cosine measure in text clustering. It is worth noting that the traditional cluster representation used in k -means does not appear to match the learned measure, and representing a cluster by its members is a better choice. When applied appropriately, the learned model is very effective, and outperforms the cosine measure with both clustering algorithms.

6.6 Discussion

Intensive research on document similarity measures has been conducted in various fields, including information retrieval, machine learning, natural language processing, and cognitive science (Strehl et al., 2000; M. D. Lee et al., 2005; Mihalcea et al., 2006). The cosine measure is one of the most widely used measures, and has been found to capture human categorization behaviour well in text clustering (Strehl et al., 2000). Therefore it was chosen as the target measure for comparison in this chapter.

Most closely related to the first evaluation task are the ESA (Gabrilovich and Markovitch, 2005) method and its improvement ESA-G (Yeh et al., 2009). ESA indexes documents with Wikipedia concepts based on full-text analysis, and the resulting representation model yields a correlation of 0.72 with human judgement using the cosine measure.

One shortcoming of ESA is that it ignores the rich hyperlink structure between Wikipedia articles. Yeh et al. enrich ESA with hyperlink structure analysis. They use an iterative random walk procedure over the hyperlink graph derived from the Wikipedia hyperlink structure.

The procedure is similar to the PageRank algorithm (Page et al., 1999), except in how the so-called *teleport probability*—the probability of randomly jumping to a page that is not linked to the current page—is chosen. In standard PageRank, this probability is uniformly distributed among all the nodes in the graph, whereas in ESA-G it is focused on the nodes that are associated with a given document, that is, the concepts that ESA identifies for the document. Because each concept corresponds to a Wikipedia page, the random walk favours pages that are associated with the document, and pages that it points to. Thus different documents result in different teleport vectors.

Given a document, the result of this iterative process is a vector whose dimensions correspond to nodes, i.e., Wikipedia pages, with weights indicating their importance. Because all settings except the teleport vector are the same for every walk, differences in the resulting vectors reflect differences in the teleport vector—the concepts identified for the document by ESA. Again, cosine similarity is then used to measure the similarity between the vectors, and this is taken as the simi-

larity between the original documents. This method obtains a correlation of 0.77 with human judgement.

ESA is an expensive technique, because it utilizes the full-text content of Wikipedia articles. ESA-G exacerbates the problem by adding another layer on top of ESA. Furthermore, the Wikipedia hyperlink structure has a considerable link density, which slows down graph walk algorithms significantly (Page et al., 1999).

In contrast, Katoa’s machine learning based method is much more efficient. First, no full-text level analysis is involved: concepts are identified through Wikipedia’s anchor text vocabulary. Second, none of the features used here involves computation at the scale of the entire Wikipedia. However, our method requires labelled training data and the training process to learn the measure, but it still compares favourably to ESA and ESA-G.

Besides the complexity problem, both ESA and ESA-G still rely on the cosine measure: connections between concepts are either not considered (in the case of ESA) or considered only implicitly (in the case of ESA-G). In contrast, our method integrates various aspects in a sound and automatic manner by utilizing machine learning techniques. The results show that it outperforms both ESA and ESA-G.

6.7 Summary

To summarize, this chapter presents a novel method for automatically learning an inter-document similarity measure from a set of features, using regression algorithms. Four types of feature are designed to capture inter-document relations at the document level (i.e., overall similarities), the concept level (i.e., context centralities) and the topic level (i.e., concept groups).

The machine learning method and the features are then evaluated in two distinct tasks: modelling human judgement and performing text clustering. Both evaluations provide strong support for the hypotheses set out at the beginning of this chapter that the learned measure can be as consistent as human judgement on thematic similarities between texts and more effective than the cosine measure in text clustering. Furthermore, by using machine learning, the learned measure

is able to integrate different representation models and semantic connections between concepts in an effective way that makes it the most effective approach for concept-based text clustering.

7

Conclusion and future work

Organizing texts into larger logical structures is a natural human practice that can be traced back to at least 26 B.C.E.. Proper organization can effectively and efficiently reveal the thematic relations among texts, relieving humans from digesting and memorizing excessive amounts of information. Thematic organization of natural language texts is a challenging task to automate: it involves identifying the themes and organizing them appropriately. Both tasks require expertise that humans usually acquire through professional training. For example, librarians learn how to assign metadata to documents to describe their subjects, and developing a thematic classification system such as the *Library of Congress Subject Headings* (LCSH) is a huge project that involves years of work and a great deal of labour.

Clustering is a technique that automatically assesses the thematic similarities between texts and groups them together or separates them into different groups, producing thematically coherent structures of the texts in a fully automatic and cheap fashion. Chapter 1 discussed several drawbacks in the standard text clustering process: semantic ambiguity in the bag-of-words representation model, and the undesirable assumption by similarity measures of orthogonality, which neglects semantic relations among words beyond their surface forms.

This thesis investigates concept-based clustering as a solution for overcoming these drawbacks. A concept is a unit of knowledge; thus each concept is unique. Concepts never exist as isolated blocks, they are always in relation to each other, and these relations are encoded, implicitly or explicitly, in concept systems. A

concept system is a structured organization of concepts, and by definition is constructed based on the relations among them (ISO, 2009).

Concepts and their relations provide just the kind of information that is needed to solve the problems of standard clustering. For example, when a text mentions *kiwi*, it could refer to one of at least three concepts: the New Zealand people, the native bird in New Zealand, and the particular kind of fruit abundant in this country. Resolving ambiguous words and phrases to their intended concepts provides an unambiguous representation of text. Meanwhile, the clustering process can determine whether this text is semantically related to other texts on *New Zealand* from the semantic connections between the two candidate concepts *kiwi* and *New Zealand*—even when they have no words in common at all.

This thesis claims that

Representing text by concepts and taking account of the relations among them can significantly improve text clustering over the bag-of-words representation, using standard clustering algorithms.

To test this hypothesis, we developed the Katoa toolkit that consults two kinds of concept system: WordNet and Wikipedia, for word and world knowledge respectively. Section 7.1 summarizes the findings for each aspect of this hypothesis, and Section 7.2 discusses the research questions set out in the Introduction (Section 1.3). Section 7.3 discusses future work and Section 7.4 concludes the thesis.

7.1 Revisiting the thesis hypothesis

We evaluate three aspects of the thesis hypothesis: 1) how effectively concepts can represent text themes, 2) how effectively the relations among concepts can be used in clustering, and 3) how effectively the different channels of information about similarity can be combined. These aspects were investigated in Chapter 4, 5 and 6, respectively. The following sections discuss and summarize the results of these investigations, and Section 7.1.4 summarizes the findings.

7.1.1 Representing texts by concepts

Both ambiguity and the assumption of orthogonality in the bag-of-words model have long been noticed, yet this does not change the fact that it is still the most prevalent representation. To address these problems, some use alternative features such as phrases, which are semantically more specific than words; others cluster or combine words that frequently co-occur with each other, so that the resulting word clusters or combinations each represents a *latent* topic of the texts. None of these approximations really solve the problems: features still can be ambiguous, and the semantic relations among them are absent and therefore ignored.

Concepts, in contrast, directly target these issues. However, for concept-based models to be widely applicable, they must be consistently effective, generic, and efficient. For this purpose, Katoa uses two concept systems that cover general knowledge—WordNet and Wikipedia—so that texts from both open and specific domains can be handled. The results of Chapter 4 showed that both systems can produce concept-based models that are consistently more effective than the bag-of-words model, across texts from different sources and domains, with the two standard clustering algorithms— k -means and hierarchical agglomerative clustering.

The two systems have their own characteristics: WordNet concepts are more lexical and ubiquitous in text, while Wikipedia concepts are more thematic and concrete. This distinction contributes to the fact that Wikipedia concepts are, in most cases, more effective at obtaining clusters that fit existing classifications based on manually assigned topics or categories. WordNet concepts are advantageous only when topics in the input data are well separated (i.e., for the NewsDiff3 dataset described in Section 3.1), in which case distinctions in lexical aspects help to clarify the distinction between thematically different documents. However, when input data contains similar topics (e.g., the NewsSim3 dataset), lexical features tend to blur the thematic distinctions.

These findings suggest that background knowledge about the texts to be clustered can help one choose the right concept system to consult. However, in practice, such prior knowledge is usually unavailable. After all, the entire clustering task is about organizing texts whose topics are unknown. Thus, Wikipedia, gen-

erally speaking, is a better choice.

We found that the choice of clustering algorithm can affect the best choice of representation model. This thesis tested five clustering algorithms in total: k -means, and hierarchical agglomerative clustering with four different criteria (see Section 2.3). Both of Katoa’s concept-based models outperformed the bag-of-words model with the most effective clustering algorithms: k -means and hierarchical agglomerative clustering using the group-average-link criterion.

We also tested the use of concepts from both systems. However, the results showed only limited success in further improving performance. The trade-off between the extra information and redundancy brought in by the additional features is the most likely reason: the combination usually only improved the inferior concept-based model.

The consistent success of Katoa’s concept-based models provides strong support for the first part of the thesis hypothesis: *representing texts with concepts can significantly improve text clustering over using the bag-of-words model*. It shows that solving the ambiguity problem definitely benefits text clustering. We firmly advocate the use of concept-based models to replace the orthodox bag-of-words paradigm in practice.

7.1.2 Utilizing relations among concepts

In general, there are two ways to utilize semantic relations among concepts: using a specific type of relation, or quantifying the overall relatedness based on all types of relations. This thesis takes the second approach, which is more generic and does not require any parameterization (see Section 1.3). For example, although *dieting* and *smoking* are related, it is difficult to define the exact type of relation that connects them.

This approach requires an effective concept relatedness measure. For WordNet, Katoa uses Leacock and Chodorow (1997)’s path-length based measure (LCH) that computes relatedness among concepts based on the length of the shortest path between them in WordNet. For Wikipedia it uses a measure based on the Wikipedia hyperlink structure (Milne and Witten, 2008b). Both measures have been tested against human judgement on semantic relatedness, and have been

shown to be both effective and efficient (Strube and Ponzetto, 2006; Milne and Witten, 2008a).

Katoa implements three methods for considering concept relatedness during clustering, the purpose being to enhance plain concept-based clustering. The first two assess a concept’s representativeness of a context—its *centrality* with respect to that context—and highlight more central ones by reweighting each concept by its centrality. We investigated two types of centrality: local centrality—a concept’s centrality with respect to its surrounding context—and relative centrality with respect to another document. Local centrality is reweighted for each document before clustering begins, while relative centrality is reweighted during clustering for each document pair. Centrality reflects the variation in a concept’s importance when mentioned in different contexts, by taking its semantic relations with the context into account, whereas the traditional bag-of-words model simply equates importance with the number of occurrences.

However, the problem of connecting texts with different surface forms still exists. Reweighting by relative centrality only influences pairs with some overlap; those with no concepts in common still receive zero similarity. The third method targets this orthogonality problem by altering the similarity measure to take relatedness among different concepts into account. Given a pair of texts, it bridges the surface difference by enriching each text’s representation with concepts that are missing in that text but are mentioned in the other, using concept relations to determine the weight of the enriched concepts. Basically, an enriched concept receives greater weight if it has a strong connection with the current text and is closely related in general. This means that if an enriched concept is unrelated to any concept mentioned in the current text (i.e., it has zero relatedness to all of them), its weight will be zero, meaning that it will not be enriched at all.

Empirical results provide strong support for the second part the thesis hypothesis: *considering the relations among concepts can significantly improve text clustering over using the bag-of-words model*. Furthermore, they show that the plain concept-based clustering method can be further improved (see Table 5.3).

Reweighting by local centrality is the most effective of the three methods: it achieves statistically significant improvements over the plain method in 11 out of 16 cases. The success of local centrality suggests that Katoa’s concept-based

representation models are indeed quite exhaustive and distinctive in capturing important thematic information in the texts. By unifying synonyms and eliminating semantic ambiguity, they provide a strong basis for clustering, and stressing the representative concepts makes the concept-based models even stronger.

The enriched similarity measure is the second most effective method. It consistently improves upon the plain method, but only with the hierarchical clustering algorithm, and less consistent performance was observed with the k -means algorithm (see Tables 5.4 and 5.5). Reweighting by relative centrality—centrality with respect to the other document—is the least effective, although intuitively it should highlight the most relevant aspects between two documents.

Comparing performance with each concept system, we found that Wikipedia was again more effective than WordNet: the number of times the plain method is statistically significantly improved is three times as great as the number of improvements with WordNet concepts (see Table 5.3). Analysis revealed two possible reasons. First is the effectiveness of the concept relatedness measures: the LCH measure for WordNet concepts seems less informative than the WLM measure for Wikipedia concepts. Second, lexical concepts are not necessarily relevant to a text’s theme, which biases the computation of context centrality and thus impacts all three methods.

It is not clear whether or not a concept’s number of occurrences in a text should be taken into account when calculating its context centrality. If it is, relatedness with the more frequently mentioned concepts will be emphasized over those mentioned only occasionally. Our investigation showed that, somewhat surprisingly, the *binary* scheme—only considering the presence or absence of concepts—is more effective than taking frequency into account (i.e., the *weighted* scheme). The enriched similarity measure with Wikipedia is consistently more effective with the weighted scheme than with the binary one. For WordNet concepts, considering occurrence frequencies is likely to do more harm, because lexical features that are thematically unrelated can occur frequently throughout a text, as a common expression for example, which introduces even more bias.

7.1.3 Learning document similarity with concepts

Concepts, and the relations among them, provide additional perspectives for modelling the thematic similarity between two texts. Instead of handcrafting an ad-hoc formula to combine the information from different aspects, we used machine learning techniques—more specifically regression algorithms—to *learn* the right combination from a small amount of training data—texts whose similarities to each other are already known.

Four types of feature were designed, each capturing a single perspective, the overall document similarity measured based on different representation models and similarity measures, the one-to-many, the one-to-one, and the many-to-many relations among the concepts in each text (see Table 6.1). The last three perspectives were represented by the distribution of both types of centrality, the strongest semantic connection between individual concepts, and the distribution of relatedness among their concept cliques respectively. All these features are generic: they are independent of any specific dataset, and can apply to any texts.

Features were evaluated both individually and in combination. Not every feature is equally informative. For example, features that describe the distribution of a concept’s local centrality are less informative because they focus on characteristics of the texts themselves rather the relations among them. In contrast, relative centrality features turned out to be more predictive (see Table 6.4). This indicates that relative centrality, which directly depicts the relation between texts, can be indicative, when utilized appropriately.

Two types of evaluation were conducted: against manually assigned similarities and in the task of clustering. For the former, the goal is to test whether the learned similarity measure can predict similarity as consistently with an average human labeler as they are amongst themselves. The average inter-labeler consistency, in terms of the Pearson correlation coefficient, served as the baseline. Empirical results showed that the learned similarity measure could be even more consistent with humans than they are with each other.

The second evaluation is of greater interest from the point of view of practical application. First, the training dataset is tiny, whereas the test datasets used in this evaluation—the four experimental datasets described in Section 3.1—are

much bigger. Second, and more importantly, the learned model was tested on previously unseen documents, which come from different sources and domains to those in the training dataset. The learned similarity measure was used to replace the standard cosine similarity measure for predicting the similarity between any documents or cluster centroids during clustering. Empirical results showed that it consistently and effectively improves the enriched clustering method (with reweighting by binary local context centrality), which was also the best baseline in this evaluation. We also found that for the learned similarity measure to be effective with the k -means algorithm, the standard mean vector representation of clusters needs to be adapted to represent a cluster by its members instead. This is because the learned measure was trained on relations between individual documents, and the latter representation is a better fit with the underlying model.

There are three options regarding the set of features. The first relates to concept groups. Concepts mentioned in the same document can form tight groups according to their relatedness—the *concept groups* section in Table 6.1. Whether or not stray concepts that cannot be assigned to any existing groups are treated as singleton groups produces the *full* and *strict* models respectively. The fact that the latter always outperformed the former (see Section 6.4.4) suggests that a certain abstraction is necessary: it is beneficial to focus on the major topics and ignore the less important ones—the singleton groups.

The second option concerns occurrence frequencies. Whether or not to take the number of occurrences of a concept into account affects most features, and empirical results showed no significant differences between the *binary* and *weighted* schemes (see Table 6.4).

The third option relates to the concept systems. Empirical results again showed Wikipedia’s advantages over WordNet. With WordNet the learned similarity measure only approximated the average consistency between human raters, whereas with Wikipedia a significantly greater consistency was achieved.

Last but not least, the choice of regression algorithm can also affect the learned similarity measure’s effectiveness. We tested four commonly used regression algorithms, and the one that uses support vector machines for regression with the RBF kernel turned out to be the most effective.

7.1.4 Summary

Based on the findings in each investigation, we can draw the following conclusions:

- Katoa’s concept-based representation models are consistently more effective for text clustering than the traditional bag-of-words model, with the most effective clustering algorithms tested (k -means and hierarchical clustering with group-average-link criterion).
- Prior knowledge about topic distributions in a given collection can help select the most appropriate concept system to use.
- Wikipedia is more effective than WordNet in general, and should be the default choice unless the given text collection is known to be well separable.
- Using concepts from both systems does not usually improve clustering, due to the additional redundancy that is introduced.
- The k -means algorithm and hierarchical agglomerative clustering using the group-average-link criterion are the most effective clustering algorithms.
- Reweighting concepts based on their binary local centrality consistently improves the clustering performance of the Wikipedia-based representation.
- Using the enriched similarity measure with weighted centrality consistently improves the clustering performance of the Wikipedia-based representation if hierarchical clustering (using group-average-link) is used.
- The learned similarity measure is more consistent with an average human than humans are with themselves, with Wikipedia as the concept system.
- The learned similarity measure is the most effective method for concept-based clustering, for both hierarchical agglomerative clustering (using group-average-link) and the adapted k -means method that represents a cluster by its members.

These conclusions strongly favour the concept-based representation models over the bag-of-words model. They also provide a useful guide for applying Katoa

in real-world clustering tasks. Furthermore, Katoa's representation models and similarity measures are not restricted to clustering, but are applicable to any tasks that involve representing texts and computing their thematic similarities.

7.2 Answering the research questions

This research involves several questions, as explained in the Introduction (Section 1.3). Here we discuss them based on the findings of this thesis.

1. What kind of concept system can be used?

Each concept system represents a structural organization of concepts, which are constructed, more or less intentionally, based on the relations among them. Concepts are designated, defined, and explained in such systems. Any system that satisfies these requirements can be used, at least in principle.

Katoa can be easily expanded to consult concept systems other than WordNet and Wikipedia. Adding a new system involves two things: a mechanism for mapping natural language expressions to concepts in the system, which usually involves sense disambiguation, recognizing and unifying synonyms; and a measure for calculating the semantic relatedness among concepts. Of course, the effectiveness of the subsequent clustering depends on the quality of the mapping mechanism and the relatedness measure.

Comparing the overall performance of WordNet and Wikipedia throughout this thesis, the latter tends to be more effective. Although many concepts were captured by both systems, WordNet is particularly helpful in identifying the lexical differences between texts, and Wikipedia can recognize specific names such as particular companies. Because this thesis investigates clustering texts by their topics, the lexical differences become less important, and Wikipedia seems to be more appropriate for the task. This suggests that the characteristics of a concept system and the task both need to be considered.

Concept systems can be either domain independent or domain specific. Although our empirical results showed that general systems are also effective for texts from specific topic domains, (e.g., for the Med100 dataset from the medical domain, see Section 3.1), research shows that domain-specific systems can be

more effective when applied to texts from that domain (Bloehdorn and Hotho, 2004). Nevertheless, WordNet and Wikipedia can serve as the default. After all, they provide comprehensive word and world knowledge respectively.

2. What kind of text can be handled?

As the empirical results showed, the topic domain does not affect Katoa’s advantage over the bag-of-words model. Nor do the topic distributions and document lengths, although the former affects the relative performance of the two concept systems. It is worth noting that the experimental datasets are not ideal: the newest one was created more than a decade ago, whereas information in the concept systems is contemporary, especially for Wikipedia. However, this mismatch did not impact Katoa’s effectiveness; thus contemporariness is not a crucial factor either.

Language is an important factor, and this thesis only uses texts in English. Katoa’s methods can easily be expanded to other languages, at least in principle. Wikipedia is written in 279 languages, while WordNet has been translated into 72 other languages.¹ Alternatively, Katoa could be expanded to a new language by adding a comprehensive concept system in that language: an effective mapping mechanism, which might involve solving additional language-dependent problems such as word segmentation for Chinese, and a concept relatedness measure.

3. How can clustering utilize relations among concepts?

Katoa uses the quantified semantic relatedness among concepts, and implements several methods for highlighting representative concepts and relating texts beyond their surface form. Concept relations were also used to identify topics by constructing cliques of closely related concepts. Empirical results showed the effectiveness of these methods, when used in both an unsupervised ad hoc fashion (i.e., methods in Chapter 5) and in supervised mode (i.e., methods in Chapter 6). The learned similarity measure was shown to be the most effective in terms of high consistency with human judgement on semantic text similarities, and the ability to obtain better clusters.

¹According to the statistics provided by the Global WordNet Association, http://www.globalwordnet.org/gwa/wordnet_table.htm

The accuracy of the concept relatedness measure is important, especially for the enriched clustering methods—the three methods described in Chapter 5. For example, empirical results showed that the WLM measure for Wikipedia concepts (see Section 4.2.3) is more effective than its WordNet counterpart (see Section 4.1.2). However, a measure’s performance is always coupled with the representation model, and so they need to be evaluated together—for example, in the clustering task.

7.3 Future work

This thesis focuses on investigating the impact of concepts and their relations on two aspects of text clustering: representation models and similarity measures. During this PhD I also began to explore several other pertinent aspects of concept-based clustering, and now leave them for future work.

Relations among concepts can be utilized to provide constraints for guiding clustering towards a specified direction, that is, for active learning in clustering. For example, documents with closely related concepts that are assigned to different groups can be presented to users, and, based on the user’s decision, a *must-link* or a *cannot-link* constraint (Wagstaff and Cardie, 2000; Wagstaff et al., 2001) can be formed and used in subsequent clustering. I undertook some initial research in this direction, and presented it in a conference publication (Huang et al., 2008).

I also noticed that relatedness among concepts might need to be adapted to the text collection at hand. For example, in general, *trade* and *coffee* are considered as unrelated or tenuously related, but if they occur in a collection of texts on *coffee imports and exports*, the relatedness between them might need to be strengthened. Alternatively, the strong relation between *coffee* and *cocoa* might need to be weakened if *coffee* only occurs in texts on *expansion of oversea coffee brands in China* and *cocoa* only in agricultural texts. In other words, we consider both *global* relatedness—as retrieved from the concept system—and *local* similarity—associations among concepts in a particular collection. It is of interest to investigate whether considering both aspects could benefit clustering, and how.

Concepts have great potential in providing succinct cluster labels or concise descriptions of cluster contents. Clusters can be visualized based on the relations

among concepts, for example, by constructing concept maps (Sebrechts et al., 1999) to replace the traditional list-based representation of clustering results (Cutting et al., 1992; Zamir and Etzioni, 1999). The description-centric clustering reviewed in Section 2.1.1 motivates us to investigate how effective concepts are in this context. Detecting salient words or phrases that are likely to form good cluster labels is the focus of description-centric clustering, and concept relations can be utilized to facilitate this task.

It would be interesting to test Katoa on contemporary text, and in an on-line fashion—for example, to cluster daily news. Concept relations provide great potential for connecting news that happens in different time frames, because they are likely to have little in common literally, due to the intermittent nature of news articles. Similarly, new articles do not necessarily use the same expressions as existing ones, for example when they try to report from a different perspective, and Katoa can effectively handle this kind of situation.

7.4 Closing remarks

It is no surprise that concepts are better thematic descriptors of text than words. During the 1980s, researchers began to develop formal concept systems like WordNet to facilitate computer processing natural language text, but success was limited and the bag-of-words model still prevails in practice. With the advent of Web 2.0 and the birth of collaboratively constructed, informal yet comprehensive online encyclopedias such as Wikipedia, the use of concepts and their relations began to attract increasing attention as replacement for words and other lexical features.

This thesis provides strong support for why people should be encouraged to abandon the old models and methods. It presents alternatives that are based on concepts, and demonstrates that they are general and effective. These methods are not exclusive to the clustering task, but apply to anything that involves analyzing and organizing texts based on their topics.

References

- Agrawal, R., and Srikant, R. (1994). Fast algorithms for mining association rules. In *Proceedings of the 20th International Conference on Very Large Data Bases (VLDB '94)* (pp. 487–499). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.
- Altmann, G., and Steedman, M. (1988). Interaction with context during human sentence processing. *Cognition*, 30(3), 191–238.
- Baeza-Yates, R. A., and Ribeiro-Neto, B. (1999). *Modern Information Retrieval*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc.
- Baker, L. D., and McCallum, A. K. (1998). Distributional clustering of words for text classification. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 96–103). New York, NY, USA: ACM.
- Banerjee, S., and Pedersen, T. (2003). Extended gloss overlaps as a measure of semantic relatedness. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence* (pp. 805–810). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.
- Banerjee, S., Ramanathan, K., and Gupta, A. (2007). Clustering short texts using Wikipedia. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 787–788). New York, NY, USA: ACM.
- Beil, F., Ester, M., and Xu, X. (2002). Frequent term-based text clustering. In *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 436–442). New York, NY, USA: ACM.
- Bekkerman, R., El-Yaniv, R., Tishby, N., and Winter, Y. (2003). Distributional word clusters vs. words for text categorization. *Journal of Machine Learning Research*, 3, 1183–1208.

REFERENCES

- Bloehdorn, S., and Hotho, A. (2004). Boosting for text classification with semantic features. In *Advances in Web Mining and Web Usage Analysis* (Vol. 3932, pp. 149–166). Berlin, Heidelberg: Springer.
- Breiman, L. (1996). Bagging predictors. *Machine Learning*, 24(2), 123–140.
- Britannica. (2011). *The Encyclopaedia Britannica (2010 copyright)*. <http://www.britannicastore.com/the-encyclopaedia-britannica-2010-copyright/invnt/printset10/>. (Last access March 26, 2011)
- Budanitsky, A., and Hirst, G. (2001). Semantic distance in WordNet: an experimental, application-oriented evaluation of five measures. In *Proceedings of the Workshop on WordNet and Other Lexical Resources, Second Meeting of the North American Chapter of the Association for Computational Linguistics*.
- Caropreso, M. F., Matwin, S., and Sebastiani, F. (2001). A learner-independent evaluation of the usefulness of statistical phrases for automated text categorization. In A. G. Chin (Ed.), *Text Databases and Document Management: Theory and Practice* (pp. 78–102). Hershey, PA, USA: IGI Global.
- Carpineto, C., Osínski, S., Romano, G., and Weiss, D. (2009). A survey of web clustering engines. *ACM Computing Surveys*, 41(3), 17–38.
- Chang, C.-C., and Lin, C.-J. (2001). LIBSVM: a library for support vector machines [Computer software manual]. (Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>)
- Chee, B. W., and Schatz, B. (2007). Document clustering using small world communities. In *Proceedings of the 7th ACM/IEEE-CS Joint Conference on Digital Libraries* (pp. 53–62). New York, NY, USA: ACM.
- Cilibrasi, R. L., and Vitányi, P. M. B. (2007). The Google similarity distance. *IEEE Transactions on Knowledge and Data Engineering*, 19(3), 370–383.
- Cutting, D. R., Karger, D. R., Pedersen, J. O., and Tukey, J. W. (1992). Scatter/Gather: a cluster-based approach to browsing large document collections. In *Proceedings of the 15th Annual International ACM SIGIR Conference on*

- Research and Development in Information Retrieval* (pp. 318–329). New York, NY, USA: ACM.
- Deerwester, S., Dumais, S., Furnas, G., Landauer, T., and Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6), 391–407.
- Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society Bulletin*, 39(1), 1–38.
- Dhillon, I. S. (2001). Co-clustering documents and words using bipartite spectral graph partitioning. In *Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 269–274). New York, NY, USA: ACM.
- Dhillon, I. S., and Modha, D. S. (2001). Concept decompositions for large sparse text data using clustering. *Machine Learning*, 42(1), 143–175.
- El-Yaniv, R., and Souroujon, O. (2001). Iterative double clustering for unsupervised and semi-supervised learning. In *Proceedings of the 12th European Conference on Machine Learning* (pp. 121–132). London, UK: Springer-Verlag.
- Fayyad, U. M., and Irani, K. B. (1993). Multi-interval discretization of continuous valued attributes for classification learning. In *Proceedings of the 13th International Joint Conference on Artificial Intelligence* (Vol. 2, pp. 1022–1027). San Francisco, CA, USA: Morgan Kaufmann Publishers.
- Fellbaum, C. (1998). *WordNet: An Electronic Lexical Database*. Cambridge, MA, USA: The MIT Press.
- Fodeh, S. J., Punch, W. F., and Tan, P. N. (2009). Combining statistics and semantics via ensemble model for document clustering. In *Proceedings of the 2009 ACM Symposium on Applied Computing* (pp. 1446–1450). New York, NY, USA: ACM.
- Freeman, L. C. (1978). Centrality in social networks conceptual clarification. *Social Networks*, 1(3), 215–239.

REFERENCES

- Fung, B. C. M., Wang, K., and Ester, M. (2003). Hierarchical document clustering using frequent itemsets. In *Proceedings of the 3rd SIAM International Conference on Data Mining* (pp. 59–70). San Francisco, CA, USA: The Society for Industrial and Applied Mathematics.
- Gabrilovich, E., and Markovitch, S. (2004). Text categorization with many redundant features: using aggressive feature selection to make SVMs competitive with C4.5. In *Proceedings of the 21st International Conference on Machine Learning* (pp. 41–48). New York, NY, USA: ACM.
- Gabrilovich, E., and Markovitch, S. (2005). Feature generation for text categorization using world knowledge. In *Proceedings of the 19th International Joint Conference on Artificial Intelligence* (pp. 1048–1053). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.
- Gabrilovich, E., and Markovitch, S. (2006). Overcoming the brittleness bottleneck using Wikipedia: Enhancing text categorization with encyclopedic knowledge. In *Proceedings of the 21st National Conference on Artificial Intelligence* (pp. 1301–1306). Menlo Park, CA, USA: AAAI Press.
- Gabrilovich, E., and Markovitch, S. (2007). Computing semantic relatedness using Wikipedia-based explicit semantic analysis. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence* (pp. 1606–1611). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.
- Gabrilovich, E., and Markovitch, S. (2009). Wikipedia-based semantic interpretation for natural language processing. *Journal of Artificial Intelligence Research*, 34(1), 443–498.
- Golub, G. H., and Loan, C. F. V. (1996). *Matrix Computations* (3rd ed.). Baltimore, MD, USA: The Johns Hopkins University Press.
- Gonzalo, J., Verdejo, F., Chugur, I., and Cigarran, J. (1998). Indexing with WordNet synsets can improve text retrieval. In *Proceedings of the COLING/ACL’98 Workshop on Usage of WordNet for NLP*.

- Hammouda, K. M., and Kamel, M. S. (2004). Efficient phrase-based document indexing for web document clustering. *IEEE Transactions on Knowledge and Data Engineering*, 16(10), 1279–1296.
- Harper, D. J., Mechkour, M., and Muresan, G. (1999). Document clustering for mediated information access. In *Proceedings of the 21st BCS-IRSG Annual Colloquium on IR Research*.
- Hartigan, J. A. (1975). *Clustering Algorithms*. New York, NY, USA: John Wiley & Sons, Inc.
- Hartmann, R. R. K., and James, G. (1998). *Dictionary of Lexicography*. London: Routledge.
- Hatzivassiloglou, V., Gravano, L., and Maganti, A. (2000). An investigation of linguistic features and clustering algorithms for topical document clustering. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 224–231). New York, NY, USA: ACM.
- Hearst, M. A., and Pedersen, J. O. (1996). Reexamining the cluster hypothesis: Scatter/Gather on retrieval results. In *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 76–84). New York, NY, USA: ACM.
- Hirst, G., and St-Onge, D. (1997). Lexical chains as representations of context for the detection and correction of malapropisms. In C. Fellbaum (Ed.), *WordNet: An Electronic Lexical Database*. Cambridge, MA, USA: The MIT Press.
- Hofmann, T. (1999). Probabilistic latent semantic indexing. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 50–57). New York, NY, USA: ACM.
- Hotho, A., Staab, S., and Stumme, G. (2003). WordNet improves text document clustering. In *Proceedings of the Semantic Web Workshop at the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*.

REFERENCES

- Hu, J., Fang, L., Cao, Y., Zeng, H.-J., Li, H., Yang, Q., et al. (2008). Enhancing text clustering by leveraging Wikipedia semantics. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 179–186). New York, NY, USA: ACM.
- Huang, A., Milne, D., Frank, E., and Witten, I. H. (2008). Clustering documents with active learning using Wikipedia. In *Proceedings of the 8th IEEE International Conference on Data Mining* (pp. 839–844). Washington, DC, USA: IEEE Computer Society.
- ISO. (2009). *ISO-704: Terminology work—Principles and methods* (3rd ed.). Geneva, Switzerland: International Organization for Standardization. Available from http://www.iso.org/iso/catalogue_detail.htm?csnumber=38109
- Jain, A. K., and Dubes, R. (1988). *Algorithms for Clustering Data*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc.
- Jain, A. K., Murty, M. N., and Flynn, P. J. (1999). Data clustering: a review. *ACM Computing Surveys*, 31(3), 264–323.
- Jarmasz, M. (2003). *Roget’s thesaurus as a lexical resource for natural language processing*. Master thesis, School of Information Technology and Engineering, University of Ottawa, Ottawa, Ontario, Canada.
- Jiang, J. J., and Conrath, D. W. (1997). Semantic similarity based on corpus statistics and lexical taxonomy. In *Proceedings of International Conference Research on Computational Linguistics (ROCLING X)*.
- Jolliffe, I. T. (2002). *Principal Component Analysis* (2nd ed.). New York: Springer-Verlag New York, Inc.
- Jones, K. S. (1971). *Automatic Keyword Classification for Information Retrieval*. Hamden, CT, USA: Archon Books.
- King, B. (1967). Step-wise clustering procedure. *Journal of the American Statistical Association*, 69(317), 86–101.

- Kononenko, I. (1995). On biases in estimating multi-valued attributes. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence* (Vol. 2, pp. 1034–1040). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.
- Leacock, C., and Chodorow, M. (1997). Combining local context and WordNet similarity for word sense identification. In C. Fellbaum (Ed.), *WordNet: An Electronic Lexical Database*. Cambridge, MA, USA: The MIT Press.
- Lee, H.-L. (2010). Organizing knowledge the Chinese way. In *Proceedings of the 73rd ASIS&T Annual Meeting on Navigating Streams in an Information Ecosystem* (Vol. 47, pp. 1–7). Silver Springs, MD, USA: American Society for Information Science.
- Lee, M. D., Pincombe, B., and Welsh, M. (2005). An empirical evaluation of models of text document similarity. In *Proceedings of the 27th Annual Conference of the Cognitive Science Society* (pp. 1254–1259). Mahwah, NJ, USA: Lawrence Erlbaum Associates.
- Lenat, D. B., and Guha, R. V. (1989). *Building large knowledge-based systems; representation and inference in the Cyc project* (1st ed.). Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc.
- Lesk, M. (1986). Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from a ice cream cone. In *Proceedings of the 5th Annual International Conference on Systems Documentation* (pp. 24–26). New York, NY, USA: ACM.
- Lewis, D. D., and Croft, W. B. (1990). Term clustering of syntactic phrases. In *Proceedings of the 13th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 385–404). New York, NY, USA: ACM.
- Lin, D. (1998). An information-theoretic definition of similarity. In *Proceedings of the 15th International Conference on Machine Learning* (pp. 296–304). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

REFERENCES

- MacKay, D. J. C. (1998). Introduction to Gaussian processes. In C. M. Bishop (Ed.), *Neural Networks and Machine Learning* (pp. 133–165). Berlin: Springer.
- Macskassy, S. A., Banerjee, A., Davison, B. D., and Hirsh, H. (1998). Human performance on clustering web pages: a preliminary study. In *Proceedings of the 4th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 264–268). New York, NY, USA: ACM.
- Manning, C. D., Raghavan, P., and Schütze, H. (2008). *Introduction to Information Retrieval*. New York, NY, USA: Cambridge University Press.
- McDonald, S., and Ramscar, M. (2001). Testing the distributional hypothesis: The influence of context on judgements of semantic similarity. In *Proceedings of the 23rd Annual Conference of the Cognitive Science Society* (pp. 611–616). London, UK: Lawrence Erlbaum Associates.
- McGuinness, D. L. (2003). Ontologies come of age. In D. Fensel, J. im Hendler, H. Lieberman, and W. Wahlster (Eds.), *Spinning the Semantic Web: Bringing the World Wide Web to Its Full Potential* (pp. 171–192). Boston, MA, USA: The MIT Press.
- McQueen, J. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability* (Vol. 1, pp. 281–297). Berkeley, CA, USA: University of California Press.
- Medelyan, O., Witten, I. H., and Milne, D. (2008). Topic indexing with Wikipedia. In *Proceedings of the AAAI 2008 Workshop on Wikipedia and Artificial intelligence* (pp. 19–24). Menlo Park, CA, USA: AAAI Press.
- Mihalcea, R., Corley, C., and Strapparava, C. (2006). Corpus-based and knowledge-based measures of text semantic similarity. In *Proceedings of the 21st National Conference on Artificial Intelligence* (pp. 775–780). Menlo Park, CA, USA: AAAI Press.
- Mihalcea, R., and Csomai, A. (2007). Wikify!: linking documents to encyclopedic knowledge. In *Proceedings of the 16th ACM Conference on Conference on*

- Information and Knowledge Management* (pp. 233–242). New York, NY, USA: ACM.
- Miller, G. A. (1985). WordNet: a dictionary browser. In *Proceedings of the 1st International Conference on Information in Data* (pp. 25–28). University of Waterloo, Waterloo, Ontario, Canada.
- Milne, D., and Witten, I. (in press). An open-source toolkit for mining Wikipedia. *Artificial Intelligence*.
- Milne, D., and Witten, I. H. (2008a). An effective, low-cost measure of semantic relatedness obtained from Wikipedia links. In *Proceedings of the AAAI 2008 Workshop on Wikipedia and Artificial intelligence* (pp. 25–30). Menlo Park, CA, USA: AAAI Press.
- Milne, D., and Witten, I. H. (2008b). Learning to link with Wikipedia. In *Proceedings of the 17th ACM Conference on Conference on Information and Knowledge Management* (pp. 509–518). New York, NY, USA: ACM.
- Milne, D., Witten, I. H., and Nichols, D. M. (2007). A knowledge-based search engine powered by Wikipedia. In *Proceedings of the 16th ACM Conference on Conference on Information and Knowledge Management* (pp. 445–454). New York, NY, USA: ACM.
- Mohler, M., and Mihalcea, R. (2009). Text-to-text semantic similarity for automatic short answer grading. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics* (pp. 567–575). Stroudsburg, PA, USA: Association for Computational Linguistics.
- Moschitti, A., and Basili, R. (2004). Complex linguistic features for text classification: a comprehensive study. In *Proceedings of the 26th European Conference on Information Retrieval Research* (pp. 181–196). Berlin, Heidelberg: Springer.
- Ng, A. Y., Jordan, M. I., and Weiss, Y. (2002). On spectral clustering: analysis and an algorithm. In *Advances in Neural Information Processing Systems 14* (pp. 849–856). La Jolla, CA, USA: Neural Information Processing Systems Foundation.

REFERENCES

- Page, L., Brin, S., Motwani, R., and Winograd, T. (1999). *The PageRank citation ranking: bringing order to the web* (SIDL-WP-1999-0120). Stanford InfoLab. Available from <http://ilpubs.stanford.edu:8090/422/>
- Pantel, P., and Lin, D. (2002). Document clustering with committees. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 199–206). New York, NY, USA: ACM.
- Park, Y., Byrd, R. J., and Boguraev, B. K. (2002). Automatic glossary extraction: beyond terminology identification. In *Proceedings of the 19th International Conference on Computational Linguistics* (Vol. 1, pp. 1–7). Stroudsburg, PA, USA: Association for Computational Linguistics.
- Passos, A., and Wainer, J. (2009). WordNet-based metrics do not seem to help document clustering. In *Proceedings of the 2nd Workshop on Web and Text Intelligence*. São Carlos, Brazil.
- Pedersen, T., Patwardhan, S., and Michelizzi, J. (2004). WordNet::Similarity: measuring the relatedness of concepts. In *Demonstration Papers at HLT-NAACL 2004* (pp. 38–41). Stroudsburg, PA, USA: Association for Computational Linguistics.
- Pelleg, D., and Moore, A. (2000). X-means: extending k -means with efficient estimation of the number of clusters. In *Proceedings of the 17th International Conference on Machine Learning* (pp. 727–734). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.
- Pereira, F. C. N., Tishby, N., and Lee, L. (1993). Distributional clustering of english words. In *Proceedings of the 81st Annual Meeting of the Association for Computational Linguistics* (pp. 183–190). Stroudsburg, PA, USA: Association for Computational Linguistics.
- Pincombe, B. (2004). *Comparison of human and latent semantic analysis (LSA) judgements of pairwise document similarities for a news corpus* (Tech. Rep. No. DSTO-RR-0278). Information Sciences Laboratory, Intelligence,

- Surveillance and Reconnaissance Division, Department of Defense, Australia Government. Available from <http://dspace.dsto.defence.gov.au/dspace/bitstream/1947/3334/1/DSTO-RR-0278PR.pdf>
- Porter, M. F. (1980). An algorithm for suffix stripping. *Program*, 14(3), 130–137.
- Potthast, M., Stein, B., and Anderka, M. (2008). A Wikipedia-based multilingual retrieval model. In *Proceedings of the IR research, 30th European Conference on Advances in Information Retrieval Research* (pp. 522–530). Berlin, Heidelberg: Springer-Verlag.
- Quinlan, R. (1993). *C4.5: Programs for Machine Learning*. San Mateo, CA, USA: Morgan Kaufmann Publishers.
- Rada, R., Mili, H., Bicknell, E., and Blettner, M. (1989). Development and application of a metric to semantic nets. *IEEE Transactions on Systems, Man and Cybernetics*, 19(1), 17–30.
- Rasmussen, E. M. (1992). Clustering algorithms. In W. B. Frakes and R. A. Baeza-Yates (Eds.), *Information retrieval: Data structures & algorithms* (pp. 419–442). Upper Saddle River, NJ, USA: Prentice-Hall, Inc.
- Recupero, D. R. (2007). A new unsupervised method for document clustering by using WordNet lexical and conceptual relations. *Information Retrieval*, 10(6), 563–579.
- Rege, M., Dong, M., and Fotouhi, F. (2006). Co-clustering documents and words using bipartite isoperimetric graph partitioning. In *Proceedings of the 6th International Conference on Data Mining* (pp. 532–541). Washington, DC, USA: IEEE Computer Society.
- Rennie, J. (2000). *WordNet::QueryData: a Perl module for accessing the WordNet database*. <http://people.csail.mit.edu/~jrennie/WordNet>. (Last access March 29, 2011)
- Resnik, P. (1995). Using information content to evaluate semantic similarity in a taxonomy. In *Proceedings of the 14th International Joint Conference on Arti-*

REFERENCES

- ficial Intelligence* (pp. 448–453). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.
- Rooney, N., Patterson, D., Galushka, M., and Dobrynin, V. (2006). A scaleable document clustering approach for large document corpora. *Information Processing and Management*, 42(5), 1163–1175.
- Rorvig, M. (1999). Images of similarity: a visual exploration of optimal similarity metrics and scaling properties of TREC topic-document sets. *Journal of the American Society for Information Science and Technology*, 50(8), 639–651.
- Saerens, M., Fouss, F., Yen, L., and Dupont, P. (2004). The principal components analysis of a graph, and its relationships to spectral clustering. In *Proceedings of the 15th European Conference on Machine Learning* (pp. 371–383). London, UK: Springer-Verlag.
- Salton, G., Wong, A., and Yang, C. S. (1975). A vector space model for automatic indexing. *Communications of the ACM*, 18(11), 613–620.
- Schütze, H., and Silverstein, C. (1997). Projections for efficient document clustering. In *Proceedings of the 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 74–81). New York, NY, USA: ACM.
- Scott, S., and Matwin, S. (1999). Feature engineering for text classification. In *Proceedings of the 16th International Conference on Machine Learning* (pp. 379–388). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.
- Sebrechts, M. M., Cugini, J. V., Laskowski, S. J., Vasilakis, J., and Miller, M. S. (1999). Visualization of search results: a comparative evaluation of text, 2D, and 3D interfaces. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 3–10). New York, NY, USA: ACM.
- Senellart, P., and Blondel, V. D. (2003). Automatic discovery of similar words. In M. W. Berry (Ed.), *Survey of Text mining I: Clustering, Classification, and*

- Retrieval* (Vol. 1, pp. 25–42). New York, NY, USA: Springer-Verlag, New York, LLC.
- Shevade, S. K., Keerthi, S. S., Bhattacharyya, C., and Murthy, K. R. K. (2000). Improvements to SMO algorithm for SVM regression. *IEEE Transactions on Neural Networks*, 11(5), 1188–1194.
- Slonim, N., Friedman, N., and Tishby, N. (2002). Unsupervised document classification using sequential information maximization. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 129–136). New York, NY, USA: ACM.
- Slonim, N., and Tishby, N. (2000). Document clustering using word clusters via the information bottleneck method. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 208–215). New York, NY, USA: ACM.
- Sneath, P. H. A., and Sokal, R. R. (1973). *Numerical Taxonomy*. San Francisco, CA, USA: W. H. Freeman.
- Song, W., Li, C. H., and Park, S. C. (2009). Genetic algorithm for text clustering using ontology and evaluating the validity of various semantic similarity measures. *Expert Systems with Applications*, 36(5), 9095–9104.
- Stefanowski, J., and Weiss, D. (2003). Carrot² and language properties in web search results clustering. In *Proceedings of the 1st International Atlantic Web Intelligence conference on Advances in Web Intelligence* (pp. 240–249). Berlin, Heidelberg: Springer-Verlag.
- Steinbach, M., Karypis, G., and Kumar, V. (2000). A comparison of document clustering techniques. In *Proceedings of the Workshop on Text Mining at the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 109–111).
- Stone, B. P., Dennis, S. J., and Kwantes, P. J. (2008). A systematic comparison of semantic models on human similarity rating data: the effectiveness of sub-

REFERENCES

- spacing. In *Proceedings of the 30th Annual Conference of the Cognitive Science Society* (pp. 1813–1818). Austin, TX, USA: Cognitive Science Society.
- Strehl, A., Ghosh, J., and Mooney, R. (2000). Impact of similarity measures on web-page clustering. In *Proceedings of the Workshop of Artificial Intelligence for Web Search in AAAI-2000* (pp. 58–64). Menlo Park, CA, USA: AAAI Press.
- Strube, M., and Ponzetto, S. P. (2006). WikiRelate! computing semantic relatedness using Wikipedia. In *Proceedings of the 21st National Conference on Artificial Intelligence* (Vol. 2, pp. 1419–1424). Menlo Park, CA, USA: AAAI Press.
- Tibshirani, R., Walther, G., and Hastie, T. (2001). Estimating the number of clusters in a data set via the Gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(2), 411–423.
- Tishby, N., Pereira, F. C. N., and Bialek, W. (2000). The information bottleneck method. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 208–215). New York, NY, USA.
- Tsatsaronis, G., Varlamis, I., Vazirgiannis, M., and Nørnvåg, K. (2009). Omiotis: a thesaurus-based measure of text relatedness. In *Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases: Part II* (pp. 742–745). Berlin, Heidelberg: Springer-Verlag.
- van Rijsbergen, C. J. (1979). *Information Retrieval* (2nd ed.). London, UK: Butterworths.
- Varelas, G., Voutsakis, E., Raftopoulou, P., Petrakis, E. G., and Milios, E. E. (2005). Semantic similarity methods in WordNet and their application to information retrieval on the web. In *Proceedings of the 7th Annual ACM International Workshop on Web Information and Data Management* (pp. 10–16). New York, NY, USA: ACM.

- Voorhees, E. M. (1998). Using WordNet for text retrieval. In C. Fellbaum (Ed.), *WordNet: An Electronic Lexical Database* (pp. 285–304). Cambridge, MA, USA: The MIT Press.
- Wagstaff, K., and Cardie, C. (2000). Clustering with instance-level constraints. In *Proceedings of the 17th International Conference on Machine Learning* (pp. 1103–1110). San Francisco, CA, USA: Morgan Kaufmann Publishers.
- Wagstaff, K., Cardie, C., Rogers, S., and Schroedl, S. (2001). Constrained k -means clustering with background knowledge. In *Proceedings of the 18th International Conference on Machine Learning* (pp. 577–584). San Francisco, CA, USA: Morgan Kaufmann Publishers.
- Wang, P., and Domeniconi, C. (2008). Building semantic kernels for text classification using Wikipedia. In *Proceeding of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 713–721). New York, NY, USA: ACM.
- Willett, P. (1983). Similarity coefficients and weighting functions for automatic document classification an empirical comparison. *International Classification*, 10(3), 138–142.
- Willett, P. (1988). Recent trends in hierarchic document clustering: a critical review. *Information Processing and Management*, 24(5), 577–597.
- Witten, I. H., Frank, E., and Hall, M. A. (2011). *Data Mining: Practical Machine Learning Tools and Techniques* (3rd ed.). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.
- WordNet. (2011). *Wnstats (7wn) manual page*. <http://wordnet.princeton.edu/wordnet/man/wnstats.7WN.html>. (Last access March 26, 2011)
- Wu, Z., and Palmer, M. (1994). Verb semantics and lexical selection. In *Proceedings of the 32nd Annual Meeting on Association for Computational Linguistics* (pp. 133–138). Stroudsburg, PA, USA: Association for Computational Linguistics.

REFERENCES

- Xu, W., Liu, X., and Gong, Y. (2003). Document clustering based on non-negative matrix factorization. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 267–273). New York, NY, USA: ACM.
- Yeh, E., Ramage, D., Manning, C. D., Agirre, E., and Soroa, A. (2009). WikiWalk: random walks on Wikipedia for semantic relatedness. In *Proceedings of the 2009 Workshop on Graph-based Methods for Natural Language Processing* (pp. 41–49). Stroudsburg, PA, USA: Association for Computational Linguistics.
- Yoo, I., Hu, X., and Song, I.-Y. (2006). Integration of semantic-based bipartite graph representation and mutual refinement strategy for biomedical literature clustering. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 791–796). New York, NY, USA: ACM.
- Zamir, O., and Etzioni, O. (1998). Web document clustering: a feasibility demonstration. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 46–54). New York, NY, USA: ACM.
- Zamir, O., and Etzioni, O. (1999). Grouper: a dynamic clustering interface to web search results. *Computer Networks*, 31(11-16), 1361–1374.
- Zeng, H. J., He, Q. C., Chen, Z., Ma, W., and Ma, J. (2004). Learning to cluster web search results. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 210–217). New York, NY, USA: ACM.
- Zhai, C., Tong, X., Milic-Frayling, N., and Evans, D. A. (1997). Evaluation of syntactic phrase indexing–CLARIT NLP track report. In D. K. Harman (Ed.), *The 5th Text Retrieval Conference (TREC-5)*. Gaithersburg, MD, USA: NIST Special Publication. Available from http://trec.nist.gov/pubs/trec5/t5_proceedings.html

- Zheng, H.-T., Kang, B.-Y., and Kim, H.-G. (2009). Exploiting noun phrases and semantic relationships for text document clustering. *Information Sciences*, 179(13), 2249–2262.
- Zheng, Z. H., Brady, S., Garg, A., and Shatkay, H. (2005). Applying probabilistic thematic clustering for classification in the TREC 2005 genomics track. In E. M. Voorhees and L. P. Buckland (Eds.), *Proceedings of the 14th Text Retrieval Conference (TREC-14)*. Gaithersburg, MD, USA: NIST Special Publication. Available from http://trec.nist.gov/pubs/trec14/t14_proceedings.html
- Zhu, S., Zeng, J., and Mamitsuka, H. (2009). Enhancing MEDLINE document clustering by incorporating MeSH semantic similarity. *Bioinformatics*, 25(15), 1944–1951.



Example documents and representations

Table A.1 to A.4 show one example document from each of the four experimental datasets used in this thesis (see Section 3.1), and the words and concepts identified from them. The total number of features for each representation is shown in the bracket, in the left column of these tables.

Wikipedia concepts are denoted by the title of their corresponding Wikipedia articles. WordNet concepts are synonym sets; and each synonym consists of three components: its lexical form, the part of speech it is associated with (the first subscript, with 1 meaning nouns, 2 verbs, 3 adjectives and 4 adverbs), and the rank of the sense in all meanings associated with that form and part of speech (the second subscript). For example, the set $\{state_{14}, nation_{11}, country_{11}, land_{19}, commonwealth_{12}, res_publica_{11}, body_politic_{11}\}$ in Table A.1 consists of seven nouns or noun phrases, including the fourth most popular sense of *state*, the ninth sense of *land* and the second sense of *commonwealth*. The number of occurrences for each feature—a word or concept—is shown in the associated bracket.

APPENDIX A. EXAMPLE DOCUMENTS AND REPRESENTATIONS

Document	U.S. EEP VEG OIL PROPOSALS STILL UNDER REVIEW U.S. Agriculture Department proposals to offer 260,000 tonnes of subsidized vegetable oil to four countries are still under consideration by an interagency trade policy group, a USDA official close to the group said. The official, who asked not to be identified, dismissed a report circulating in markets today that the interagency trade policy review group had rejected the proposals. Under the proposals, USDA would offer vegetable oil under the export enhancement program, EEP, to four countries, including 80,000 tonnes to Turkey and 60,000 tonnes to Algeria, Morocco and Tunisia, industry sources said. The proposals “are still under review” by the interagency working group, the USDA official said.							
Words (34)	propos (5) usda (3) agricultur (1) export (1) report (1)	group (4) countri (2) algeria (1) identifi (1) sourc (1)	interag (3) eep (2) circul (1) includ (1) subsid (1)	offici (3) offer (2) close (1) industri (1) todai (1)	oil (3) polici (2) depart (1) market (1) turkei (1)	review (3) trade (2) dismiss (1) program (1) work (1)	tonn (3) veget (2) enhanc (1) reject (1)	
Wikipedia concepts (17)	United States Department of Agriculture (4) Vegetable fats and oils (2) Export (1) Policy Review (1)				Oil (3) Vegetable (2) Turkey (1) Morocco (1) Tonne (3) United States (2) Algeria (1) Working group (1)			
WordNet concepts (39)	{proposal ₁₁ } (5) {group ₁₁ , grouping ₁₁ } (4) {oil ₁₁ } (3) {still ₄₁ } (3) {official ₁₁ , functionary ₁₁ } (3) {metric.ton ₁₁ , MT ₁₂ , tonne ₁₁ , t ₁₃ } (3) {state ₂₁ , say ₂₁ , tell ₂₁ } (3) {vegetable ₁₁ , veggie ₁₁ , veg ₁₁ } (3) {reappraisal ₁₁ , revaluation ₁₁ , review ₁₁ , reassessment ₁₁ } (3) {Department_of_Agriculture ₁₁ , Agriculture_Department ₁₁ , Agriculture ₁₃ , USDA ₁₁ } (3) {offer ₂₁ } (2) {policy ₁₁ } (2) {trade ₁₁ } (2) {state ₁₄ , nation ₁₁ , country ₁₁ , land ₁₉ , commonwealth ₁₂ , res_publica ₁₁ , body_politic ₁₁ } (2) {consideration ₁₁ } (1) {include ₂₁ } (1) {industry ₁₁ } (1) {reject ₂₁ } (1) {today ₁₁ } (1) {work ₂₁ } (1) {uracil ₁₁ , U ₁₁ } (1) {ask ₂₁ , inquire ₂₁ , enquire ₂₁ } (1) {department ₁₁ , section ₁₂ } (1) {enhancement ₁₁ , sweetening ₁₂ } (1) {export ₁₁ , exportation ₁₁ } (1) {go_around ₂₂ , spread ₂₆ , circulate ₂₁ } (1) {identify ₂₁ , place ₂₈ } (1) {have ₂₁ , have_got ₂₁ , hold ₂₄ } (1) {near ₄₁ , nigh ₄₁ , close ₄₁ } (1) {plan ₁₁ , program ₁₁ , programme ₁₆ } (1) {report ₁₁ , study ₁₃ , written_report ₁₁ } (1) {subsidized ₃₁ , subsidised ₃₁ } (1) {Tunisia ₁₁ , Republic_of_Tunisia ₁₁ } (1) {agribusiness ₁₁ , agriculture ₁₁ , factory_farm ₁₁ } (1) {beginning ₁₄ , origin ₁₁ , root ₁₂ , rootage ₁₃ , source ₁₁ } (1) {market ₁₁ , marketplace ₁₁ , market_place ₁₂ } (1) {Algeria ₁₁ , Algeria ₁₁ , Democratic_and_Popular_Republic_of_Algeria ₁₁ } (1) {Morocco ₁₁ , Kingdom_of_Morocco ₁₁ , Maroc ₁₁ , Marruecos ₁₁ , Al-Magrib ₁₁ } (1) {dismiss ₂₁ , disregard ₂₂ , brush_aside ₂₁ , brush_off ₂₁ , discount ₂₁ , push_aside ₂₂ , ignore ₂₂ } (1)							

Table A.1: Document from SmallReuters dataset’s *oil* category

APPENDIX A. EXAMPLE DOCUMENTS AND REPRESENTATIONS

Document	Cutaneous melanoma and bilateral retinoblastoma. We report the case of an otherwise healthy 37-year-old man who had had bilateral enucleation during early childhood for bilateral retinoblastomas, in addition to two cutaneous melanomas (the first appearing at age 27 years). He also had dysplastic melanocytic nevi and a history of cutaneous melanoma in his mother. Retinoblastoma may aggregate in families and is associated with DNA abnormalities of chromosome 13. Recent reports have emphasized the appearance of second malignancies in retinoblastoma survivors. The second malignancies include osteosarcoma, soft tissue sarcoma, and cutaneous melanoma. Cutaneous melanoma also may aggregate in families, usually in the setting of dysplastic melanocytic nevi. The features of this case and of similar reported cases suggest that there may be a greater than expected association between retinoblastoma and cutaneous melanoma.				
Words (38)	melanoma (6) report (3) melanocyt (2) childhood (1) expect (1) includ (1) sarcoma (1) survivor (1)	cutan (6) aggreg (2) year (2) chromosom (1) featur (1) man (1) set (1) tissu (1)	retinoblastoma (5) associ (2) abnorm (1) dna (1) greater (1) mother (1) similar (1) usual (1)	bilater (3) famili (2) addit (1) earli (1) healthi (1) osteosarcoma (1) soft (1)	case (3) malign (2) ag (1) emphas (1) histori (1) recent (1) suggest (1)
Wikipedia concepts (16)	Skin (6) Dysplasia (2) Chromosome (1) Sarcoma (1)	Melanoma (6) Melanocyte (2) DNA (1) Soft tissue (1)	Retinoblastoma (4) Melanocytic nevus (2) Enucleation of the Eye (1) Soft tissue sarcoma (1)	Cancer (2) Nevus (2) Osteosarcoma (1) Tissue (biology) (1)	
WordNet concepts (50)	{cutaneous ₃₁ , cutaneal ₃₁ , dermal ₃₃ } (6) {melanoma ₁₁ , malignant_melanoma ₁₁ } (6) {retinoblastoma ₁₁ } (5) {have ₂₁ , have_got ₂₁ , hold ₂₄ } (4) {case ₁₁ , instance ₁₁ , example ₁₅ } (3) {bilateral ₃₁ , isobilateral ₃₁ , bilaterally_symmetrical ₃₂ , bilaterally_symmetric ₃₁ } (3) {aggregate ₂₁ } (2) {dysplastic ₃₁ } (2) {besides ₄₂ , too ₄₂ , also ₄₁ , likewise ₄₂ , as_well ₄₁ } (2) {birthmark ₁₁ , nevus ₁₁ } (2) {malignancy ₁₁ , malignance ₁₁ } (2) {second ₃₁ , 2nd ₃₁ , 2d ₃₁ } (2) {family ₁₁ , household ₁₁ , house ₁₁ , home ₁₈ , menage ₁₁ } (2) {age ₁₁ } (1) {association ₁₁ } (1) {childhood ₁₁ } (1) {chromosome ₁₁ } (1) {early ₃₁ } (1) {enucleation ₁₁ } (1) {first ₃₁ } (1) {greater ₃₁ } (1) {healthy ₃₁ } (1) {history ₁₁ } (1) {include ₂₁ } (1) {otherwise ₄₁ } (1) {recent ₃₁ } (1) {reported ₃₁ } (1) {sarcoma ₁₁ } (1) {similar ₃₁ } (1) {tissue ₁₁ } (1) {soft ₃₁ } (1) {abnormality ₁₁ , abnormalcy ₁₁ } (1) {addition ₁₁ , add-on ₁₁ , improver ₁₂ } (1) {appearance ₁₁ , visual_aspect ₁₁ } (1) {expect ₂₁ , anticipate ₂₁ } (1) {feature ₁₁ , characteristic ₁₁ } (1) {look ₂₂ , appear ₂₁ , seem ₂₁ } (1) {man ₁₁ , adult_male ₁₁ } (1) {mother ₁₁ , female_parent ₁₁ } (1) {osteosarcoma ₁₁ , osteogenic_sarcoma ₁₁ } (1) {propose ₂₁ , suggest ₂₁ , advise ₂₃ } (1) {report ₂₁ , describe ₂₂ , account ₂₃ } (1) {report ₁₁ , study ₁₃ , written_report ₁₁ } (1) {setting ₁₁ , scene ₁₉ } (1) {survivor ₁₁ , subsister ₁₁ } (1) {old_age ₁₁ , years ₁₁ , age ₁₅ , eld ₁₁ , geezerhood ₁₁ } (1) {associate ₂₁ , tie_in ₂₂ , relate ₂₁ , link ₂₁ , colligate ₂₁ , link_up ₂₃ , connect ₂₂ } (1) {deoxyribonucleic_acid ₁₁ , desoxyribonucleic_acid ₁₁ , DNA ₁₁ } (1) {normally ₄₁ , usually ₄₁ , unremarkably ₄₁ , commonly ₄₁ , ordinarily ₄₁ } (1) {stress ₂₁ , emphasize ₂₁ , emphasise ₂₂ , punctuate ₂₂ , accent ₂₁ , accentuate ₂₁ } (1)				

Table A.2: Document from Med100 dataset’s *Nervous System Diseases* category

APPENDIX A. EXAMPLE DOCUMENTS AND REPRESENTATIONS

Document	Widget source code needed I'm considering writing my own widgets, but I like to have some sample widget source code to look over first. Where could I find something like this? Are there any archives accessible by anonymous ftp that contain such information? Thanks, Edward						
Words (13)	widget (3) edward (1)	sourc (2) find (1)	code (2) ftp (1)	access (1) inform (1)	anonym (1) sampl (1)	archiv (1) write (1)	consid (1)
Wikipedia concepts (4)	GUI widget (3) File Transfer Protocol (2) Source code (2) Anonymity (1)						
WordNet concepts (21)	{doodad ₁₁ , doohickey ₁₁ , widget ₁₁ } (3) {code ₁₁ , codification ₁₂ } (2) {accessible ₃₁ } (1) {archives ₁₁ } (1) {such ₃₁ } (1) {thanks ₁₁ } (1) {Edward ₁₁ , Edward.VIII ₁₁ } (1) {have ₂₁ , have_got ₂₁ , hold ₂₄ } (1) {information ₁₁ , info ₁₁ } (1) {own ₃₁ , ain ₃₁ } (1) {first ₄₁ , firstly ₄₁ , foremost ₄₂ , first_of.all ₄₁ , first_off ₄₁ } (1) {see ₂₅ , consider ₂₁ , reckon ₂₃ , view ₂₁ , regard ₂₁ } (1) {necessitate ₂₁ , ask ₂₆ , postulate ₂₃ , need ₂₁ , require ₂₁ , take ₂₁₄ , involve ₂₄ , call_for ₂₂ , demand ₂₂ } (1) {beginning ₁₄ , origin ₁₁ , root ₁₂ , rootage ₁₃ , source ₁₁ } (2) {look ₂₁ } (1) {sample ₁₁ } (1) {anonymous ₃₁ , anon. ₃₁ } (1) {file_transfer_protocol ₁₁ , FTP ₁₁ } (1) {incorporate ₂₂ , contain ₂₁ , comprise ₂₂ } (1) {write ₂₁ , compose ₂₃ , pen ₂₁ , indite ₂₁ } (1) {find ₂₁ , happen ₂₅ , chance ₂₃ , bump ₂₂ , encounter ₂₂ } (1)						

Table A.3: Document from NewsSim3 dataset's *comp.windows.x* category

Document	Re: Terraforming Venus: can it be done “cheaply”? Would someone please send me James Oberg’s email address, if he has one and if someone reading this list knows it? I wanted to send him a comment on something in his terraforming book. Paul F. Dietz dietz@cs.rochester.edu Potential explosive yield of the annual global production of borax: 5 million megatons						
Words (22)	dietz (2) email (1) paul (1) yield (1)	send (2) explos (1) pleas (1)	address (1) global (1) potenti (1)	annual (1) ha (1) product (1)	book (1) jame (1) read (1)	cheapli (1) list (1) rochest (1)	comment (1) million (1) venu (1)
Wikipedia concepts (10)	James Oberg (2) E-mail address (1) Nuclear weapon yield (1)		Terraforming (2) Explosive material (1) Terraforming of Venus (1)		Borax (1) Fahrenheit (1) Venus (1)		
WordNet concepts (27)	{send ₂₁ , direct ₂₆ } (2) {person ₁₁ , individual ₁₁ , someone ₁₁ , somebody ₁₁ , mortal ₁₁ , soul ₁₂ } (2) {book ₁₁ } (1) {borax ₁₁ } (1) {explosive ₃₁ } (1) {megaton ₁₁ } (1) {production ₁₁ } (1) {read ₂₁ } (1) {Rochester ₁₁ } (1) {Venus ₁₁ } (1) {annual ₃₁ , one-year ₃₁ } (1) {address ₁₁ , computer_address ₁₁ , reference ₁₉ } (1) {desire ₂₁ , want ₂₁ } (1) {electronic_mail ₁₁ , e-mail ₁₁ , email ₁₁ } (1) {have ₂₁ , have_got ₂₁ , hold ₂₄ } (1) {James ₁₁ , James_IV ₁₁ } (1) {know ₂₁ , cognize ₂₁ , cognise ₂₁ } (1) {list ₁₁ , listing ₁₁ } (1) {make ₂₁ , do ₂₁ } (1) {output ₁₂ , yield ₁₁ } (1) {Paul ₁₁ , Alice_Paul ₁₁ } (1) {please ₂₁ , delight ₂₁ } (1) {potential ₃₁ , possible ₃₂ } (1) {rhenium ₁₁ , Re ₁₁ , atomic_number_75 ₁₁ } (1) {stingily ₄₁ , cheaply ₄₁ , chintzily ₄₁ } (1) {global ₃₁ , planetary ₃₄ , world ₃₁ , worldwide ₃₂ , world-wide ₃₁ } (1)						

Table A.4: Document from NewsDiff3 dataset's *sci.space* category

B

Katoa: A toolkit for concept-based text clustering

Katoa (knowledge assisted text organization algorithms) is an open source toolkit for concept-base text clustering developed in this thesis. It is written in Java and built on the Weka machine learning workbench (Witten et al., 2011). It has three modules: data preprocessors such as representation creators, similarity measures, and adapted clustering algorithms. This appendix first describes Weka and other third-party software used by Katoa, and then presents each of its modules.

B.1 Third party software

Katoa is built on top of several well-developed open-source projects, as described below.

Weka is a Java-based data mining framework that contains a collection of machine learning algorithms (Witten et al., 2011). It provides tools for data preprocessing, classification, regression, clustering, association rules, and visualization, and has been widely used in solving practical machine learning and data mining problems. Weka provides a sound infrastructure: new algorithms can be easily added, and published through its package management system. It is available at <http://www.cs.waikato.ac.nz/~ml/weka/index.html>; this thesis uses the 3.7.2 version.

APPENDIX B. KATOA: A TOOLKIT FOR CONCEPT-BASED TEXT CLUSTERING

WordNet::QueryData is a Perl library for querying WordNet terms and concepts (Rennie, 2000). Its most recent version (1.49) is used, which is available at <http://search.cpan.org/dist/WordNet-QueryData/>. Katoa mainly utilizes two functions: **validForms** that calls WordNet’s morphological functions to get the valid form of a given term, which returns an empty string if the term does not exist in WordNet; and **querySense** that retrieves all the concepts (i.e., synsets) associated with the given term. Katoa specifies a term’s part of speech when querying for its associated concepts, and thus only concepts belonging to that part of speech are retrieved.

WordNet::Similarity is a Perl library that implements nine commonly used semantic concept relatedness measures for WordNet (Pedersen et al., 2004), including Leacock and Chodorow (1997)’s path-length-based measure used by Katoa. Katoa uses the most recent version (2.05), which is available at <http://wn-similarity.sourceforge.net/>.

OpenNLP is an Apache project that develops a collection of language-dependent natural language processing models. Katoa uses three models of the English language: tokenizer, sentence detector, and the maximum-entropy-based part of speech tagger, and their most recent versions (1.5.0) are used, which are available at <http://opennlp.sourceforge.net/models-1.5/>.

Wikipedia Miner is a Java-based toolkit that Katoa uses to associate expressions in running text to concepts in Wikipedia (Milne and Witten, in press). It provides scripts to gather information such as anchor texts from Wikipedia, functions for training the sense disambiguator (Milne and Witten, 2008b), and the WLM concept relatedness measure (Milne and Witten, 2008a). The software is available at <http://wikipedia-miner.sourceforge.net/>, and the most recent version (1.1) is used.

The concept systems are required. Katoa uses the most recent WordNet 3.0 version, which is available at <http://wordnet.princeton.edu/wordnet/download/>; and the Wikipedia snapshot taken on March 6, 2009, whose summaries are available from the Wikipedia Miner project as well. Perl and Java runtime environment are also required.

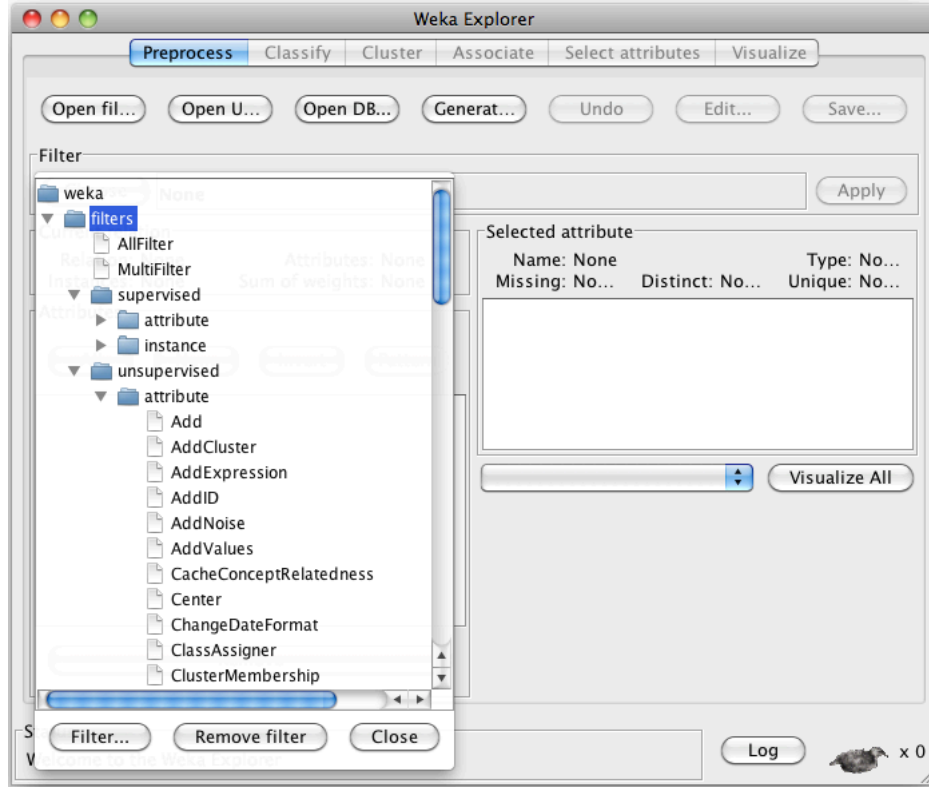


Figure B.1: Weka’s organization structure of data preprocessors

B.2 Concept-based representation creators

Katoa implements the concept-based representation creators as data preprocessors—filters—in Weka. Figure B.1 shows Weka’s structure of filters. Katoa creates two new filters: `StringToWikipediaConceptVector` and `StringToWordNetConceptVector`, both belonging to the unsupervised attribute filters category. They take a dataset that has one or more string attributes as input, and output a new dataset whose attributes correspond to the concepts identified from all string values, together with other non-string attributes in the input such as the class attribute.

Each filter has several options to specify how concepts should be identified, as Figure B.2 shows. They have seven options in common, as listed below:

- **IDFTransform**: sets whether to transform a concept’s weight to its $tf \times idf$ weight.

APPENDIX B. KATOA: A TOOLKIT FOR CONCEPT-BASED TEXT CLUSTERING

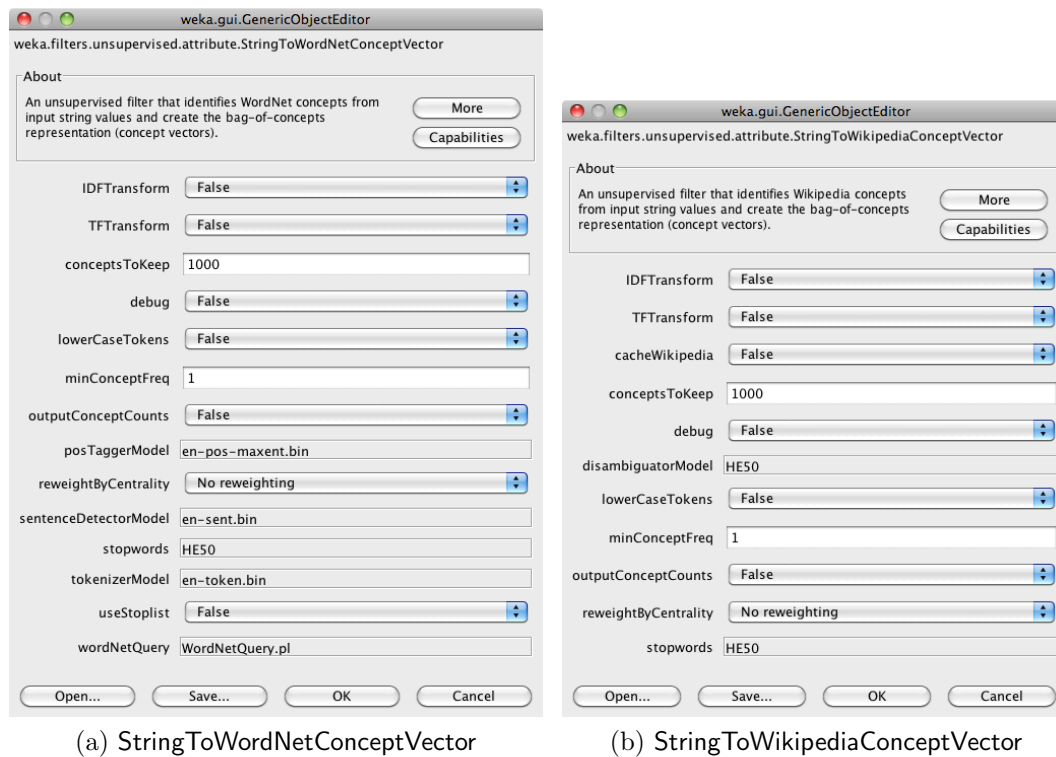


Figure B.2: Options of the filters for creating concept-based text representations

- **TFTransform**: sets whether to transform a concept's term frequency to $\log(1 + tf)$.
- **conceptsToKeep**: sets the maximum number of concepts to be kept. Default is 1000.
- **debug**: sets whether to turn on output of debugging information.
- **lowerCaseTokens**: sets whether to convert all letters to lower case before matching them against terms in the corresponding concept system.
- **minConceptFreq**: sets the minimum concept frequency, and is enforced on an all-classes basis.
- **outputConceptCounts**: sets whether to output concept counts rather than boolean 0 or 1 indicating absence or presence of a concept.

- **reweightByCentrality**: sets whether to weight a concept by its centrality with the local context: $tf \times LocalCentrality$. If so, whether to take concept counts into account (the *reweight by weighted centrality* option) or not (the *reweight by binary centrality* option).

The `StringToWordNetConceptVector` filter has six more options. Three of them concern the natural language processing models, which are language dependent, and alternatives for other languages are available at <http://opennlp.sourceforge.net/models-1.5/>. These options are:

- **posTaggerModel**: sets the part of speech tagger model.
- **sentenceDetectorModel**: sets the sentence detection model.
- **stopwords**: sets the stopwords list to be used. If the **useStoplist** option is turned on and no list is specified with this option, Weka's default stopwords list as described in Section 4.3 will be used.
- **tokenizerModel**: sets the tokenizer model.
- **useStoplist**: sets whether to remove stopwords before matching terms in input text against terms in WordNet.
- **wordNetQuery**: sets the Perl script for querying WordNet.

The `StringToWikipediaConceptVector` filter has three more options:

- **cacheWikipedia**: sets whether to cache relevant Wikipedia information in memory to improve efficiency.
- **disambiguatorModel**: sets the disambiguation model for Wikipedia concepts, which can be trained using the `org.wikipedia.miner.annotation.Disambiguator` class.
- **stopwords**: sets the stopwords list used by the Wikipedia Miner, which should be a plain text file with one stopwords per line. Words and phrases in this list will not be matched against the anchor text vocabulary.

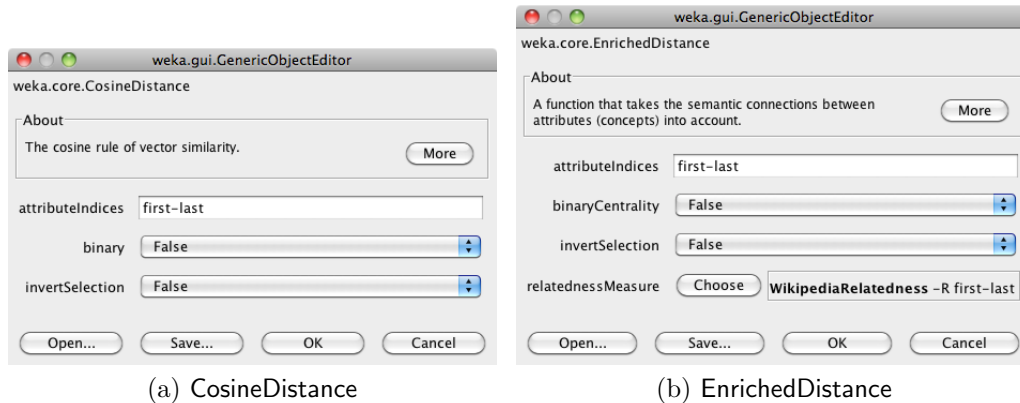


Figure B.3: Options of the plain cosine and the semantically enriched distance functions

B.3 Similarity measures

Weka implements similarity measures as distance functions, and we take $1 - \text{similarity}$ as the distance value, because all similarity measures in this thesis are bounded between 0 and 1. Katoa implements three distance functions: the cosine measure **CosineDistance** (see Section 2.2), **EnrichedDistance** that enriches similarity measure with semantic concept relatedness described in Section 5.4.2, and the machine learned measure **LearnedDistance** described in Chapter 6. Figure B.3 shows the options for each class.

The **CosineDistance** is the most basic measure, with three options:

- **attributeIndices**: specifies a range of attributes that are counted for computing the distance between the given texts, and by default all numeric attributes will be used.
- **binary**: sets whether to count the weights of each attribute or just its presence or absence.
- **invertSelection**: sets whether the range specified with **attributeIndices** is an inverse selection.

The **EnrichedDistance** has four options: **relatednessMeasure**, which specifies the concept relatedness measure; **binaryCentrality**, which specifies whether a concept's

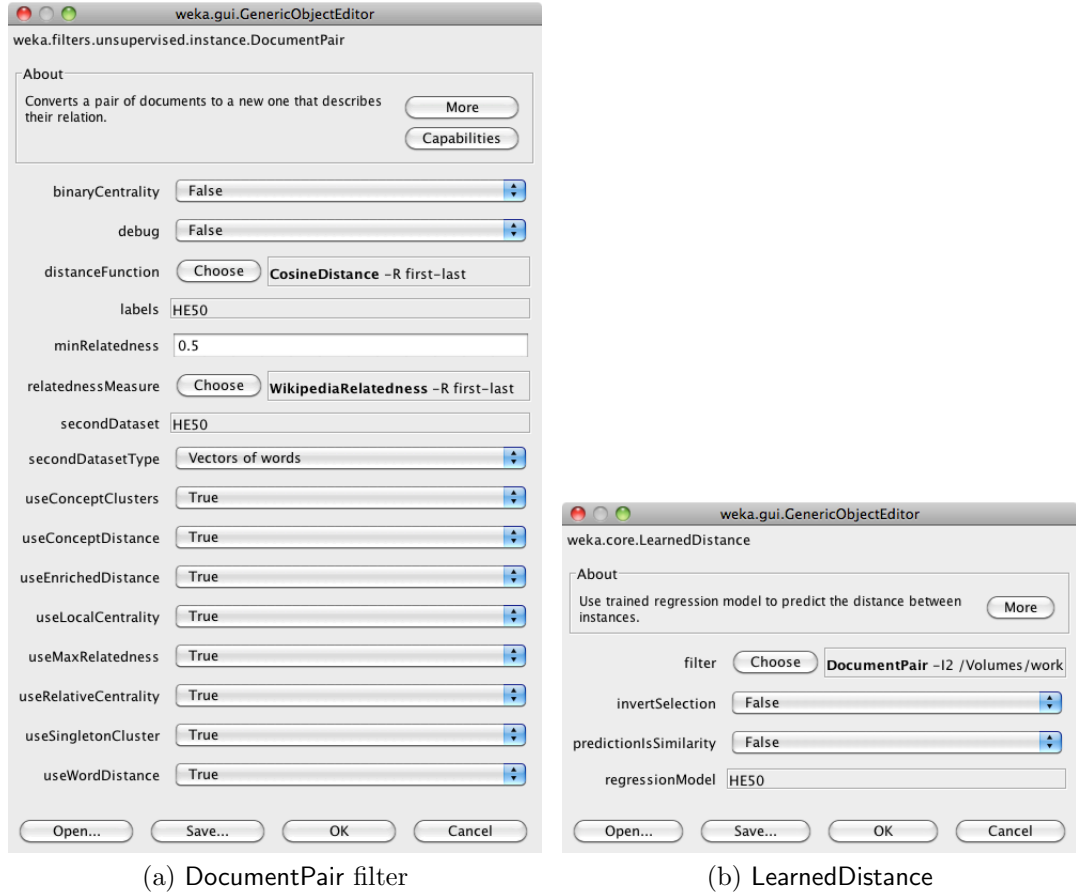


Figure B.4: Options of the DocumentPair filter and the LearnedDistance function

occurrence frequency should be counted when computing its centrality or only its absence and occurrence; and the others—`attributeIndices` and `invertSelection`—are the same as `CosineDistance`.

The `LearnedDistance` class implements the learned similarity measure, as shown in Figure B.4. It involves another unsupervised instance filter: `DocumentPair`, as shown in Figure B.4(a), which converts a pair of documents to a new instance that describes their thematic similarity, using the features described in Section 6.2. Given a pair of texts (their bag-of-words and bag-of-concepts representations), the `LearnedDistance` measure first applies the `DocumentPair` filter to create an instance on their relation, based on which the trained regression model predicts the similarity between the input texts.

As a result, `LearnedDistance`’s `filter` option allows the `DocumentPair` filter to

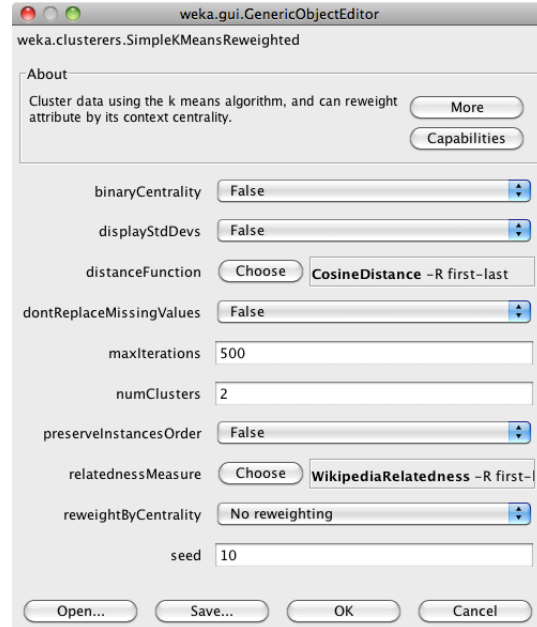


Figure B.5: Options of the **SimpleKMeansReweighted** clustering algorithm

be configured. The `regressionModel` option sets the trained regression model, and `predictionIsSimilarity` sets whether the prediction of the regression model is the similarity rather than the distance between the two texts and thus should be converted to $1 - \text{prediction}$.

B.4 Clustering algorithms

This thesis tested Katoa with two popular clustering algorithms: k -means and hierarchical clustering with different linkage function. The standard k -means algorithm needs to be adapted so as to implement the enriched clustering methods—reweighting by local and relative centrality. Figure B.5 shows the modified k -mean clustering algorithm, which is extended based on Weka’s implementation of k -means. The `binaryCentrality`, `relatednessMeasure` and `reweightedByCentrality` options configure the semantic concept relatedness measure and the reweighting scheme. The remaining options configure the standard k -means clustering algorithm.

B.5 The Katoa toolkit

The toolkit and detailed documentation are available at <http://www.cs.waikato.ac.nz/~lh92/katoa>.



Category distribution of the experimental datasets

Tables C.1 and C.2 show the distribution of categories in each of the four experimental datasets. Table C.3 shows the distribution before restricting category size to 100 documents. Document length is computed as the number of words per document. SmallReuters has the largest number of categories, whereas its documents are the shortest. Documents in the two datasets from the 20Newsgroup collection are much longer. This difference will impact the dimensionality of the bag-of-words model and the concept-based models, hence affect clustering.

As Table C.3 shows, the original OHSUMed dataset is highly unbalanced, which makes this dataset even more difficult. Considering that the main purpose of this dataset is to test how Katoa can be applied to domain-specific texts, balancing the categories reduces the bias that might be introduced by an unbalanced category distribution.

APPENDIX C. CATEGORY DISTRIBUTION OF THE EXPERIMENTAL DATASETS

Category	Num. Docs	Avg. Length
1 comp.os.ms-windows.misc	985	255.1
2 comp.windows.x	980	360.8
3 comp.graphics	973	327.2
total	2938	314.2

(a) NewsSim3

Category	Num. Docs	Avg. Length
1 rec.sport.baseball	994	274.7
2 sci.space	987	357.2
3 alt.atheism	799	391.8
total	2780	337.6

(b) NewsSim3

Table C.1: Category distribution of the NewsSim3 and NewsDiff3 datasets

Category	Num. Docs	Avg. Length
1 money-supply	161	105.2
2 ship	158	148
3 sugar	143	169.4
4 coffee	116	209.1
5 gold	99	150.8
6 gnp	83	228.2
7 cpi	79	114.7
8 cocoa	63	213.9
9 jobs	55	117.8
10 copper	54	150
11 reserves	53	114
12 grain	51	189.2
13 alum	50	140.8
14 ipi	49	140.7
15 iron-steel	47	146.2
16 nat-gas	45	160.1
17 rubber	41	213.7
18 veg-oil	37	202.9
19 bop	32	160.3
20 tin	30	229.9
21 cotton	26	137.1
22 wpi	26	110.8
23 orange	22	111.4
24 gas	22	179
25 retail	22	176.9
26 pet-chem	21	127.9
27 livestock	20	135.7
28 strategic-metal	19	145.7
29 housing	18	110.1
30 zinc	16	126.3
total	1658	157.2

(a) SmallReuters

Category	Num. Docs	Avg. Length
1 Cardiovascular Diseases	100	187
2 Neoplasms	100	184.1
3 Pathological Conditions, Signs and Symptoms	100	192.8
4 Nervous System Diseases	100	195.5
5 Disorders of Environmental Origin	100	167.1
6 Immunologic Diseases	100	207.3
7 Urologic and Male Genital Diseases	100	145.9
8 Digestive System Diseases	100	196.1
9 Nutritional and Metabolic Diseases	100	153.4
10 Respiratory Tract Diseases	100	173.3
11 Bacterial Infections and Mycoses	100	164.3
12 Skin and Connective Tissue Diseases	100	191.6
13 Musculoskeletal Diseases	100	175.6
14 Female Genital Diseases and Pregnancy Complications	100	222.9
15 Neonatal Diseases and Abnormalities	100	194.2
16 Eye Diseases	100	184.9
17 Hemic and Lymphatic Diseases	100	177.5
18 Virus Diseases	100	226.9
19 Endocrine Diseases	100	227.9
20 Parasitic Diseases	100	202.3
21 Otorhinolaryngologic Diseases	100	182.5
22 Stomatognathic Diseases	100	178.5
23 Animal Diseases	56	205
total	2256	188.2

(b) Med100

Table C.2: Category distribution of the SmallReuters and Med100 datasets

Category	Num. Docs	Avg. Length
1 Cardiovascular Diseases	2876	225.5
2 Neoplasms	2513	198.1
3 Pathological Conditions, Signs and Symptoms	1924	196.1
4 Nervous System Diseases	1328	174.3
5 Disorders of Environmental Origin	1283	171.4
6 Immunologic Diseases	1060	203.8
7 Urologic and Male Genital Diseases	842	193.5
8 Digestive System Diseases	837	200.4
9 Nutritional and Metabolic Diseases	815	222.3
10 Respiratory Tract Diseases	634	198.2
11 Bacterial Infections and Mycoses	631	185.4
12 Skin and Connective Tissue Diseases	592	176.3
13 Musculoskeletal Diseases	505	169.2
14 Female Genital Diseases and Pregnancy Complications	473	178
15 Neonatal Diseases and Abnormalities	356	190.1
16 Eye Diseases	337	156.5
17 Hemic and Lymphatic Diseases	307	208.1
18 Virus Diseases	249	188.1
19 Endocrine Diseases	200	231.3
20 Parasitic Diseases	183	200
21 Otorhinolaryngologic Diseases	169	153
22 Stomatognathic Diseases	132	150.3
23 Animal Diseases	56	205
total	18302	196.2

Table C.3: Category distribution of the original OHSUMed datasets

D

Results of different disambiguation techniques on WordNet

Table D.1 compares the two disambiguation strategies for WordNet—the most common sense rule (denoted by MCS) and the context-based disambiguation (see Section 4.1.2). Both strategies were tested in clustering the four experimental datasets with k -means algorithm and the exact number of clusters—the same setup as in Section 4.4.3. It shows that the former is consistently more effective across datasets and evaluation measures, and the improvements are statistically significant (with paired t -test and $p = 0.05$, see Section 3.2).

The extremely fine-grained distinction between WordNet meanings is the most likely reason for the unsatisfactory performance of the context-based approach.

Dataset	Strategy	Purity	InvPurity	NMI	FMeasure
SmallReuters	MCS	0.675*	0.665*	0.697*	0.501*
	Context	0.502	0.461	0.496	0.432
Med100	MCS	0.258*	0.277*	0.211*	0.124*
	Context	0.151	0.208	0.115	0.076
NewsSim3	MCS	0.389*	0.931*	0.056*	0.498*
	Context	0.346	0.923	0.003	0.492
NewsDiff3	MCS	0.904*	0.933*	0.767*	0.864*
	Context	0.872	0.913	0.752	0.831

*: statistically significant improvements

Table D.1: Performance of the most-common-sense-rule and context-based disambiguation strategies for WordNet

APPENDIX D. RESULTS OF DIFFERENT DISAMBIGUATION TECHNIQUES ON WORDNET

For example, WordNet has 16 and 41 senses (i.e., concepts) for *run* as a noun and a verb respectively. These results suggest that the subtle distinctions in meanings might be undesirable for the text clustering task.

The most-common-sense-rule strategy has its limitation as well: sometimes the first sense might not be the most popular one, because the ranking is determined based on a particular corpus (the British National Corpus). For example, the first sense of *tiger* is *a fierce or audacious person* rather than the feline animal. Nevertheless, performance in Table D.1 shows that it does not impact the strategy’s overall effectiveness in clustering.



Configuration of the regression algorithms

In Chapter 6.3 we tested four regression algorithms for learning semantic document similarity from human labelled training data. They are: linear regression; support vector machines (SVM) for regression (abbreviated *SVMreg*) (Shevade et al., 2000); *LibSVM*, which uses the *libsvm* tools (Chang and Lin, 2001) to build SVM classifiers; and the Gaussian process for regression (denoted *GaussianProcess*) (MacKay, 1998). All these methods are commonly used techniques and are implemented in the Weka software (Witten et al., 2011). We describe the configuration used to produce the the results in Section 6.3.

The best performance of the non-linear regression algorithms—*SVMreg*, *LibSVM* and *GaussianProcess*—was achieved with the radial basis function (denoted by *RBFKernel* in *SVMreg* and *GaussianProcess*). Two Other kernel functions were also tested *SVMreg* and *LibSVM* both use support vector machines for regression, and differ in that *SVMreg* used the first type epsilon-SVM regression and *LibSVM* used the second type nu-SVM regression. *SVMreg* and *GaussianProcess* both standardized the training data, and *LibSVM*'s best performance was achieved when the training data was normalized. Standardization makes all numeric attributes, except for the class attribute, have zero mean and unit variance. Normalization translates all numeric attributes, except for the class attribute, into the $[0, 1]$ range.

Except for the above configurations, default values were used for all the re-

APPENDIX E. CONFIGURATION OF THE REGRESSION ALGORITHMS

maintaining options, including those of the linear regression algorithm. Combinations of different parameter values were tested, using grid search and cross-validation. Yet statistically significant differences were observed on very few occasions, when it is likely that the model has overfit the dataset, for example, when assigning greater values to SVMreg's complexity parameter and RBF kernel's gamma parameter. Therefore, the default values were adopted.